code4sp
coding for social promotion

Με συγχρηματοδότηση από το πρόγραμμα «Erasmus+» της Ευρωπαϊκής Ένωσης

Το ηλεκτρονικό βιβλίο Code4SP

WP3:

Εκπαιδευτικό Υλικό του Code4SP

Εκπονήθηκε από:

CD DR

ESPE

social hackers academy

CITIZENS IN POWER

Πληροφορίες του Έργου

Ακρώνυμο του Έργου: Code4SP

Τίτλος Έργου: Κωδικοποίηση για Κοινωνική Ένταξη/ Ενσωμάτωση

Αριθμός Έργου: 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN

Ιστοσελίδα του έργου: www.code4sp.eu

Εταίρος συντάκτης: CodeDoor, CEPROF, SHA and CIP

Εκδοχή/Έκδοση Εγγράφου: 2

Ημερομηνία Προετοιμασίας:15/04/2022

Ιστορικό Εγγράφου			
Ημερομηνία	Έκδοση/Εκδοχή	Συντάκτης	Περιγραφή
04/04/2022	1	Διάφοροι Συντάκτες	Προσχέδιο
15/04/2022	2	Διάφοροι Συντάκτες	Τελική Εκδοχή

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εισαγωγή

Αυτό το ηλεκτρονικό βιβλίο αποτελεί μέρος του έργου KA3 Code4SP, με αριθμό αναφοράς 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN, που συγχρηματοδοτείται από το πρόγραμμα Erasmus+, KA3 – για τη Μεταρρύθμιση Πολιτικής, από την Ευρωπαϊκή Επιτροπή.

Το έργο συντονίζεται από το SPEL και η σύμπραξη αποτελείται από επτά ακόμη εταίρους (CEPROF, CIP – Citizens in Power, CSI - Center for Social Innovation Ltd., CODEDOOR. ORG EV, ZAUG, Action Sinergy SA και Social Hackers Academy), από τέσσερις χώρες (Κύπρος, Ελλάδα, Γερμανία και Πορτογαλία).

Το έργο αυτό έχει ως στόχο:

- Να προωθήσει την κοινωνικοοικονομική πρόοδο, με την παροχή κατάρτισης για την ένταξη στην αγορά εργασίας στον τομέα του προγραμματισμού.

- Να μεταφέρει τις καλές πρακτικές που εφαρμόζονται σήμερα στη Μη τυπική εκπαίδευση, στον τομέα του προγραμματισμού, στις χώρες της Νότιας Ευρώπης οι οποίες θεωρούνται ταυτόχρονα πιο οικονομικά ευάλωτες και υπόκεινται σε έκθεση, άνευ προηγουμένου, σε μεταναστευτικά κύματα ατόμων υπό δυσμενείς κοινωνικοοικονομικές συνθήκες.

Το ηλεκτρονικό βιβλίο προκύπτει από το περιεχόμενο του οδηγού υλοποίησης που έχει ήδη αναπτυχθεί στο πλαίσιο του έργου, προκειμένου να συγκεντρωθεί το εκπαιδευτικό υλικό για τους εκπαιδευτές, έτσι ώστε να μπορούν να εφαρμόσουν τη μεθοδολογία του Code4SP στις χώρες τους. Το εκπαιδευτικό υλικό αποτελείται από σενάρια μαθημάτων, παρουσιάσεις στο PowerPoint, έτοιμα μαθήματα και οποιαδήποτε άλλη προσαρμοσμένη διδακτέα ύλη που θα προκύψει από την αρχική μεθοδολογία του CodeDoor, καθώς και από επιτυχημένες βέλτιστες πρακτικές από κάθε εθνικό επίπεδο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Table of Contents

Εισαγωγή 3
1. Προγραμμα-τισμός Η/Υ και βασικές έννοιες 5
Πληροφορίες για την ενότητα 6
2. HTML – HyperText Markup Language 21
Πληροφορίες για την ενότητα 22
2.1. Τα βασικά της HTML 24
2.2. Προηγμένες έννοιες της HTML 75
2.3 Λειτουργίες στην HTML5 91
2.4 Αναφορές της HTML5 130
3. CSS – Cascading Style Sheets 131
Πληροφορίες αναφορικά με την ενότητα 132
3. CSS 132
3.1 Σύντομη επισκόπηση στην CSS 135
3.2 Προηγμένοι κανόνες σύνταξης CSS 176
3.2. Το CSS Responsive Web Design (RWD) 196
4. SQL - Structured Query Language 226
Πληροφορίες για την ενότητα 227
4.1. Τα βασικά της SQL 230
4.2. Βάση δεδομένων στην SQL 290
4.3. Αναφορές στην SQL (References) 324
4.4. Παραδείγματα της SQL 333
5. JavaScript 334
Πληροφορίες για την ενότητα 335
5.1. JavaScript Basic (CEPROF) 338
5.2. JavaScript & DOM 421
5.3. JavaScript & BOM 452
5.4. JavaScript Advanced 473

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

1. Προγραμματισμός Η/Υ και βασικές έννοιες

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πληροφορίες για την ενότητα

Ενότητα:
1. Προγραμματισμός Η/Υ και βασικές έννοιες

Προϋποθέσεις:
Βασικές γνώσεις ηλεκτρονικών υπολογιστών, εγκατάσταση βασικού λογισμικού και βασικές γνώσεις επεξεργασίας αρχείων.

Φόρτος εργασίας:
5 ώρες

Περιγραφή:
Στο υποκεφάλαιο αυτό, θα δώσουμε μια σύντομη εισαγωγή στον προγραμματισμό ηλεκτρονικών υπολογιστών και τις βασικές έννοιες που σχετίζονται με αυτόν. Θα αναλύσουμε έννοιες υλισμικού και λογισμικού ηλεκτρονικών υπολογιστών, καθώς και τον τρόπο με τον οποίο οι υπολογιστές επεξεργάζονται τα δεδομένα. Στο τέλος, θα ρίξουμε μια ματιά για το πώς λειτουργεί ο προγραμματισμός και τι είδους γλώσσες προγραμματισμού είναι διαθέσιμες.

Μαθησιακά αποτελέσματα:
- Οι εκπαιδευόμενοι θα είναι σε θέση να αναγνωρίζουν την έννοια της Γλώσσας Σήμανσης Υπερκειμένου/ HyperText Markup Language (HTML) από την οικογένεια των γλωσσών περιγραφής και επεξεργασίας εγγράφων.
- Θα μπορούν να διακρίνουν τη δομή, το περιεχόμενο και το στυλ Διάκριση της δομής, του περιεχομένου και τα στυλ μιας σελίδας.
- Θα μπορούν να χρησιμοποιούν το HTML για τον σχεδιασμό ιστοσελίδων στο διαδίκτυο.

Απαιτούμενα υλικά:
- Υπολογιστής ή φορητός υπολογιστής
- Σύνδεση στο Διαδίκτυο

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σενάριο μαθήματος:

Ο συνολικός χρόνος για αυτό το θέμα είναι 10 ώρες και θα εναπόκειται στον εκπαιδευτή να αποφασίσει πόσο χρόνο θα αφιερώσει στη διδασκαλία κάθε υποθέματος. Προκειμένου να αξιοποιήσουμε στο έπακρο όλο τον διαθέσιμο χρόνο, προτείνουμε τη χρήση του εκπαιδευτικού υλικού που συντάχθηκε στο πλαίσιο του έργου (παρουσιάσεις PPT), το οποίο σχεδιάστηκε με γνώμονα την αποτελεσματική χρήση του χρόνου. Αυτές οι παρουσιάσεις αποτελούνται από τα ακόλουθα στοιχεία:

- Ανάπτυξη του υποθέματος και των βασικών ιδεών που πρέπει να διατηρηθούν.
- Προτεινόμενες δραστηριότητες.

Επομένως, εάν ο εκπαιδευτής ακολουθήσει τη λογική σειρά των PPTs, σίγουρα θα μπορέσει να ολοκληρώσει τη συνεδρία εντός του καθορισμένου χρονικού ορίου. Αυτές οι παρουσιάσεις μπορούν επίσης να διατεθούν στους μαθητές για ατομική μελέτη.

Υποθέματα:

1.1. Εισαγωγή στους υπολογιστές και στον προγραμματισμό

1.2. Έννοια υλισμικού και λογισμικού

1.3. Πώς αποθηκεύουν τα δεδομένα οι υπολογιστές

1.4. Πώς λειτουργεί ένα πρόγραμμα

1.5. Προγράμματα και γλώσσες προγράμματος

Επιπλέον πόροι:

- Khan Academy: χρήσιμες πηγές σε θέματα ηλεκτρονικών υπολογιστών σε διάφορες γλώσσες
- Marc Andreessen: https://future.a16z.com/so software-is-eating-the-world/

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

1.1 Εισαγωγή στους υπολογιστές και στον προγραμματισμό

Οι υπολογιστές είναι μηχανές που μπορούν να προγραμματιστούν για την εκτέλεση μιας ακολουθίας εντολών. Ο προγραμματισμός είναι η διαδικασία σχεδιασμού αυτών των οδηγιών. Τα προγράμματα είναι γραμμένα σε μια συγκεκριμένη γλώσσα που παρέχει μια δομή για τον προγραμματιστή και χρησιμοποιεί συγκεκριμένες οδηγίες για τον έλεγχο της ακολουθίας των λειτουργιών που εκτελεί ο υπολογιστής.
«Software is eating the world» (Marc Andreessen, https://future.a16z.com/soware-is-eating-the-world/), οπότε ενημερώστε τους μαθητές σας για το γεγονός ότι στις μέρες μας σχεδόν τα πάντα είναι μηχανογραφημένα και προγραμματισμένα (βλ. επίσης: Internet of Things).

Με ποιους τρόπους μπορεί κανείς να χρησιμοποιήσει τους ηλεκτρονικούς υπολογιστές;

Αφήστε τους μαθητές σας να κάνουν ένα καταιγισμό ιδεών ή να σχεδιάσουν ένα νοητικό χάρτη για το πώς χρησιμοποιούν στην πραγματικότητα τους υπολογιστές. Υπάρχουν πολλοί τρόποι με τους οποίους μπορείτε να χρησιμοποιήσετε τους ηλεκτρονικούς υπολογιστές στην καθημερινή σας ζωή. Ακολουθούν μερικοί από τους πιο συνηθισμένους τρόπους:

- Για να περιηγηθείτε στο διαδίκτυο
- Για να ελέγξετε το ηλεκτρονικό σας ταχυδρομείο
- Για να κάνετε έρευνα για ένα έργο
- Για να παίξετε παιχνίδια
- Για να παρακολουθήσετε ταινίες ή τηλεοπτικές εκπομπές
- Για να ακούσετε μουσική
- Για να επεξεργαστείτε ένα έγγραφο ή ένα λογιστικό φύλλο
- Για να διαβάσετε τις ειδήσεις

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ποιες άλλες συσκευές είναι υπολογιστικά συστήματα;

Οι υπολογιστές δεν είναι μόνο οι συσκευές που βρίσκονται στα γραφεία μας ή στις τσέπες μας. Είναι επίσης οι συσκευές που λειτουργούν τα αυτοκίνητα, τα αεροπλάνα και τα πλοία. Είναι στις τηλεοράσεις μας, στα ψυγεία μας, ακόμα και στα ρολόγια μας. Εν ολίγοις, οι υπολογιστές είναι παντού.

Τι είδους λογισμικό έχετε χρησιμοποιήσει;

Αφήστε τους μαθητές σας να κάνουν ένα καταιγισμό ιδεών ή να σχεδιάσουν ένα νοητικό χάρτη για το λογισμικό που έχουν ήδη χρησιμοποιήσει. Προσπαθήστε να τους κάνετε να συνειδητοποιήσουν ότι κάθε λογισμικό αναπτύχθηκε από κάποιον προγραμματιστή. Ακόμη και το λογισμικό που είναι εγκατεστημένο στα smartphone κινητά, στα μηχανήματα αυτόματης ανάληψης μετρητών ή ακόμα και στις τηλεοράσεις που χρησιμοποιούμε καθημερινά.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

1.2 Οι έννοιες του υλισμικού και του λογισμικού

Ορισμός του υλισμικού και του λογισμικού

Το υλισμικό αναφέρεται στα φυσικά στοιχεία ενός συστήματος υπολογιστή, ενώ το λογισμικό αναφέρεται στις εντολές και τα δεδομένα που κάνουν τον υπολογιστή να λειτουργεί.

Τα κύρια μέρη ενός υπολογιστή και οι λειτουργίες τους

Ένας υπολογιστής αποτελείται από 4 βασικά μέρη: την κεντρική μονάδα επεξεργασίας (CPU), τη μνήμη, τις συσκευές εισόδου και τις συσκευές εξόδου. Η κεντρική μονάδα επεξεργασίας/μητρική πλακέτα (CPU) είναι το μέρος του υπολογιστή που εκτελεί τους υπολογισμούς και ελέγχει τα άλλα μέρη. Η κύρια μνήμη είναι το μέρος όπου ο υπολογιστής αποθηκεύει τα δεδομένα που επεξεργάζεται. Οι συσκευές εισόδου είναι οι συσκευές εκείνες που χρησιμοποιεί ο χρήστης για την εισαγωγή δεδομένων στον υπολογιστή, όπως το πληκτρολόγιο και το ποντίκι. Οι συσκευές εξόδου είναι οι συσκευές εκείνες που χρησιμοποιεί ο υπολογιστής για να εμφανίζει τα αποτελέσματα των υπολογισμών του, όπως το μόνιτορ και ο εκτυπωτής.

1.3 Πώς αποθηκεύουν τα δεδομένα οι υπολογιστές

Έννοια: το δυαδικό σύστημα

Ένα δυαδικό σύστημα είναι ένας τρόπος αναπαράστασης πληροφοριών χρησιμοποιώντας δύο σύμβολα: το 0 και το 1. Το δυαδικό σύστημα είναι η απλούστερη μορφή αναπαράστασης πληροφοριών και χρησιμοποιείται στα υπολογιστικά συστήματα για την αποθήκευση και την επεξεργασία πληροφοριών. Το δυαδικό σύστημα είναι ένας εύχρηστος τρόπος αναπαράστασης πληροφοριών επειδή είναι πολύ απλός και μπορεί να επεξεργαστεί πολύ εύκολα από τα υπολογιστικά συστήματα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ASCII

Τα υπολογιστικά συστήματα αποθηκεύουν κείμενο και αριθμούς με διάφορους τρόπους, ο καθένας από τους οποίους παρουσιάζει τα δικά του πλεονεκτήματα ή μειονεκτήματα. Ο πιο συνηθισμένος τρόπος αποθήκευσης κειμένου είναι υπό την μορφή χαρακτήρων ASCII (Αμερικανικός Πρότυπος Κώδικας για Ανταλλαγή Πληροφοριών) Στο ASCII, κάθε χαρακτήρας αντιπροσωπεύεται από έναν αριθμό, από το 0 μέχρι το 127. Αυτός ο αριθμός ονομάζεται κώδικας ASCII. Όταν ένας υπολογιστής αποθηκεύει ένα κείμενο υπό την μορφή χαρακτήρων ASCII, απλά αποθηκεύει τους κώδικες ASCII για κάθε χαρακτήρα στο κείμενο.

Κωδικοποίηση Unicode

Ένας άλλος τρόπος για να αποθηκεύσετε ένα κείμενο είναι υπό την μορφή χαρακτήρων Unicode. Η κωδικοποίηση Unicode είναι ένα διεθνές πρότυπο που ορίζει έναν μοναδικό αριθμό για κάθε χαρακτήρα σε κάθε γλώσσα. Όταν ένας υπολογιστής αποθηκεύει ένα κείμενο υπό τη μορφή χαρακτήρων Unicode, αποθηκεύει τον κώδικα Unicode για κάθε χαρακτήρα στο κείμενο.

UTF-8

Το UTF-8 είναι μια κωδικοποίηση χαρακτήρων που μπορεί να αποθηκεύσει ένα κείμενο και αριθμούς σε έναν μοναδικό χαρακτήρα Unicode. Αυτή η κωδικοποίηση χρησιμοποιείται ευρέως στο διαδίκτυο επειδή μπορεί να κωδικοποιήσει όλους τους χαρακτήρες σε ένα ευρύ φάσμα γλωσσών. Το UTF-8 είναι μια κωδικοποίηση μεταβλητού μήκους, που σημαίνει ότι μπορεί να κωδικοποιήσει χαρακτήρες διαφορετικών μεγεθών. Η μικρότερη κωδικοποίηση είναι 1 ψηφιολέξη, και η μεγαλύτερη κωδικοποίηση είναι 4 ψηφιολέξεις. Αυτή η κωδικοποίηση είναι αντιστρόφως συμβατή με την κωδικοποίηση ASCII, που σημαίνει ότι το κείμενο ASCII θα κωδικοποιηθεί σε UTF-8 χρησιμοποιώντας 1 ψηφιολέξη.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αριθμοί

Ο πιο συνηθισμένος τρόπος αποθήκευσης αριθμών είναι σε ακέραια δυαδική μορφή. Στο δυαδικό σύστημα, κάθε αριθμός αντιπροσωπεύεται από μια συμβολοσειρά 0s και 1s. Για παράδειγμα, ο αριθμός 12 μπορεί να αναπαρασταθεί ως 01001000. Ο αριθμός 12 μπορεί επίσης να αναπαρασταθεί σε δεκαεξαδικό σύστημα, το οποίο είναι ένα σύστημα αρίθμησης με βάση το 16. Στο δεκαεξαδικό σύστημα, κάθε αριθμός αντιπροσωπεύεται από μια σειρά δεκαεξαδικών ψηφίων. Για παράδειγμα, ο αριθμός 12 μπορεί να αναπαρασταθεί ως C. Όταν ένας υπολογιστής αποθηκεύει έναν αριθμό στο δυαδικό ή δεκαεξαδικό σύστημα, αποθηκεύει την ακέραια τιμή του αριθμού.

Άλλοι τύποι δεδομένων

Οι υπολογιστές αναφέρονται συχνά ως ψηφιακές συσκευές. Ο όρος ψηφιακός μπορεί να χρησιμοποιηθεί για να περιγράψει οτιδήποτε χρησιμοποιεί δυαδικούς αριθμούς. Τα δυαδικά δεδομένα είναι δεδομένα τα οποία αποθηκεύονται σύμφωνα με το δυαδικό σύστημα και μια ψηφιακή συσκευή είναι οποιαδήποτε συσκευή που λειτουργεί με δυαδικά δεδομένα. Στο κεφάλαιο αυτό, έχουμε συζητήσει για το πώς οι αριθμοί και οι χαρακτήρες αποθηκεύονται σε δυαδική μορφή, αλλά οι υπολογιστές λειτουργούν επίσης με πολλούς άλλους τύπους ψηφιακών δεδομένων. Για παράδειγμα, σκεφτείτε τις φωτογραφίες που βγάζετε με την ψηφιακή σας κάμερα. Αυτές οι εικόνες αποτελούνται από μικροσκοπικές κουκκίδες χρώματος γνωστές και ως εικονοστοιχεία. (Ο όρος pixel σημαίνει εικονοστοιχείο.) Κάθε εικονοστοιχείο σε μια εικόνα μετατρέπεται σε έναν αριθμητικό κώδικα που αντιπροσωπεύει το χρώμα του εικονοστοιχείου. Ο αριθμητικός κώδικας αποθηκεύεται στη μνήμη ως δυαδικός αριθμός.

Η μουσική που παίζετε στο CD player σας, iPod ή MP3 player είναι επίσης ψηφιακή. Ένα ψηφιακό τραγούδι αποτελείται από μικρά κομμάτια μουσικής που ονομάζονται δείγματα ή ντέμο. Κάθε δείγμα/ντέμο μετατρέπεται σε δυαδικό αριθμό, ο οποίος μπορεί να αποθηκευτεί στη μνήμη ενός υπολογιστή. Όσο περισσότερα δείγματα/ντέμο έχει ένα τραγούδι, τόσο περισσότερο θα μοιάζει με την πρωτότυπη μουσική που γράφτηκε όταν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

αναπαράγεται. Ένα CD τραγούδι υψηλής ποιότητας έχει περισσότερα από 44.000 δείγματα/ντέμο ανά δευτερόλεπτο!

1.4 Πώς λειτουργεί ένα πρόγραμμα

Υπάρχουν πολλοί διαφορετικοί τύποι προγραμμάτων ηλεκτρονικών υπολογιστών, αλλά όλα παρουσιάζουν τα ίδια βασικά στοιχεία: μια διεπαφή χρήστη, έναν επεξεργαστή και η μνήμη. Η διεπαφή χρήστη επιτρέπει στο χρήστη να εισάγει πληροφορίες και οδηγίες στο πρόγραμμα, ο επεξεργαστής εκτελεί τις οδηγίες και η μνήμη αποθηκεύει το πρόγραμμα και τα δεδομένα που επεξεργάζεται. Τα περισσότερα προγράμματα ηλεκτρονικών υπολογιστών είναι γραμμένα σε μια γλώσσα προγραμματισμού υψηλού επιπέδου, η οποία είναι μια γλώσσα που έχει σχεδιαστεί για να μπορούν να την διαβάζουν και να την γράφουν εύκολα οι άνθρωποι. Ωστόσο, ο επεξεργαστής μπορεί να κατανοήσει μόνο τον κώδικα της μηχανής, ο οποίος είναι μια σειρά από μονάδες και μηδενικά. Έτσι, πριν ένα πρόγραμμα μπορεί να εκκινήσει τη λειτουργία του, πρέπει να μετατραπεί σε κώδικα μηχανής. Αυτό γίνεται από ένα πρόγραμμα που ονομάζεται μεταγλωττιστής ή μεταφραστής. Ο μεταγλωττιστής διαβάζει το πρόγραμμα και το μετατρέπει σε κώδικα μηχανής. Στη συνέχεια αποθηκεύει τον κώδικα του μηχανήματος σε ένα αρχείο που ονομάζεται εκτελέσιμο. Όταν ο χρήστης εκτελεί/τρέχει το πρόγραμμα, το εκτελέσιμο φορτώνεται στη μνήμη και ο επεξεργαστής εκτελεί τις εντολές.

Ο κύκλος προσκόμισης – αποκωδικοποίησης – εκτέλεσης

Ο κύκλος προσκόμισης - αποκωδικοποίησης - εκτέλεσης είναι μια βασική διαδικασία που χρησιμοποιεί ένας ηλεκτρονικός υπολογιστής για την εκτέλεση εντολών. Ο κύκλος ξεκινά όταν ο υπολογιστής λαμβάνει μια εντολή από τη μνήμη. Στη συνέχεια, αποκωδικοποιεί την οδηγία για να προσδιορίσει αυτό που πρέπει να εκτελέσει. Τέλος, εκτελεί τις οδηγίες. Ο κύκλος στη συνέχεια επαναλαμβάνεται, λαμβάνοντας την επόμενη οδηγία από τη μνήμη.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Από τη γλώσσα της μηχανής στη συμβολική γλώσσα

Καθώς ο προγραμματισμός στη γλώσσα μηχανής, που αποτελείται μόνο από δυαδικό κώδικα, είναι πολύ περίπλοκος για τον άνθρωπο, υπάρχει επίσης και η συμβολική γλώσσα ή γλώσσα assembly. Η συμβολική γλώσσα είναι μια γλώσσα προγραμματισμού χαμηλού επιπέδου για έναν υπολογιστή, μικροεπεξεργαστή ή άλλη προγραμματιζόμενη συσκευή, στην οποία ο προγραμματιστής χρησιμοποιεί εντολές στη συμβολική γλώσσα για τον έλεγχο της λειτουργίας της συσκευής. Η συμβολική γλώσσα πρέπει να είναι συγκεκριμένη για να ανταποκρίνεται σε ένα συγκεκριμένο μικροεπεξεργαστή ή οικογένεια μικροεπεξεργαστών. Αποτελείται από μια σειρά μνημονικών κωδικών, συμβολικών ονομάτων για τις λειτουργίες που μπορεί να εκτελέσει ο μικροεπεξεργαστής, και τους χειριστές (δεδομένα) πάνω στους οποίους πρόκειται να εκτελεστούν αυτές οι λειτουργίες. Η συμβολική γλώσσα μετατρέπεται σε κώδικα μηχανής, μια μορφή δυαδικού κώδικα που είναι συγκεκριμένη για έναν συγκεκριμένο τύπο υπολογιστή και μπορεί να γίνει κατανοητή από τον επεξεργαστή του υπολογιστή. Ο κώδικας μηχανής είναι η μόνη μορφή κώδικα που μπορεί να εκτελέσει άμεσα ο επεξεργαστής.

Ο προγραμματισμός της συμβολικής γλώσσας είναι ευκολότερος από τον προγραμματισμό της γλώσσας μηχανής γιατί ο τελευταίος δεν είναι αρκετά ταχύς και αποτελεσματικός ως προς την ανάγνωση ή την παραγωγή ενός πηγαίου κωδικού. Γι' αυτό λοιπόν το λόγο, δημιουργήθηκαν υψηλού επιπέδου γλώσσες προγραμματισμού (όπως C# ή python).

Σήμερα υπάρχουν πολλές γλώσσες προγραμματισμού υψηλού επιπέδου. Μερικά από τα πιο κοινά είναι Java, Python, και Ruby. Οι γλώσσες προγραμματισμού υψηλού επιπέδου είναι πιο εύχρηστες από τις γλώσσες προγραμματισμού χαμηλού επιπέδου. Αυτές οι γλώσσες επιτρέπουν στο χρήση να επικεντρωθεί μόνο στην βασική αποστολή του και όχι στις λεπτομέρειες του υπολογιστή. Αυτό τις καθιστά ιδανικές για τη δημιουργία εφαρμογών και προγραμμάτων. Οι γλώσσες προγραμματισμού υψηλού

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

επιπέδου τείνουν επίσης να είναι πιο «επιεικείς» σε σύγκριση με τις γλώσσες προγραμματισμού χαμηλού επιπέδου. Εάν κάνετε ένα λάθος κατά την εγγραφή ενός κώδικα σε μια γλώσσα υψηλού επιπέδου, ο μεταγλωττιστής θα είναι συνήθως σε θέση να το διορθώσει για σας. Αυτό θα σας βοηθήσετε να εξοικονομήσετε σημαντικό χρόνο ή ταλαιπωρία κατά την κωδικοποίηση.

Λέξεις-κλειδιά, Χειριστές/Εκτελεστές, και Σύνταξη: μια επισκόπηση

Σήμερα υπάρχουν πολλές γλώσσες προγραμματισμού υψηλού επιπέδου. Η καθεμιά διαθέτει το δικό της μοναδικό σύνολο λέξεων-κλειδιών, εκτελεστών και σύνταξης. Για να είστε αποτελεσματικοί με μια γλώσσα προγραμματισμού υψηλού επιπέδου, είναι σημαντικό να είστε εξοικειωμένοι με τις συγκεκριμένες λέξεις-κλειδιά, τους χειριστές/εκτελεστές και τη σύνταξη που χρησιμοποιείται από την εκάστοτε γλώσσα ξεχωριστά. Μερικές από τις πιο κοινές λέξεις-κλειδιά που χρησιμοποιούνται στις γλώσσες προγραμματισμού υψηλού επιπέδου περιλαμβάνουν: *αν, στη συνέχεια, διαφορετικά, ενώ, για, κάνε, σπάσε, συνέχισε*. Αυτές οι λέξεις-κλειδιά χρησιμοποιούνται για τον έλεγχο της ροής της εκτέλεσης του προγράμματος. Οι χειριστές/εκτελεστές είναι σύμβολα που αντιπροσωπεύουν λειτουργίες που μπορούν να εκτελεστούν σε τιμές. Οι συνηθέστεροι χειριστές/εκτελεστές περιλαμβάνουν: + *(προσθήκη)*, - *(αφαίρεση)*, * *(πολλαπλασιασμός)*, / *(διαίρεση)* και % *(ποσοστό)*. Αυτοί οι εκτελεστές μπορούν να χρησιμοποιηθούν για τον υπολογισμό των αποτελεσμάτων των εκφράσεων. Η σύνταξη μιας γλώσσας προγραμματισμού είναι το σύνολο των κανόνων που διέπουν τον τρόπο με τον οποίο πρέπει να γραφτεί ο κώδικας για να ερμηνευτεί από τον μεταγλωττιστή ή τον διερμηνέα. Η σύνταξη μιας γλώσσας προγραμματισμού υψηλού επιπέδου είναι συνήθως πιο επιεικής από τη σύνταξη μιας γλώσσας χαμηλότερου επιπέδου. Αυτό μπορεί να διευκολύνει τους αρχάριους να μάθουν να προγραμματίζουν.

Μεταγλωττιστές και διερμηνείς

Οι μεταγλωττιστές και οι διερμηνείς υπολογιστών είναι σημαντικά εργαλεία για τους προγραμματιστές λογισμικού. Ένας μεταγλωττιστής λαμβάνει έναν κώδικα γραμμένο σε μια γλώσσα και τον μετατρέπει σε κώδικα που μπορεί να εκτελεστεί σε μια διαφορετική

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

μηχανή. Ένας διερμηνέας λαμβάνει έναν κώδικα γραμμένο σε μία γλώσσα και τον εκτελεί όπως είναι, χωρίς να τον συντάξει πρώτα. Οι μεταγλωττιστές χρησιμοποιούνται συνήθως για γλώσσες που έχουν μεγάλη δομή, όπως το C ή το Java. Οι διερμηνείς χρησιμοποιούνται συνήθως για γλώσσες που είναι πιο ευέλικτες, όπως το Python ή το Ruby. Οι μεταγλωττιστές συνήθως παράγουν ταχύτερο κώδικα από τους διερμηνείς. Ωστόσο, οι διερμηνείς είναι συνήθως πιο φορητοί, πράγμα που σημαίνει ότι μπορούν να τρέξουν σε περισσότερους τύπους μηχανών. Η επιλογή μεταξύ των δύο εξαρτάται πάντα από την περίσταση/ περίπτωση. Εάν θεωρείται ότι είναι σημαντικότερη για τις ανάγκες σας, τότε ένας μεταγλωττιστής είναι καταλληλότερος. Εάν η φορητότητα σας είναι πιο σημαντική, τότε ο διερμηνέας είναι καλύτερη επιλογή.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

1.5. Προγράμματα και γλώσσες προγραμματισμού

Οι γλώσσες προγραμματισμού χρησιμεύουν στη δημιουργία προγραμμάτων, τα οποία χρησιμοποιούνται για τον έλεγχο της συμπεριφοράς μιας μηχανής, συνήθως ενός υπολογιστή. Μια γλώσσα προγραμματισμού παρέχει τη δομή στον προγραμματιστή για να δώσει τις εντολές στη μηχανή αλλά και για επικοινωνήσει αυτές τις εντολές σε άλλους προγραμματιστές. Υπάρχουν πολλές γλώσσες προγραμματισμού που χρησιμοποιούνται σήμερα. Οι πιο δημοφιλείς είναι οι C, Java, Python και JavaScript.

Τύποι γλωσσών προγραμματισμού

Υπάρχουν δεκάδες γλώσσες προγραμματισμού που χρησιμοποιούνται σήμερα, αλλά μπορούν να ταξινομηθούν ευρέως σε πέντε βασικές κατηγορίες:

- **Γλώσσες προγραμματισμού χαμηλού επιπέδου:** Αυτές οι γλώσσες προγραμματισμού είναι πολύ κοντά στην γλώσσα μηχανής και χρησιμοποιούνται για τον προγραμματισμό μικροεπεξεργαστών και άλλων συσκευών χαμηλού επιπέδου. Δεν είναι εύληπτες και γι’ αυτό δεν είναι τόσο δημοφιλείς όσον αφορά τον προγραμματισμό για γενικούς σκοπούς. Παραδείγματα: Συμβολική γλώσσα, γλώσσα προγραμματισμού C και συμβολική γλώσσα χαμηλού επιπέδου.
- **Γλώσσες προγραμματισμού υψηλού επιπέδου:** Αυτές οι γλώσσες προγραμματισμού έχουν σχεδιαστεί για να είναι εύληπτες και εύχρηστες. Είναι δημοφιλείς για τον προγραμματισμό για γενικούς σκοπούς. Παραδείγματα: Java, C++ και Python.
- **Γλώσσες σεναρίου:** Οι γλώσσες σεναρίου έχουν σχεδιαστεί για να είναι εύκολες στη χρήση και είναι δημοφιλείς για σκοπούς σεναρίου. Παραδείγματα: Python, Ruby και JavaScript.
- **Γλώσσες ανά τομέα:** Οι γλώσσες ανά τομέα έχουν σχεδιαστεί για μια συγκεκριμένη εργασία ή κλάδο. Δεν είναι εύληπτες και γι’ αυτό δεν είναι τόσο δημοφιλείς όσον αφορά τον προγραμματισμό για γενικούς σκοπούς. Παραδείγματα: MATLAB, SQL και FORTRAN.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- **Αντικειμενοστραφείς γλώσσες προγραμματισμού:** Αυτές οι γλώσσες προγραμματισμού βασίζονται σε ένα πρότυπο αντικειμενοστραφούς προγραμματισμού. Παραδείγματα: Java, C++ και Python.

Από ένα πρόγραμμα υψηλού επιπέδου σε ένα εκτελέσιμο αρχείο

Όταν ένα πρόγραμμα υπολογιστή είναι γραμμένο σε μια γλώσσα υψηλού επιπέδου, μεταφράζεται πρώτα σε μια γλώσσα χαμηλότερου επιπέδου, η οποία μπορεί να γίνει πιο εύκολα κατανοητή από τις μηχανές. Η γλώσσα χαμηλότερου επιπέδου μεταγλωττίζεται στη συνέχεια σε ένα εκτελέσιμο αρχείο, το οποίο μπορεί να εκτελεστεί σε έναν υπολογιστή.

IDEs (Ολοκληρωμένο Περιβάλλον Ανάπτυξης)

Ένα Ολοκληρωμένο Περιβάλλον Ανάπτυξης («Integrated Development Environment», IDE) είναι μία εφαρμογή λογισμικού η οποία παρέχει εκτεταμένες διευκολύνσεις στους προγραμματιστές ηλεκτρονικών υπολογιστών για την ανάπτυξη λογισμικών. Ένα IDE αποτελείται συνήθως από έναν επεξεργαστή πηγαίου κώδικα, εργαλεία αυτοματισμού κατασκευής και έναν αποσφαλματωτή. Ο επεξεργαστής πηγαίου κώδικα επιτρέπει στον προγραμματιστή να γράψει κώδικα, ενώ τα εργαλεία αυτοματοποίησης κατασκευής αυτοματοποιούν τη διαδικασία σύνταξης αυτού του κώδικα σε μια μορφή που μπορεί να τρέξει ο υπολογιστής. Ο αποσφαλματωτής επιτρέπει στον προγραμματιστή να περάσει μέσα από τον κώδικα, εξετάζοντας την κατάσταση του προγράμματος σε κάθε σημείο της εκτέλεσής του. Τα IDE χρησιμοποιούνται συχνά σε συνδυασμό με ένα σύστημα ελέγχου έκδοσης, το οποίο επιτρέπει σε διαφορετικούς προγραμματιστές που εργάζονται πάνω στην ίδια εργασία να μοιράζονται και να συγχωνεύουν τις αλλαγές τους απρόσκοπτα.

Τα κοινά στοιχεία στις γλώσσες προγραμματισμού

Οι γλώσσες προγραμματισμού υπολογιστών παρουσιάζουν ορισμένα κοινά στοιχεία, παρά τις διαφορές τους. Όλες οι γλώσσες προγραμματισμού έχουν έναν τρόπο να αντιπροσωπεύουν τις εντολές στον υπολογιστή σε μια μορφή που ο υπολογιστής

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

μπορεί να καταλάβει. Αυτό συνήθως ονομάζεται κώδικας, ή πηγαίος κώδικας. Οι προγραμματιστές χρησιμοποιούν κώδικες για να δημιουργήσουν προγράμματα και εφαρμογές λογισμικού. Όλες οι γλώσσες προγραμματισμού έχουν επίσης έναν τρόπο οργάνωσης των εντολών έτσι ώστε να μπορούν να επαναχρησιμοποιηθούν, να τροποποιηθούν ή να μοιραστούν με άλλους προγραμματιστές. Αυτό συνήθως ονομάζεται βιβλιοθήκη ή άρθρωμα. Οι βιβλιοθήκες και τα αρθρώματα επιτρέπουν στους προγραμματιστές να δημιουργούν σύνθετα προγράμματα με βάση το έργο άλλων προγραμματιστών. Τέλος, όλες οι γλώσσες προγραμματισμού έχουν έναν τρόπο να μεταφέρουν πληροφορίες στον χρήστη σχετικά με το τι κάνει το πρόγραμμα και πώς λειτουργεί. Αυτό συνήθως ονομάζεται έξοδος ή αποσφαλμάτωση πληροφοριών. Οι πληροφορίες εξόδου και αποσφαλμάτωσης βοηθούν τους προγραμματιστές να κατανοήσουν και να διορθώσουν τα προβλήματα που παρουσιάζουν τα προγράμματά τους.

Διαδικαστικός και αντικειμενοστραφής προγραμματισμός

Υπάρχουν δύο κύριοι τύποι προγραμματισμού: ο διαδικαστικός και ο αντικειμενοστραφής. Ο διαδικαστικός προγραμματισμός περιλαμβάνει μια διαδικασία βήμα προς βήμα, ενώ ο αντικειμενοστραφής προγραμματισμός περιλαμβάνει τη δημιουργία αντικειμένων που αλληλεπιδρούν μεταξύ τους. Ο διαδικαστικός προγραμματισμός θεωρείται συχνά απλούστερος από τον αντικειμενοστραφή προγραμματισμό. Είναι εύκολο να μάθετε τα βήματα που απαιτούνται για να ολοκληρώσετε μια εργασία και είναι εύκολο να αλλάξετε τη σειρά αυτών των βημάτων χωρίς να επηρεάσετε το αποτέλεσμα. Ωστόσο, ο διαδικαστικός προγραμματισμός μπορεί να είναι λιγότερο αποτελεσματικός επειδή μπορεί να είναι δύσκολη η επαναχρησιμοποίηση κώδικα που έχει συνταχθεί για μια συγκεκριμένη εργασία. Ο αντικειμενοστραφής προγραμματισμός είναι πιο περίπλοκος από τον διαδικαστικό προγραμματισμό, αλλά επιτρέπει μεγαλύτερη ευελιξία και επαναχρησιμοποίηση του κώδικα. Τα αντικείμενα μπορούν να δημιουργηθούν για συγκεκριμένες εργασίες και στη συνέχεια να επαναχρησιμοποιηθούν όπως απαιτείται. Επιπλέον, ο αντικειμενοστραφής κώδικας είναι συχνά ευκολότερο να διαβαστεί και να κατανοηθεί από ό, τι ο διαδικαστικός κώδικας. Ωστόσο, ο αντικειμενοστραφής προγραμματισμός μπορεί να

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

είναι πιο δύσκολο να διδαχθεί και μπορεί να είναι λιγότερο αποτελεσματικός από τον διαδικαστικό προγραμματισμό.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

2. HTML – HyperText Markup Language

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πληροφορίες για την ενότητα

Ενότητα:

2. HTML

Προαπαιτούμενες γνώσεις:

Βασικές γνώσεις υπολογιστών, εγκατεστημένο βασικό λογισμικό και βασικές γνώσεις εργασίας με αρχεία.

Φόρτος εργασίας:

10 ώρες.

Περιγραφή:

Σε αυτή την ενότητα, θα καλύψουμε τα βασικά της γλώσσας λογισμικού HTML, για να μάθουν οι μαθητές για τον κόσμο του προγραμματισμού, αφού αποκτήσουν γνώση για κάποιες βασικές έννοιες. Θα ορίσουμε τα στοιχεία, τα χαρακτηριστικά και όλους τους άλλους σημαντικούς όρους που μπορεί να έχουν ακούσει, καθώς και πού ταιριάζουν αυτοί οι όροι στη συγκεκριμένη γλώσσα. Θα δείξουμε επίσης πώς είναι δομημένο ένα στοιχείο HTML, πώς είναι δομημένη μια τυπική σελίδα σε HTML και εξηγούμε άλλα σημαντικά βασικά χαρακτηριστικά της γλώσσας.

Μαθησιακά αποτελέσματα:

- Αναγνώριση της έννοιας της HyperText Markup Language (HTML) στην οικογένεια των γλωσσών περιγραφής εγγράφων.
- Διάκριση της δομής, του περιεχομένου και των στυλ μιας σελίδας.
- Χρήση της HTML στη δημιουργία ιστοσελίδων.

Απαιτούμενα υλικά:

- Υπολογιστής ή φορητός υπολογιστής
- Σύνδεση στο Διαδίκτυο
- Εργαλείο δημιουργίας ιστότοπου στο διαδίκτυο (https://sites.google.com/new)
- Διαδικτυακό πρόγραμμα επεξεργασίας κειμένου (https://www.w3schools.com/html/default.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σενάριο μαθήματος:

Ο συνολικός χρόνος που απαιτείται για αυτή την ενότητα είναι 10 ώρες και εναπόκειται στον εκπαιδευτή να αποφασίσει πόσο χρόνο θα αφιερώσει στη διδασκαλία κάθε υποενότητας. Προκειμένου να αξιοποιήσουμε στο έπακρο όλο τον διαθέσιμο χρόνο, προτείνουμε τη χρήση του εκπαιδευτικού υλικού που παρήγαγαν οι εταίροι του έργου (παρουσιάσεις PowerPoint), το οποίο σχεδιάστηκε με γνώμονα την αποτελεσματική χρήση του χρόνου. Αυτές οι παρουσιάσεις αποτελούνται από τα ακόλουθα στοιχεία:

- Ανάπτυξη των υποενοτήτων και των βασικών ιδεών που πρέπει να κρατήσουν οι εκπαιδευόμενοι.
- Προτεινόμενες Δραστηριότητες/Ασκήσεις.

Επομένως, εάν ο εκπαιδευτής ακολουθήσει τη λογική σειρά των παρουσιάσεων PowerPoint, σίγουρα θα μπορέσει να ολοκληρώσει τα μαθήματα εντός του καθορισμένου χρονικού ορίου. Αυτές οι παρουσιάσεις μπορούν επίσης να διατεθούν στους εκπαιδευόμενους για ατομική μελέτη.

Υποενότητες:

2.1. Τα βασικά της HTML

2.2. Προηγμένες έννοιες της HTML

2.3. Χαρακτηριστικά της HTML5

2.4. Αναφορές στην HTML5

Επιπλέον πόροι:

- Οδηγός αναφοράς HTML
- W3Schools - Οδηγός για κάθε στοιχείο HTML και κανόνα CSS, και παραδείγματα για κάθε ένα από αυτά
- Khan Academy: χρήσιμοι πόροι και βίντεο για την κωδικοποίηση HTML & CSS, σε πολλές διαφορετικές γλώσσες

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

2.1. Τα βασικά της HTML

Η HTML (Hypertext Markup Language) **δεν είναι** γλώσσα προγραμματισμού. Είναι μια γλώσσα σήμανσης που επικοινωνεί με τα προγράμματα περιήγησης ιστού για το πώς να δομήσουν τις ιστοσελίδες που επισκέπτονται οι χρήστες. Αυτό μπορεί να είναι τόσο περίπλοκο ή όσο απλό θέλει ο προγραμματιστής ιστού. Η HTML περιλαμβάνει μια σειρά στοιχείων, τα οποία χρησιμοποιεί ο προγραμματιστής για να περικλείσει, να αναδιπλώσει ή να αυξήσει διάφορα μέρη του περιεχομένου για να εμφανιστεί ή να ενεργήσει με συγκεκριμένο τρόπο. Οι ετικέτες που εσωκλείουν μπορούν να μετατρέψουν το περιεχόμενο σε υπερσύνδεσμο για σύνδεση σε άλλη σελίδα, πλάγια γραφή λέξεων και ούτω καθεξής. Για παράδειγμα, λαμβάνοντας υπόψη την ακόλουθη γραμμή κειμένου:

```
HTML is cool.
```

***Εικόνα 1** - Γραμμή κειμένου «Η HTML είναι κουλ» (Πηγή: Συντάκτης)*

Εάν κάποιος θέλει να κάνει το κείμενό του να λειτουργεί από μόνο του, μπορεί να προσδιορίσει ότι είναι μια παράγραφος, περικλείοντάς το σε ένα στοιχείο παραγράφου (<p>):

```
<p> HTML is cool.</p>
```

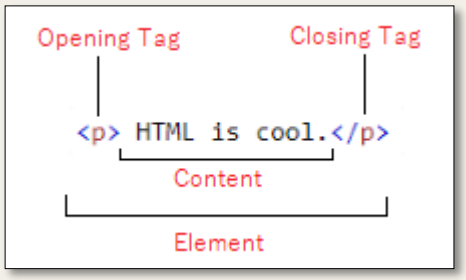
***Εικόνα 2** – Κωδικοποίηση για την παράγραφο «Η HTML είναι κουλ» (Πηγή: Συντάκτης)*

Σημείωση: Οι ετικέτες HTML δεν κάνουν διάκριση πεζών-κεφαλαίων. Αυτό σημαίνει ότι μπορούν να γραφτούν με κεφαλαία ή πεζά γράμματα. Για παράδειγμα, μια <title> ετικέτα <title> θα μπορούσε να γραφτεί ως<title> <TITLE> <Title>,<TiTlE>, , , κ.λπ., και να λειτουργήσει. Ωστόσο, η καλύτερη πρακτική είναι να γράφετε όλες τις ετικέτες με πεζά γράμματα για συνέπεια και αναγνωσιμότητα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η ανατομία ενός στοιχείου HTML

Ας εξερευνήσουμε περαιτέρω το στοιχείο της παραγράφου μας από την προηγούμενη ενότητα:



Εικόνα 3 *– Ανατομία ενός στοιχείου HTML (Πηγή: Συγγραφέας)*

Επομένως, η ανατομία ενός στοιχείου της HTML αποτελείται από:

Την **ετικέτα αρχής**: το όνομα του στοιχείου (σε αυτό το παράδειγμα είναι το p για paragraph-παράγραφο), μέσα σε γωνιακές παρενθέσεις. Αυτή η ετικέτα αρχής σηματοδοτεί το σημείο που αρχίζει ή τίθεται σε ισχύ το στοιχείο. Σε αυτό το παράδειγμα, έρχεται πρώτο, στην αρχή του κειμένου της παραγράφου.

Το **περιεχόμενο**: Αυτό είναι το περιεχόμενο του στοιχείου. Σε αυτό το παράδειγμα, είναι το κείμενο της παραγράφου.

Την **ετικέτα τέλους**: Είναι ίδια με την ετικέτα αρχής, με τη διαφορά ότι περιλαμβάνει μια κάθετο προς τα εμπρός πριν από το όνομα του στοιχείου. Αυτό σηματοδοτεί πού τελειώνει το στοιχείο. Η απουσία συμπερίληψης μιας ετικέτας τέλους είναι ένα συνηθισμένο σφάλμα αρχαρίων που μπορεί να οδηγήσει σε περίεργα αποτελέσματα.

Το **στοιχείο** είναι η ετικέτα αρχής, ακολουθούμενη από το περιεχόμενο, ακολουθούμενο από την ετικέτα τέλους.

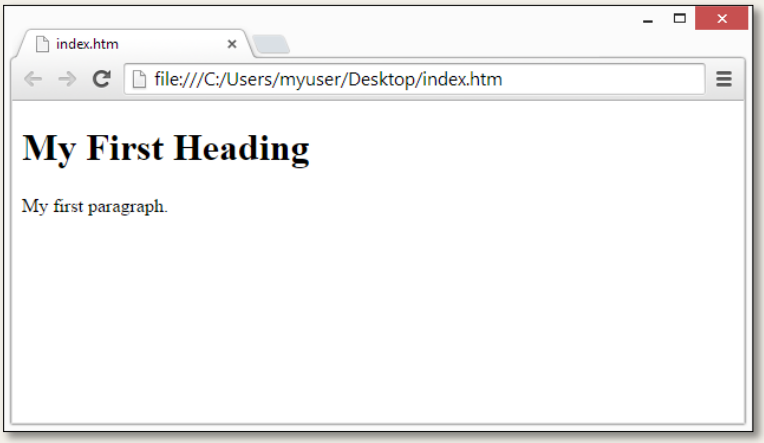
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σημείωση: Ορισμένα στοιχεία HTML δεν έχουν περιεχόμενο (όπως το στοιχείο <br>). Αυτά τα στοιχεία ονομάζονται «κενά στοιχεία». Δεν έχουν ετικέτα τέλους!

Προγράμματα Περιήγησης

Ο σκοπός ενός προγράμματος περιήγησης (Chrome, Edge, Firefox, Safari) είναι να διαβάζει έγγραφα HTML και να τα εμφανίζει σωστά.

Ένα πρόγραμμα περιήγησης δεν εμφανίζει τις ετικέτες HTML. Τις χρησιμοποιεί μόνο για να καθορίσει τον τρόπο εμφάνισης του εγγράφου:

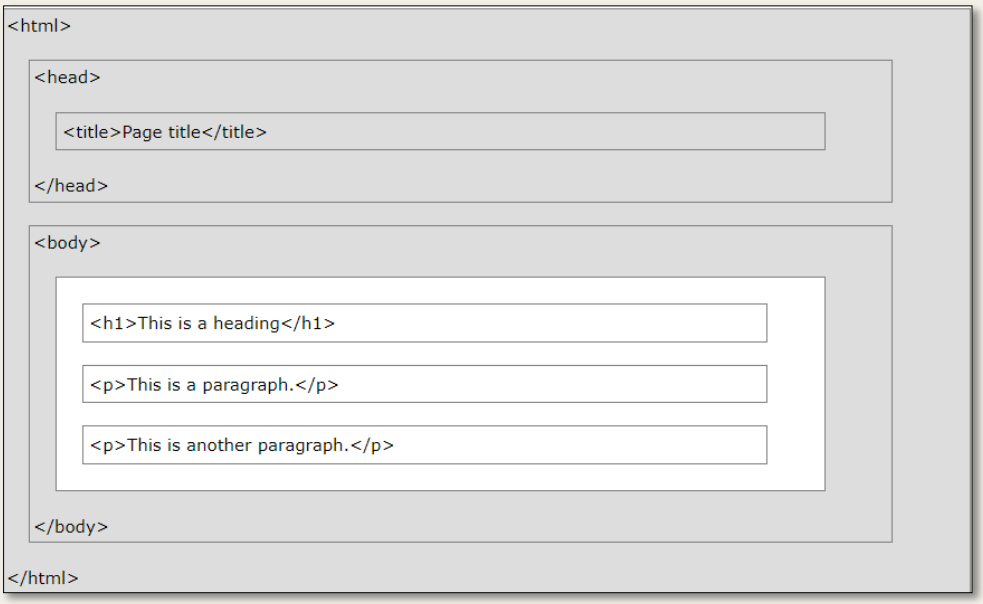


Εικόνα 4 *– Ένα έγγραφο HTML που προσδιορίζεται σε ένα πρόγραμμα περιήγησης ιστού (Πηγή: https://www.w3schools.com/html/html_intro.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Δομή μιας σελίδας HTML

Μια σελίδα HTML πρέπει να είναι δομημένη ως εξής:



***Εικόνα 5** – Δομή μιας σελίδας HTML (Πηγή: https://www.w3schools.com/html/html_intro.asp)*

Το περιεχόμενο μέσα στην ενότητα <body> (η λευκή περιοχή παραπάνω) θα εμφανίζεται σε ένα πρόγραμμα περιήγησης. Το περιεχόμενο μέσα στο στοιχείο <title> θα εμφανίζεται στη γραμμή τίτλου του προγράμματος περιήγησης ή στην καρτέλα της σελίδας.

Ιστορικό της HTML

Από τη δημιουργία του Παγκόσμιου Ιστού (World Wide Web), υπήρξαν πολλές εκδόσεις της HTML:

Έτος	**Έκδοση**
1989	Ο Τιμ Μπέρνερς Λι επινόησε το www
1991	Ο Τιμ Μπέρνερς Λι επινόησε την HTML
1993	Ο Ντέιβ Ράγκετ συνέταξε την HTML+

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

1995	Η HTML Working Group καθόρισε την HTML 2.0
1997	Σύσταση του W3C: HTML 3.2
1999	Σύσταση του W3C: HTML 4.01
2000	Σύσταση του W3C: XHTML 1.0
2008	Πρώτο δημοσιευμένο προσχέδιο WHATWG HTML5
2012	Καθιερωμένο πρότυπο WHATWG HTML5
2014	Σύσταση του W3C: HTML5
2016	Υποψήφια Σύσταση του W3C: HTML 5.1
2017	Σύσταση του W3C: HTML5.1 Δεύτερη έκδοση
2017	Σύσταση του W3C: HTML5.2

Πίνακας 1 *– ιστορία της εξέλιξης της HTML (Πηγή: https://www.w3schools.com/html/html_intro.asp)*

Αυτός ο οδηγός ακολουθεί το πιο πρόσφατο πρότυπο της HTML5.

Επεξεργαστές HTML

Ένα απλό πρόγραμμα επεξεργασίας κειμένου είναι το μόνο που χρειάζεστε για να μάθετε την HTML.

Μάθετε την HTML χρησιμοποιώντας το Notepad ή το TextEdit

Οι ιστοσελίδες μπορούν να δημιουργηθούν και να τροποποιηθούν χρησιμοποιώντας επαγγελματικούς επεξεργαστές HTML.

Ωστόσο, για την εκμάθηση της HTML συνιστάται ένας απλός επεξεργαστής κειμένου όπως το Notepad (PC) ή το TextEdit (Mac). Η χρήση ενός απλού επεξεργαστή κειμένου μπορεί να είναι ένας καλός τρόπος για να μάθετε την HTML

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Θα πρέπει να ακολουθήσετε τα παρακάτω βήματα για να δημιουργήσετε την πρώτη ιστοσελίδα με τους μαθητές σας με το Notepad ή το TextEdit.

Βήμα 1: Ανοίξτε το Notepad (στον υπολογιστή)

(Windows 8 ή αργότερη έκδοση: Ανοίξτε την οθόνη έναρξης (το σύμβολο του παραθύρου κάτω αριστερά στην οθόνη σας). Πληκτρολογήστε Notepad.

Windows 7 ή παλαιότερη έκδοση: Ανοίξτε Έναρξη > Προγράμματα > Αξεσουάρ > Notepad)

- Ανοίξτε το πρόγραμμα TextEdit (Mac)
- Στην Αναζήτηση > Εφαρμογές > TextEdit
- Αποκτήστε την εφαρμογή για να αποθηκεύσετε σωστά τα αρχεία. Στις Προτιμήσεις > Μορφή > επιλέξτε «Απλό κείμενο»
- Στη συνέχεια, στην ενότητα «Άνοιγμα και αποθήκευση», επιλέξτε το πλαίσιο που λέει «Εμφάνιση αρχείων HTML ως κώδικα HTML αντί για μορφοποιημένο κείμενο» *(Display HTML files as HTML code instead of formatted text)*.
- Στη συνέχεια**, ανοίξτε ένα νέο έγγραφο** για να γράψετε τον κώδικα.

Βήμα 2: Γράψτε μερικά πράγματα στην HTML

- Γράψτε ή αντιγράψτε τον ακόλουθο κώδικα HTML στο Notepad:

```
<!DOCTYPE html>
<html>
<body>
<h1> My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

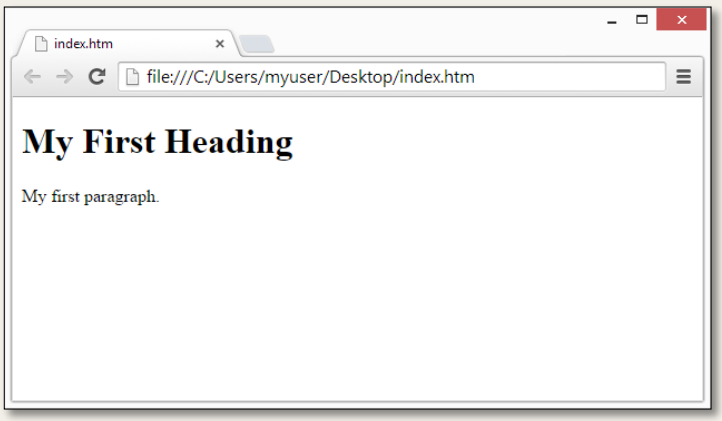
Βήμα 3: Αποθηκεύστε τη σελίδα HTML

- Αποθηκεύστε το αρχείο στον υπολογιστή σας.
- Επιλέξτε Αρχείο > Αποθήκευση ως από το μενού στο Notepad.
- Ονομάστε το αρχείο "index.htm" και ορίστε την κωδικοποίηση σε UTF-8 (που είναι η προτιμώμενη κωδικοποίηση για αρχεία HTML).

Βήμα 4: Προβάλετε τη σελίδα HTML στο πρόγραμμα περιήγησής σας

- Ανοίξτε το αποθηκευμένο αρχείο HTML στο αγαπημένο σας πρόγραμμα περιήγησης (κάντε διπλό κλικ στο αρχείο ή κάντε δεξί κλικ - και επιλέξτε «Άνοιγμα με»).

Το αποτέλεσμα θα είναι παρόμοιο με το εξής:



Εικόνα 6 *– Ένα έγγραφο HTML που προσδιορίζεται σε ένα πρόγραμμα περιήγησης ιστού (Πηγή: https://www.w3schools.com/html/html_intro.asp)*

Βασικά Παραδείγματα HTML

Σε αυτήν την ενότητα, θα παρουσιαστούν μερικά βασικά παραδείγματα HTML.

Έγγραφα HTML

Όλα τα έγγραφα HTML πρέπει να ξεκινούν με μια εντολή τύπου εγγράφου: <!DOCTYPE html>.

Η εντολή <!DOCTYPE> αντιπροσωπεύει τον τύπο του εγγράφου και βοηθά τα προγράμματα περιήγησης να εμφανίζουν σωστά τις ιστοσελίδες. Πρέπει να εμφανίζεται

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

μόνο μία φορά, στο επάνω μέρος της σελίδας (πριν από τυχόν ετικέτες HTML). Δεν υπάρχει διάκριση πεζών και κεφαλαίων χαρακτήρων.

Η δήλωση <!DOCTYPE> για HTML5 είναι: <!DOCTYPE html>

Επικεφαλίδες HTML

Οι επικεφαλίδες HTML ορίζονται με τις ετικέτες <h1> με <h6>.

Η ετικέτα <h1> ορίζει την πιο σημαντική επικεφαλίδα. Η ετικέτα <h6> ορίζει τη λιγότερο σημαντική επικεφαλίδα:

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
```

***Εικόνα 7** – Επικεφαλίδες HTML (Πηγή: https://www.w3schools.com/html)*

Παράγραφοι HTML

Οι παράγραφοι HTML ορίζονται με την ετικέτα <p>:

```
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
```

***Εικόνα 8** – παράγραφοι HTML (Πηγή: https://www.w3schools.com/html)*

Σύνδεσμοι HTML

Οι σύνδεσμοι HTML ορίζονται με την ετικέτα <a>:

```
<a href="https://www.w3schools.com">This is a link</a>
```

***Εικόνα 9** – Σύνδεσμος HTML (Πηγή: https://www.w3schools.com/html)*

Ο προορισμός του συνδέσμου καθορίζεται στο χαρακτηριστικό href.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τα χαρακτηριστικά χρησιμοποιούνται για την παροχή πρόσθετων πληροφοριών σχετικών με τα στοιχεία HTML.

Περισσότερα για τα χαρακτηριστικά θα αναφερθούν αργότερα.

Εικόνες HTML

Οι εικόνες HTML ορίζονται με την ετικέτα <img>.

Το αρχείο προέλευσης (src), το εναλλακτικό κείμενο (alt), το πλάτος και το ύψος παρέχονται ως χαρακτηριστικά:

```
<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">
```

Εικόνα 10 *– Εικόνες HTML (Πηγή: https://www.w3schools.com/html)*

Πώς να προβάλετε μια πηγή HTML;

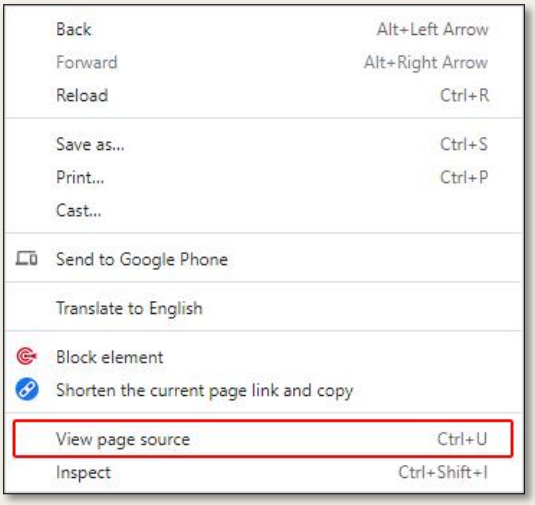
Προβολή πηγαίου κώδικα HTML:

- Κάντε δεξί κλικ σε μια σελίδα HTML και επιλέξτε «Προβολή προέλευσης σελίδας» (στο Chrome) ή «Προβολή προέλευσης» (στο Edge) ή παρόμοια σε άλλα προγράμματα περιήγησης. Αυτό θα ανοίξει ένα παράθυρο που περιέχει τον πηγαίο κώδικα HTML της ιστοσελίδας.

Επιθεώρηση στοιχείου HTML:

- Κάντε δεξί κλικ σε ένα στοιχείο (ή σε μια κενή περιοχή) και επιλέξτε «Επιθεώρηση» ή «Επιθεώρηση στοιχείου» για να δείτε από ποια στοιχεία αποτελούνται (θα δείτε τόσο στην HTML όσο και στην CSS). Μπορείτε επίσης να επεξεργαστείτε σε HTML ή CSS γρήγορα στο πλαίσιο Στοιχεία ή Στυλ που εμφανίζεται.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



***Εικόνα 11** – Τρόπος προβολής της πηγής σελίδας (Πηγή:Συγγραφέας)*

Η δομή ενός εγγράφου HTML

Το ίδιο το έγγραφο HTML αρχίζει με <html> και τελειώνει με </html>. Το ορατό τμήμα του εγγράφου HTML βρίσκεται μεταξύ <body> και</body>.

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

***Εικόνα 12** – Εντολή τύπου εγγράφου, HTML και ορατό τμήμα του εγγράφου HTML (Πηγή: https://www.w3schools.com/html)*

Χαρακτηριστικά HTML

Τα χαρακτηριστικά HTML παρέχουν πρόσθετες πληροφορίες σχετικά με τα στοιχεία HTML και όλα τα στοιχεία HTML μπορούν να τα έχουν. Τα χαρακτηριστικά παρέχουν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

πρόσθετες πληροφορίες σχετικά με στοιχεία, τα οποία προσδιορίζονται πάντα στην ετικέτα αρχής. Συνήθως έρχονται σε ζεύγη ονόματος/τιμής όπως: name="value".

Λίστα συνηθισμένων χαρακτηριστικών:

- **href** – η ετικέτα <a> ορίζει έναν υπερσύνδεσμο. Το χαρακτηριστικό href καθορίζει τη διεύθυνση URL της σελίδας στην οποία μεταβαίνει ο σύνδεσμος, ως εξής:

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

- **src** – η ετικέτα <img> χρησιμοποιείται για την ενσωμάτωση μιας εικόνας σε μια σελίδα HTML. Το χαρακτηριστικό src καθορίζει τη διαδρομή προς την εικόνα που θα εμφανιστεί, όπως φαίνεται παρακάτω:

```
<img src="img_girl.jpg">
```

Υπάρχουν δύο τρόποι για να καθορίσετε τη διεύθυνση URL στο χαρακτηριστικό src:

1. **Απόλυτη διεύθυνση URL** - Σύνδεσμοι σε μια εξωτερική εικόνα που φιλοξενείται σε άλλο ιστότοπο. Π.χ.: *src="https://www.w3schools.com/images/img_girl.jpg".*

Σημειώσεις: Οι εξωτερικές εικόνες ενδέχεται να υπόκεινται σε **πνευματικά δικαιώματα**. Εάν κάποιος δεν πάρει άδεια να το χρησιμοποιήσει, μπορεί να παραβιάζει τους νόμους περί πνευματικών δικαιωμάτων. Επιπλέον, οι εξωτερικές εικόνες δεν μπορούν να ελεγχθούν αλλά μπορεί να αφαιρεθούν ή να αλλάξουν απότομα.

2. **Σχετική διεύθυνση URL** - Σύνδεσμοι σε μια εικόνα που φιλοξενείται στον ιστότοπο. Εδώ, η διεύθυνση URL δεν περιλαμβάνει το όνομα τομέα. Εάν η διεύθυνση URL ξεκινά χωρίς κάθετο, θα είναι σχετική με την τρέχουσα σελίδα. Π.χ.*: src="img_girl.jpg"*. Εάν η διεύθυνση URL ξεκινά με κάθετο, θα είναι σε σχέση με τον τομέα. Π.χ.: *src="/images/img_girl.jpg".*

Συμβουλή: Είναι σχεδόν πάντα καλύτερο να χρησιμοποιείτε σχετικές διευθύνσεις URL. Δεν θα σπάσουν αν αλλάξει ο τομέας.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

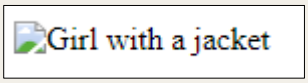
- **χαρακτηριστικά πλάτους και ύψους** – η ετικέτα <img> πρέπει επίσης να περιλαμβάνει τα χαρακτηριστικά πλάτους και ύψους, τα οποία καθορίζουν το πλάτος και το ύψος της εικόνας (σε pixel):

  < img src ="img_girl.jpg" width ="500" height ="600">

- **alt –** το απαιτούμενο χαρακτηριστικό alt για την ετικέτα <img> καθορίζει ένα εναλλακτικό κείμενο για μια εικόνα, εάν η εν λόγω εικόνα δεν μπορεί να εμφανιστεί για κάποιο λόγο. Αυτό μπορεί να οφείλεται σε αργή σύνδεση ή σε σφάλμα στο χαρακτηριστικό src ή εάν ο χρήστης χρησιμοποιεί πρόγραμμα ανάγνωσης οθόνης.

  < img src ="img_girl.jpg" alt ="Girl with a jacket">

  Εάν προσπαθήσουμε να εμφανίσουμε μια εικόνα που δεν υπάρχει, θα εμφανιστεί η τιμή του χαρακτηριστικού alt, ως εξής:



Εικόνα 13 *– Τιμή Alt εάν δεν υπάρχει εικόνα (Πηγή: https://www.w3schools.com/html)*

- **στυλ (style) –** το χαρακτηριστικό style χρησιμοποιείται για την προσθήκη στυλ σε ένα στοιχείο, όπως χρώμα, γραμματοσειρά, μέγεθος και άλλα.

```
<p style="color:red;">This is a red paragraph.</p>
```

Αποτέλεσμα:

```
<!DOCTYPE html>
<html>
<body>

<h2>The style Attribute</h2>
<p>The style attribute is used to add
styles to an element, such as color:
</p>

<p style="color:red;">This is a red
paragraph.</p>

</body>
</html>
```

The style Attribute

The style attribute is used to add styles to an element, such as color:

This is a red paragraph.

Εικόνα 14 *– Κώδικας για την προσθήκη χρώματος σε παράγραφο (Πηγή: https://www.w3schools.com/html)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- **lang** – το χαρακτηριστικό lang πρέπει πάντα να περιλαμβάνεται μέσα στην ετικέτα <html>, για να δηλωθεί η γλώσσα της ιστοσελίδας. Αυτό προορίζεται για την υποστήριξη μηχανών αναζήτησης και προγραμμάτων περιήγησης. Το παρακάτω παράδειγμα ορίζει τα αγγλικά ως τη γλώσσα που χρησιμοποιείται:

```
<!DOCTYPE html>
<html lang="en">
<body>
…
</body>
</html>
```

Οι κωδικοί χωρών μπορούν επίσης να προστεθούν στον κωδικό γλώσσας στο χαρακτηριστικό lang. Έτσι, οι δύο πρώτοι χαρακτήρες εκφράζουν τη **γλώσσα** της σελίδας HTML και οι δύο τελευταίοι χαρακτήρες περιγράφουν τη **χώρα**.

Το ακόλουθο παράδειγμα προσδιορίζει τα πορτογαλικά ως γλώσσα και την Πορτογαλία ως χώρα:

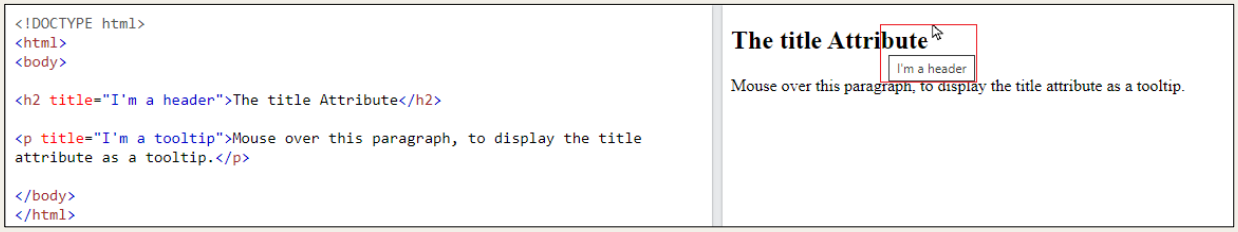
```
<!DOCTYPE html>
<html lang="pt-PT">
<body>
...
</body>
</html>
```

Η αναφορά κώδικα γλώσσας HTML περιλαμβάνει όλους τους κώδικες γλώσσας.

- **τίτλος (title) –** αυτό το χαρακτηριστικό περιγράφει μερικές πρόσθετες πληροφορίες σχετικά με ένα στοιχείο.

Η τιμή του θα εμφανίζεται ως εργαλείο συμβουλών όταν ο δείκτης του ποντικιού περάσει πάνω από το στοιχείο, ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 15 *– Κώδικας για την εμφάνιση ενός εργαλείου «ττitle» (Πηγή: Συντάκτης)*

Συμβουλή: Συνιστάται να χρησιμοποιείτε πάντα πεζούς χαρακτήρες και να αναφέρετε πάντα τιμές χαρακτηριστικών για αυστηρότερους τύπους εγγράφων όπως το XHTML.

Επικεφαλίδες HTML

Οι επικεφαλίδες HTML είναι τίτλοι ή υπότιτλοι που θέλουμε να εμφανίσει μια ιστοσελίδα.

Εικόνα 16 *– Κώδικας για την προσθήκη επικεφαλίδων (Πηγή: Συγγραφέας)*

Οι επικεφαλίδες HTML οριοθετούνται με τις ετικέτες <h1> με <h6>. Η <h1> προσδιορίζει την πιο σημαντική επικεφαλίδα. Η <h6> περιγράφει τη λιγότερο σημαντική επικεφαλίδα. Οι επικεφαλίδες <h1> θα πρέπει να χρησιμοποιούνται για τις κύριες επικεφαλίδες, ακολουθούμενες από επικεφαλίδες<h2>, μετά τις λιγότερο σημαντικές με την <h3> και ούτω καθεξής.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Οι επικεφαλίδες είναι υψίστης σημασίας, καθώς οι μηχανές αναζήτησης τις χρησιμοποιούν για να δείξουν τη δομή και το περιεχόμενο των ιστοσελίδων. Οι χρήστες συνήθως περιηγούνται σε μια ιστοσελίδα με βάση τις επικεφαλίδες της. Είναι σημαντικό να χρησιμοποιείτε επικεφαλίδες για να παρουσιάσετε τη δομή του εγγράφου.

Είναι σημαντικό να αναφερθεί ότι, τα προγράμματα περιήγησης προσθέτουν αυτόματα ένα περιθώριο πριν και μετά από μια επικεφαλίδα από προεπιλογή.

Συμβουλή: Συνιστάται να χρησιμοποιείτε επικεφαλίδες HTML μόνο για επικεφαλίδες, και όχι για να κάνετε το κείμενο πιο μεγάλο ή με έντονα γράμματα.

Επιπλέον, στο θέμα CSS, θα διδαχθεί ότι το μέγεθος των επικεφαλίδων μπορεί να καθοριστεί χρησιμοποιώντας το χαρακτηριστικό **style**, χρησιμοποιώντας την ιδιότητα μεγέθους γραμματοσειράς CSS, ως εξής:

```
<h1 style="font-size:60px;">Heading 1</h1>
```

Παράγραφοι HTML

Μια παράγραφος ξεκινά πάντοτε σε μια νέα γραμμή και είναι συνήθως ένα μπλοκ κειμένου. Ορίζεται από το στοιχείο <p> στην HTML και, όπως οι επικεφαλίδες, τα προγράμματα περιήγησης προσθέτουν αυτόματα κάποιο περιθώριο πριν και μετά από μια παράγραφο.

```
<!DOCTYPE html>
<html>
<body>

<p>Code4SP helps me to code.</p>
<p>I love Code4SP.</p>
<p>Coding is so great!</p>

</body>
</html>
```

Code4SP helps me to code.

I love Code4SP.

Coding is so great!

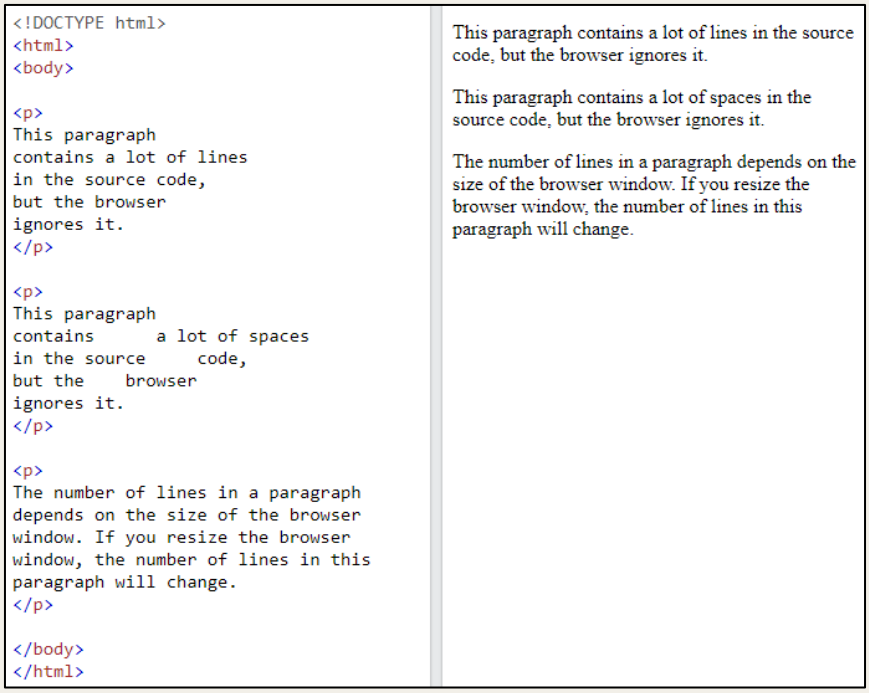
Εικόνα 17 *– Κώδικας για την προσθήκη παραγράφων (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εμφάνιση στην HTML

Δεν μπορούμε ποτέ να είμαστε σίγουροι πως θα παρουσιαστεί η HTML, καθώς μπορεί να διαφέρει από οθόνη σε οθόνη. Στην HTML, η εμφάνιση δεν μπορεί να αλλάξει προσθέτοντας επιπλέον κενά ή επιπλέον γραμμές στον κώδικα HTML.

Το πρόγραμμα περιήγησης θα αφαιρέσει αυτόματα τυχόν επιπλέον κενά και γραμμές όταν εμφανίζεται η σελίδα, όπως φαίνεται στο ακόλουθο παράδειγμα:



```
<!DOCTYPE html>
<html>
<body>

<p>
This paragraph
contains a lot of lines
in the source code,
but the browser
ignores it.
</p>

<p>
This paragraph
contains      a lot of spaces
in the source     code,
but the    browser
ignores it.
</p>

<p>
The number of lines in a paragraph
depends on the size of the browser
window. If you resize the browser
window, the number of lines in this
paragraph will change.
</p>

</body>
</html>
```

This paragraph contains a lot of lines in the source code, but the browser ignores it.

This paragraph contains a lot of spaces in the source code, but the browser ignores it.

The number of lines in a paragraph depends on the size of the browser window. If you resize the browser window, the number of lines in this paragraph will change.

Εικόνα 18 *– Παράδειγμα του τρόπου με τον οποίο τα προγράμματα περιήγησης τείνουν να αγνοούν τα κενά (Πηγή: https://www.w3schools.com/html)*

Αλλαγή γραμμής στην HTML

Το στοιχείο
 HTML ορίζει μια αλλαγή γραμμής. Είναι μια κενή ετικέτα, που σημαίνει ότι δεν έχει ετικέτα τέλους.

Το στοιχείο
 θα πρέπει να χρησιμοποιείται όταν θέλουμε να αλλάξουμε γραμμή, χωρίς να αλλάξουμε παράγραφο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html>
<body>

<p>Code4SP really is<br>an
amazing<br>project.</p>

</body>
</html>
```

Code4SP really is
an amazing
project.

Εικόνα 19 *– Το στοιχείο
 (Πηγή: Συντάκτης)*

Στυλ της HTML (Styles)

Το χαρακτηριστικό style της HTML χρησιμοποιείται για την προσθήκη ενός στυλ σε ένα στοιχείο, όπως χρώμα, γραμματοσειρά, μέγεθος κ.λπ. Για να ορίσετε το στυλ ενός στοιχείου HTML, πρέπει να χρησιμοποιηθεί το χαρακτηριστικό style. Έχει την ακόλουθη σύνταξη (θα πρέπει να σημειωθεί ότι η *ιδιότητα* και η *τιμή* είναι χαρακτηριστικά CSS, που θα μάθουμε αργότερα).

```
<tagname style="property:value;">
```

- **Χρώμα φόντου**

Η ιδιότητα *χρώματος* φόντου CSS καθορίζει το χρώμα φόντου για ένα στοιχείο HTML.

```
<!DOCTYPE html>
<html>
<body style="background-color:green;">

<h1>Code4SP</h1>
<p>Coding for Social Promotion.</p>

</body>
</html>
```

Code4SP

Coding for Social Promotion.

Εικόνα 20 *– Ορισμός χρώματος φόντου (Πηγή: Συντάκτης)*

- **Χρώμα κειμένου**

Η ιδιότητα *χρώματος* CSS περιγράφει το χρώμα κειμένου για ένα στοιχείο HTML:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;">Code4SP</h1>
<p style="color:red;">Coding for Social
Promotion.</p>

</body>
</html>
```

Code4SP

Coding for Social Promotion.

Εικόνα 21 *– Ορισμός χρώματος φόντου (Πηγή: Συντάκτης)*

- **Γραμματοσειρές**

Η ιδιότητα CSS *font-family* ορίζει τη γραμματοσειρά που θα χρησιμοποιηθεί για ένα στοιχείο HTML:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="font-
family:verdana;">Code4SP</h1>
<p style="font-family:courier;">Coding
for Social Promotion.</p>

</body>
</html>
```

Code4SP

Coding for Social Promotion.

Εικόνα 22 *– Ρύθμιση της γραμματοσειράς που θα χρησιμοποιηθεί για ένα στοιχείο HTML (Πηγή: Συντάκτης)*

- **Μέγεθος Κειμένου**

Η ιδιότητα *μεγέθους γραμματοσειράς* CSS προσδιορίζει το μέγεθος κειμένου για ένα στοιχείο HTML:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="font-
size:300%;">CODE4SP</h1>
<p style="font-size:160%;">Coding for
Social Promotion.</p>

</body>
</html>
```

CODE4SP

Coding for Social Promotion.

Εικόνα 23 *– Ρύθμιση του μεγέθους κειμένου για ένα στοιχείο HTML (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- **Στοίχιση Κειμένου**

Η δυνατότητα *στοίχισης* κειμένου CSS ορίζει την οριζόντια στοίχιση κειμένου για ένα στοιχείο HTML:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="text-
align:center;">CODE4SP</h1>
<p style="text-align:center;">Coding
for Social Promotion.</p>

</body>
</html>
```

CODE4SP

Coding for Social Promotion.

Εικόνα 24 *– Λειτουργία στοίχισης κειμένου (Πηγή: Συντάκτης)*

Μορφοποίηση κειμένου στην HTML

Η HTML περιλαμβάνει διάφορα στοιχεία για τον καθορισμό του κειμένου με ένα ειδικό χαρακτηριστικό (έντονη, πλάγια γραφή, δείκτης, εκθέτης κ.λπ.).

Τα παρακάτω είναι τα **στοιχεία μορφοποίησης στην HTML**:

Στοιχείο	**Αποτέλεσμα**	**Ορισμός**	**Παράδειγμα**
<b>	Εντονο Κείμενο	Το στοιχείο <b> της HTML καθορίζει το έντονο κείμενο, χωρίς καμία επιπλέον σημασία.	**Εντονο Κείμενο**
	Σημαντικό κείμενο	Το στοιχείο της HTML περιγράφει κείμενο με μεγάλη σημασία. Το περιεχόμενο στο εσωτερικό εμφανίζεται συνήθως με έντονη γραφή.	**Σημαντικό κείμενο**
<i>	Πλάγιο κείμενο	Το στοιχείο <i> της HTML ορίζει ένα μέρος του κειμένου με	*Πλάγιο κείμενο*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

		εναλλακτικό ύφος ή διάθεση. Το περιεχόμενο στο εσωτερικό εμφανίζεται συνήθως σε πλάγια γραφή.	
	Κείμενο με έμφαση	Το στοιχείο της HTML ορίζει κείμενο με έμφαση. Το περιεχόμενο στο εσωτερικό εμφανίζεται συνήθως με πλάγια γραφή.	*Κείμενο με έμφαση*
<mark>	Επισημασμένο κείμενο	Το στοιχείο <mark> της HTML ορίζει κείμενο που πρέπει να επισημανθεί ή να υπογραμμιστεί.	Επισημασμένο κείμενο
<small>	Μικρότερο κείμενο	Το στοιχείο <small> της HTML ορίζει ένα μικρότερο κείμενο.	Μικρότερο κείμενο
<del>	Διαγραμμένο κείμενο	Το στοιχείο <del> της HTML ορίζει κείμενο που έχει διαγραφεί από ένα έγγραφο. Τα προγράμματα περιήγησης συνήθως έχουν μια γραμμή πάνω από το διαγραμμένο κείμενο	Διαγραμμένο ~~κείμενο~~
<ins>	Εισαγωγή κειμένου	Το στοιχείο <ins> της HTML ορίζει ένα κείμενο που έχει εισαχθεί σε ένα έγγραφο. Τα προγράμματα περιήγησης συνήθως υπογραμμίζουν το εισαγόμενο κείμενο:	~~Εισαγωγή~~ <u>κειμένου</u>.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

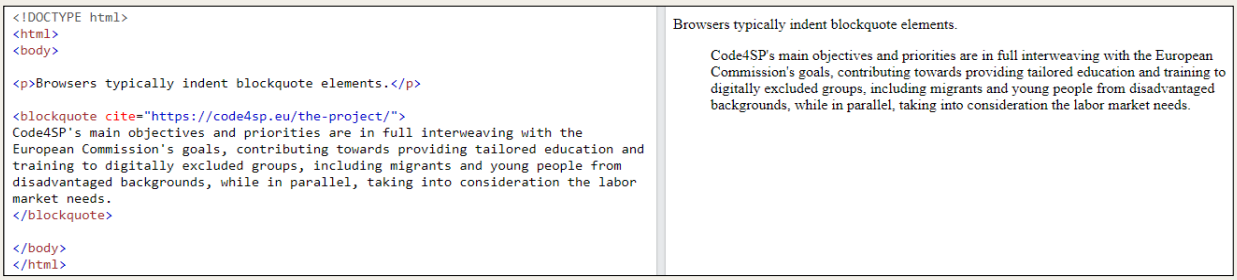
<sub>	Δείκτης κειμένου	Το στοιχείο <sub> της HTML εκφράζει έναν δείκτη κειμένου. Το κείμενο δείκτη εμφανίζει μισό χαρακτήρα κάτω από την κανονική γραμμή και μερικές φορές αποδίδεται με μικρότερη γραμματοσειρά. Ο δείκτης κειμένου μπορεί να χρησιμοποιηθεί για τη Χημεία, όπως στο H_2O.	Δείκτης κειμένου
<sup>	Εκθέτης κειμένου	Το στοιχείο <sup> της HTML καθορίζει έναν εκθέτη κειμένου. Εκθέτης κειμένου εμφανίζεται κατά μισό χαρακτήρα πάνω από την κανονική γραμμή και μερικές φορές αποδίδεται με μικρότερη γραμματοσειρά. Ο εκθέτης κειμένου μπορεί να χρησιμοποιηθεί για υποσημειώσεις, όπως WWW [1]:	Εκθέτης κειμένου

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μορφοποίηση κειμένου στην HTML

Η ετικέτα <blockquote> για παραθέσεις

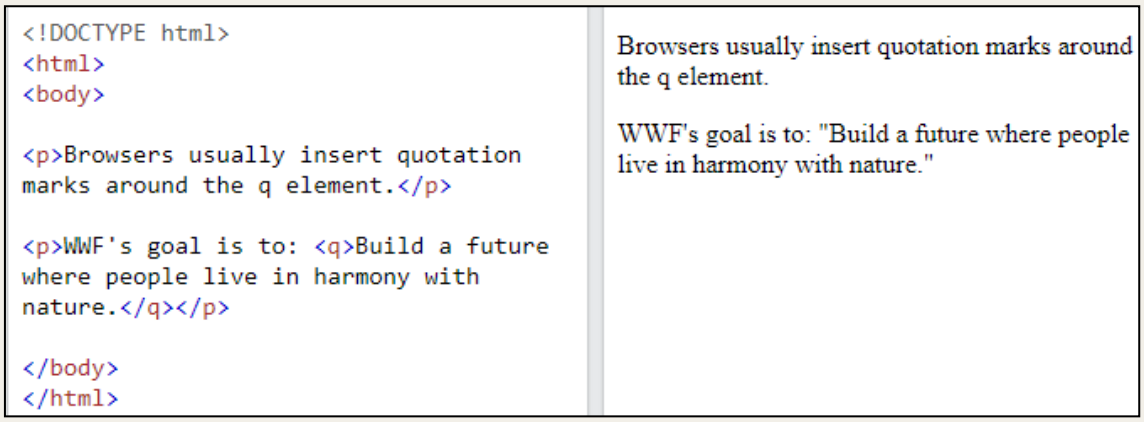
Η ετικέτα <blockquote> της HTML προσδιορίζει μια ενότητα που παρατίθεται από άλλη πηγή. Τα προγράμματα περιήγησης συνήθως εισάγουν στοιχεία <blockquote>, όπως φαίνεται παρακάτω:



Εικόνα 25 *– Το στοιχείο blockquote (Πηγή: Συντάκτης)*

Η ετικέτα <q> για σύντομες παραθέσεις

Η ετικέτα < q > στην HTML καθορίζει μια σύντομη παράθεση. Τα προγράμματα περιήγησης γενικά εισάγουν εισαγωγικά γύρω από το παράθεμα, ως εξής:



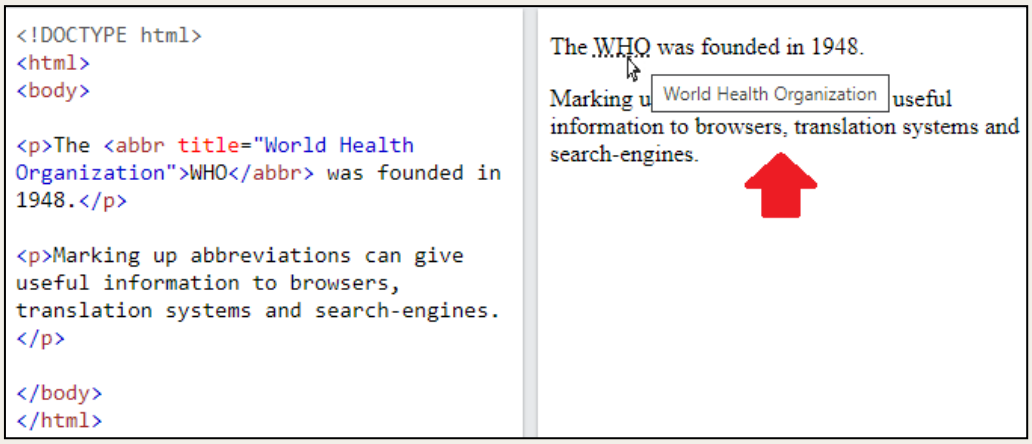
Εικόνα 26 *– Το στοιχείο του σύντομου παραθέματος (Πηγή: https://www.w3schools.com/html/html_quotation_elements.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η ετικέτα <abbr> για συντομογραφίες

Η ετικέτα <abbr> στην HTML καθορίζει μια συντομογραφία ή ένα ακρωνύμιο, όπως «HTML», «CSS», «Mr.», «Dr.», «ASAP» κ.λπ. Οι συντομογραφίες σήμανσης μπορούν να δώσουν πολύτιμες πληροφορίες σε προγράμματα περιήγησης, συστήματα μετάφρασης και μηχανές αναζήτησης, όπως είδαμε προηγουμένως.

Σε περίπτωση που κάποιος δεν γνωρίζει τη σημασία οποιασδήποτε δεδομένης συντομογραφίας, θα μπορούσε να χρησιμοποιήσει το παγκόσμιο χαρακτηριστικό title για να εμφανίσει την περιγραφή για τη συντομογραφία/ακρωνύμιο όταν τοποθετείται ο δείκτης του ποντικιού πάνω από το στοιχείο, όπως φαίνεται παρακάτω:



Εικόνα 27 *– Το στοιχείο <abbr> και ο δείκτης του ποντικιού πάνω από τη συντομογραφία (Πηγή: Συγγραφέας)*

Η ετικέτα <address> για Στοιχεία Επικοινωνίας

Η ετικέτα <address> στην HTML ορίζει τα στοιχεία επικοινωνίας για τον συντάκτη/κάτοχο ενός εγγράφου ή ενός άρθρου. Μπορεί να είναι μια διεύθυνση ηλεκτρονικού ταχυδρομείου, διεύθυνση URL, φυσική διεύθυνση, αριθμός τηλεφώνου κ.λπ. Το κείμενο που περιλαμβάνεται στο στοιχείο <address> παρουσιάζεται συνήθως με πλάγια γραφή και τα προγράμματα περιήγησης αλλάζουν γραμμή πριν και μετά από αυτό, ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html>
<body>

<p>Please contact us:.</p>

<address>
https://code4sp.eu/<br>
https://www.facebook.com/Code4SP<br>
</address>

</body>
</html>
```

Please contact us:.

https://code4sp.eu/
https://www.facebook.com/Code4SP

Εικόνα 28 *– Το στοιχείο <address> (Πηγή: Συντάκτης)*

Η ετικέτα <cite> για Τίτλο Έργου

Η ετικέτα <cite> στην HTML ορίζει τον τίτλο ενός βιβλίου, ενός ποιήματος, ενός τραγουδιού, μιας ταινίας, ενός πίνακα ζωγραφικής και όλων των δημιουργικών έργων. Ωστόσο, πρέπει να σημειωθεί ότι το όνομα του δημιουργού δεν αποτελεί τίτλο έργου. Όπως και στις προαναφερθείσες ετικέτες, το κείμενο εντός του στοιχείου συνήθως αποδίδεται με πλάγια γραφή:

```
<!DOCTYPE html>
<html>
<body>

<img src="img_the_scream.jpg"
width="220" height="277" alt="The
Scream">
<p><cite>The Scream</cite> by Edvard
Munch, 1893.</p>

</body>
</html>
```

The Scream by Edvard Munch, 1893.

Εικόνα 29 *– Το στοιχείο <cite> (Πηγή: https://www.w3schools.com/html/html_quotation_elements.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η ετικέτα <bdo> για αμφίδρομη παράκαμψη

Η ετικέτα <bdo> HTML είναι τα αρχικά των λέξεων «bidirectional override» που σημαίνουν «παράκαμψη διπλής κατεύθυνσης» που χρησιμοποιείται για την παράκαμψη της τρέχουσας/προεπιλεγμένης κατεύθυνσης κειμένου. Αυτή η ετικέτα ορίζει την κατεύθυνση του περιεχομένου της για εκτέλεση στο πρόγραμμα περιήγησης από αριστερά προς τα δεξιά ή από δεξιά προς τα αριστερά (rtl – για δεξιά προς τα αριστερά, ltf – για αριστερά προς τα δεξιά).

```
<!DOCTYPE html>
<html>
<body>

<p>If your browser supports bi-
directional override (bdo), the next
line will be written from right to left
(rtl):</p>

<bdo dir="rtl">Code4SP</bdo>

</body>
</html>
```

If your browser supports bi-directional override (bdo), the next line will be written from right to left (rtl):

PS4edoC

Εικόνα 30 *– Το στοιχείο <bdo> (Πηγή: https://www.w3schools.com/html/html_quotation_elements.asp)*

Σχόλια στην HTML

Προσθήκη σχολίων

Αυτό το στοιχείο χρησιμοποιείται για την προσθήκη σχολίων σε ένα έγγραφο HTML. Ένα σχόλιο στην HTML ξεκινά με <!— και τελειώνει με —>. Τα σχόλια στην HTML είναι ορατά σε οποιονδήποτε βλέπει τον πηγαίο κώδικα της σελίδας, αλλά δεν αποδίδεται όταν το έγγραφο HTML αποδίδεται από ένα πρόγραμμα περιήγησης. Πρέπει να σημειωθεί ότι υπάρχει θαυμαστικό στην ετικέτα αρχής, αλλά όχι στην ετικέτα τέλους. Αυτή η δυνατότητα είναι ιδιαίτερα χρήσιμη για την τοποθέτηση ειδοποιήσεων και υπενθυμίσεων στον κώδικα HTML:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html>
<body>

<!-- This is a comment -->
<p>Code4SP project.</p>
<!-- Comments are not displayed in the
browser -->

</body>
</html>
```

Code4SP project.

Εικόνα 31 *– Το στοιχείο προσθήκης σχολείων (Πηγή: Συντάκτης)*

Απόκρυψη περιεχομένου

Τα σχόλια μπορούν επίσης να χρησιμοποιηθούν για την απόκρυψη περιεχομένου και αυτό μπορεί να είναι χρήσιμο αν θέλετε να κρύψετε κάτι προσωρινά. Μπορείτε επίσης να αποκρύψετε περισσότερες από μία γραμμές, οτιδήποτε βρίσκεται μεταξύ των <!-- και του --> δεν θα φαίνεται. Τα σχόλια είναι επίσης εξαιρετικά για την αποσφαλμάτωση στην HTML, καθώς μας επιτρέπει να χρησιμοποιήσουμε τα σχόλια σε γραμμές κώδικα της HTML, μία-μία για να αναζητήσουμε σφάλματα.

```
<!DOCTYPE html>
<html>
<body>

<p>Code4SP project.</p>

<!-- <p>This content is hidden. </p> --
>

<p>But this will appear.</p>

</body>
</html>
```

Code4SP project.

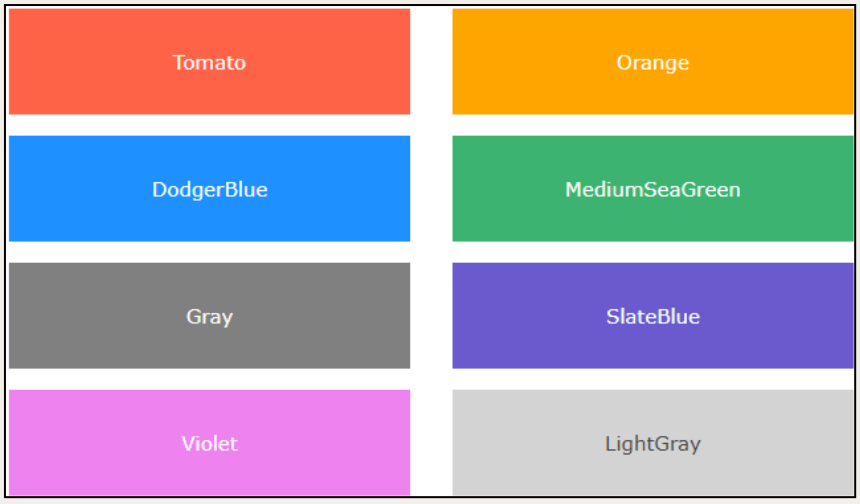
But this will appear.

Εικόνα 32 *– Η δυνατότητα απόκρυψης περιεχομένου (Πηγή: Συντάκτης)*

Χρώματα στην HTML

Τα χρώματα στην HTML ορίζονται με προκαθορισμένα ονόματα χρωμάτων ή με τιμές RGB, HEX, HSL, RGBA ή HSLA.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 33 *– Ορισμένα ονόματα χρωμάτων που μπορούν να οριστούν (Πηγή: https://www.w3schools.com/html/html_colors.asp)*

Τα χρώματα μπορούν επίσης να οριστούν για το **φόντο της σελίδας**:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="background-
color:DodgerBlue;">Code4SP</h1>

<p style="background-color:Tomato;">
Code4SP's main objectives and
priorities are in full interweaving
with the European Commission's goals,
contributing towards providing tailored
education and training to digitally
excluded groups, including migrants and
young people from disadvantaged
backgrounds, while in parallel, taking
into consideration the labor market
needs.

</p>

</body>
</html>
```

Code4SP

Code4SP's main objectives and priorities are in full interweaving with the European Commission's goals, contributing towards providing tailored education and training to digitally excluded groups, including migrants and young people from disadvantaged backgrounds, while in parallel, taking into consideration the labor market needs.

Εικόνα 34 *– Καθορισμός του χρώματος φόντου μιας ιστοσελίδας (Πηγή:Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η ίδια αρχή μπορεί να εφαρμοστεί και για το **χρώμα του κειμένου**:

```
<!DOCTYPE html>
<html>
<body>

<h3 style="color:Tomato;">Code4SP</h3>

<p style="color:DodgerBlue;">The target
will be reached through the upscaling
of an already existing good practice at
a local level in Germany, which had as
a result, top paid programming jobs for
asylum seekers.</p>

<p
style="color:MediumSeaGreen;">Enhance
employers' motivation and
predisposition for potential employment
of individuals that belong to
disadvantaged populations, thus
breaking any negative stereotypes on
this issue.</p>

</body>
</html>
```

Code4SP

The target will be reached through the upscaling of an already existing good practice at a local level in Germany, which had as a result, top paid programming jobs for asylum seekers.

Enhance employers' motivation and predisposition for potential employment of individuals that belong to disadvantaged populations, thus breaking any negative stereotypes on this issue.

Εικόνα 35 *– Καθορισμός του χρώματος του κειμένου (Πηγή:Συντάκτης)*

Επίσης, για το χρώμα των **περιγραμμάτων**:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="border: 2px solid
Tomato;">Code4SP</h1>

<h1 style="border: 2px solid
DodgerBlue;">Code4SP</h1>

<h1 style="border: 2px solid
Violet;">Code4SP</h1>

</body>
</html>
```

Code4SP

Code4SP

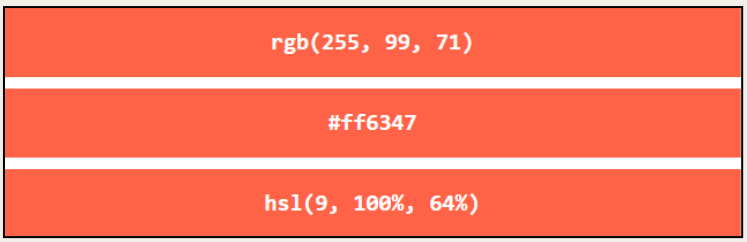
Code4SP

Εικόνα 36 *– Προσθήκη περιγραμμάτων και καθορισμός του χρώματός τους (Πηγή:Συντάκτης)*

Τιμές χρωμάτων

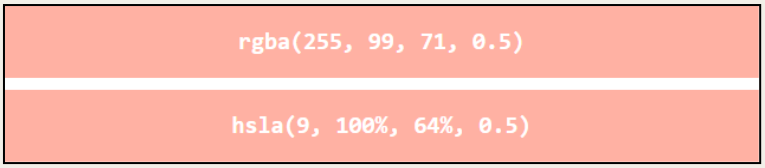
Τα χρώματα στην HTML ορίζονται με προκαθορισμένα ονόματα χρωμάτων ή με τιμές RGB, HEX, HSL, RGBA ή HSLA. Στα ακόλουθα τρία στοιχεία το χρώμα φόντου τους καθορίστηκε με τις τιμές RGB, HEX και HSL:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 37 *– Τιμές RGB, HEX και HSL για το χρώμα Tomato (Πηγή: https://www.w3schools.com/html/html_colors.asp)*

Η διαφάνεια είναι άλλη μια δυνατότητα που θα μπορούσε να προστεθεί κατά τον καθορισμό ενός χρώματος, προσθέτοντας ένα άλφα στις τιμές. Το παράδειγμα που ακολουθεί παρουσιάζει 50% διαφάνεια:



Εικόνα 38 *– Ορισμός τιμών διαφάνειας για το χρώμα Tomato (Πηγή: https://www.w3schools.com/html/html_colors.asp)*

Ο κώδικας για τη ρύθμιση και των δύο λειτουργιών θα βοηθήσει τους εκπαιδευόμενους να τα μάθουν καλύτερα. Όπως φαίνεται παρακάτω, περιλαμβάνει χαρακτηριστικά CSS τα οποία θα εξερευνηθούν αργότερα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html>
<body>

<p>Same as color name "Tomato":</p>

<h1 style="background-color:rgb(255,
99, 71);">rgb(255, 99, 71)</h1>
<h1 style="background-
color:#ff6347;">#ff6347</h1>
<h1 style="background-color:hsl(9,
100%, 64%);">hsl(9, 100%, 64%)</h1>

<p>Same as color name "Tomato", but 50%
transparent:</p>
<h1 style="background-color:rgba(255,
99, 71, 0.5);">rgba(255, 99, 71, 0.5)
</h1>
<h1 style="background-color:hsla(9,
100%, 64%, 0.5);">hsla(9, 100%, 64%,
0.5)</h1>

<p>In addition to the predefined color
names, colors can be specified using
RGB, HEX, HSL, or even transparent
colors using RGBA or HSLA color values.
</p>

</body>
</html>
```

Same as color name "Tomato":

rgb(255, 99, 71)

#ff6347

hsl(9, 100%, 64%)

Same as color name "Tomato", but 50% transparent:

rgba(255, 99, 71, 0.5)

hsla(9, 100%, 64%, 0.5)

In addition to the predefined color names, colors can be specified using RGB, HEX, HSL, or even transparent colors using RGBA or HSLA color values.

Εικόνα 39 *– Ρύθμιση χρώματος και διαφάνειας του χρώματος Tomato (Πηγή: https://www.w3schools.com/html/html_colors.asp)*

Περισσότερες πληροφορίες σχετικά με τις τιμές RGB, HEX, HSL, RGBA ή HSLA μπορείτε να βρείτε στη διεύθυνση https://www.w3schools.com/html/html_colors.asp

Σύνδεσμοι στην HTML

Οι σύνδεσμοι βρίσκονται σχεδόν σε όλες τις ιστοσελίδες. Επιτρέπουν στους χρήστες του Διαδικτύου να πλοηγούνται από σελίδα σε σελίδα. Δεν χρειάζεται να είναι κείμενο, καθώς μπορεί να είναι μια εικόνα ή οποιοδήποτε άλλο στοιχείο HTML.

Υπερσύνδεσμος

Οι σύνδεσμοι HTML είναι υπερσύνδεσμοι, στους οποίους μπορείτε να κάνετε κλικ, μεταβαίνοντας σε άλλο έγγραφο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Όταν το ποντίκι μετακινηθεί πάνω από έναν σύνδεσμο, το βέλος του ποντικιού θα μετατραπεί σε ένα μικρό χέρι.

Σύνταξη

Η ετικέτα <a> HTML ορίζει έναν υπερσύνδεσμο. Έχει την εξής σύνταξη:

```
<a href="url">link text</a>
```

Το πιο σημαντικό χαρακτηριστικό του στοιχείου <a> είναι το χαρακτηριστικό href, το οποίο υποδεικνύει τον προορισμό του συνδέσμου. Το κείμενο του συνδέσμου είναι το τμήμα που θα είναι ορατό στον χρήστη. Κάνοντας κλικ στο κείμενο του συνδέσμου, ο χρήστης θα ανακατευθυνθεί στη συνδεδεμένη διεύθυνση URL.

Από προεπιλογή, οι σύνδεσμοι θα εμφανίζονται σε όλα τα προγράμματα περιήγησης ως εξής:

- Ένας σύνδεσμος που δεν έχει πατηθεί είναι υπογραμμισμένος και μπλε
- Ένας σύνδεσμος που έχει πατηθεί είναι υπογραμμισμένος και μωβ
- Ένας ενεργός σύνδεσμος είναι υπογραμμισμένος και κόκκινος

Σημείωση: Τα χρώματα των συνδέσμων μπορούν να αλλάξουν χρησιμοποιώντας λειτουργίες CSS.

Το χαρακτηριστικό target

Από προεπιλογή, η συνδεδεμένη σελίδα θα εμφανίζεται στο τρέχον παράθυρο του προγράμματος περιήγησης. Για να τροποποιηθεί αυτό, οι εκπαιδευόμενοι θα πρέπει να υποδείξουν έναν άλλο στόχο (target) για τον σύνδεσμο.

Το χαρακτηριστικό target υποδεικνύει πού θα ανοίξει το συνδεδεμένο έγγραφο. Μπορεί να έχει μία από τις ακόλουθες τιμές:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- _self - Προεπιλογή. Ανοίγει το έγγραφο στο ίδιο παράθυρο/καρτέλα όπου έγινε κλικ
- _blank - Ανοίγει το έγγραφο σε νέο παράθυρο ή καρτέλα
- _parent - Ανοίγει το έγγραφο στο κεντρικό πλαίσιο
- _top - Ανοίγει το έγγραφο σε όλο το παράθυρο

Χρησιμοποιώντας μια εικόνα ως σύνδεσμο

Για να χρησιμοποιήσετε μια εικόνα ως σύνδεσμο, απλώς τοποθετήστε την ετικέτα <img> μέσα στην ετικέτα <a>, όπως μπορείτε να δείτε και στον παρακάτω οδηγό: https://www.w3schools.com/html/tryit.asp?filename=tryhtml_links_image

Σύνδεση διεύθυνσης ηλεκτρονικού ταχυδρομείου

Για να δημιουργήσετε έναν σύνδεσμο που ανοίγει το λογισμικό ηλεκτρονικού ταχυδρομείου (email) του χρήστη (επιτρέποντάς του να στείλει ένα νέο email), πρέπει να προστεθεί το mailto: μέσα στο χαρακτηριστικό href, ακολουθώντας το επόμενο παράδειγμα:

```
<a href="mailto:someone@example.com">Send email</a>
```

Δημιουργία Σελιδοδείκτη στην HTML

Οι σύνδεσμοι στην HTML μπορούν να εφαρμοστούν για τη δημιουργία σελιδοδεικτών, έτσι ώστε οι αναγνώστες να μπορούν να μεταβούν σε συγκεκριμένα μέρη μιας ιστοσελίδας, κάτι που μπορεί να είναι πολύ χρήσιμο εάν η ιστοσελίδα είναι πολύ μεγάλη. Αυτή η διαδικασία αποτελείται από δύο πολύ απλά βήματα:

- Για να δημιουργήσετε έναν σελιδοδείκτη - πρώτα πρέπει να δημιουργηθεί ο σελιδοδείκτης και στη συνέχεια να προστεθεί ένας σύνδεσμος σε αυτόν. Για τη δημιουργία, θα πρέπει να χρησιμοποιηθεί το χαρακτηριστικό id (π.χ. <h2 id="C1">Chapter 1</h2>), στη συνέχεια, θα πρέπει να προστεθεί ένας σύνδεσμος προς τον σελιδοδείκτη, από την ίδια σελίδα (π.χ. Jump to Chapter 4)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Όταν κάνετε κλικ στον σύνδεσμο, η σελίδα θα μετακινηθεί στο σημείο όπου τοποθετήθηκε ο σελιδοδείκτης.

Εικόνες στην HTML

Εισαγωγή εικόνων σε ιστοσελίδες

Οι εικόνες βελτιώνουν την οπτική εμφάνιση των ιστοσελίδων καθιστώντας τις πιο ελκυστικές και πολύχρωμες. Η ετικέτα <img> χρησιμοποιείται για την προσθήκη εικόνων σε σελίδες HTML. Είναι ένα κενό στοιχείο και περιέχει μόνο ιδιότητες.

Κάθε εικόνα πρέπει να έχει τουλάχιστον δύο χαρακτηριστικά: τα χαρακτηριστικά src και alt. Το χαρακτηριστικό src ενημερώνει το πρόγραμμα περιήγησης για το πού να βρει την εικόνα, καθώς η τιμή του είναι η διεύθυνση URL του αρχείου εικόνας. Το χαρακτηριστικό alt παρέχει ένα εναλλακτικό κείμενο για την εικόνα εάν δεν είναι προσβάσιμη ή για κάποιο λόγο δεν μπορεί να εμφανιστεί (π.χ. αργή σύνδεση, η εικόνα να μην είναι διαθέσιμη στην καθορισμένη διεύθυνση URL ή εάν ο χρήστης χρησιμοποιεί πρόγραμμα ανάγνωσης οθόνης ή πρόγραμμα περιήγησης χωρίς γραφικά). Η τιμή του θα πρέπει να είναι ένα ουσιαστικό υποκατάστατο της εικόνας, κατά προτίμηση ένα προτεινόμενο κείμενο.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Placing Images in HTML Documents</title>
</head>
<body>
    <img
src="https://i.guim.co.uk/img/media/19e048685db029092e5999e0b393e8318dacf87b/130
_238_4779_2867/master/4779.jpg?
width=700&quality=85&auto=format&fit=max&s=e3816218448245e52cf3cd3d931b8387"
alt="Wild Lion">
</body>
</html>
```

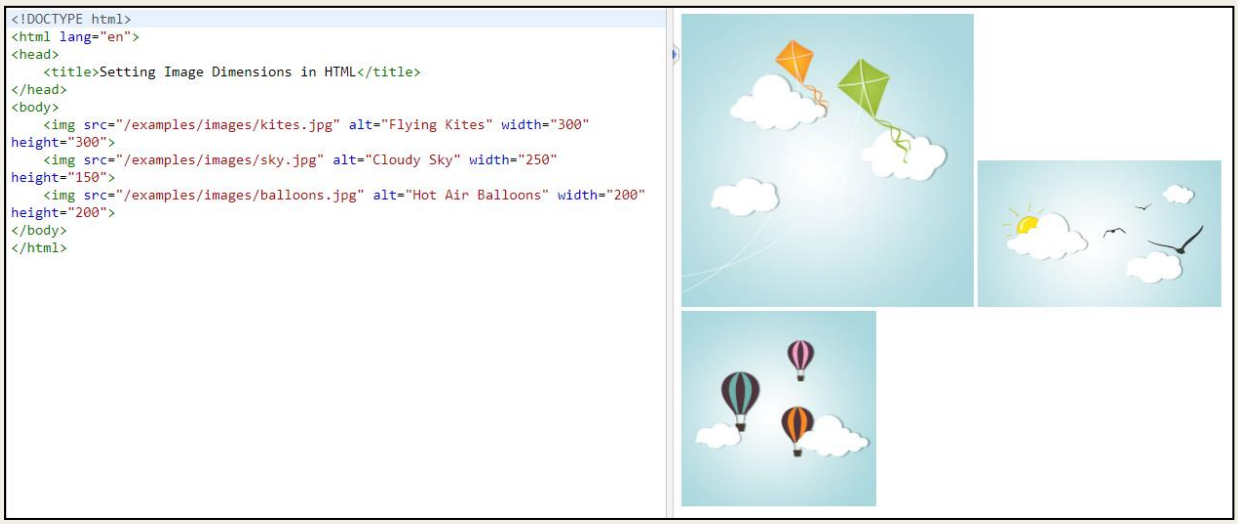
Εικόνα 40 *– Προσθήκη εικόνας σε έγγραφο HTML, χρησιμοποιώντας τα χαρακτηριστικά src και alt (Πηγή:Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ρύθμιση του πλάτους και του ύψους μιας εικόνας

Τα χαρακτηριστικά width και height χρησιμοποιούνται για να υποδείξουν το πλάτος και το ύψος μιας εικόνας.

Οι τιμές αυτών των χαρακτηριστικών ερμηνεύονται σε pixel από προεπιλογή. Είναι καλή πρακτική να προσδιορίζετε και τα δύο χαρακτηριστικά, έτσι ώστε το πρόγραμμα περιήγησης να μπορεί να παραχωρήσει αρκετό χώρο στην εικόνα προτού μεταφερθεί.



Εικόνα 41 *– Ρύθμιση του ύψους και του πλάτους μιας εικόνας (Πηγή: https://www.tutorialrepublic.com/codelab.php?topic=html&file= specify-dimensions-for-images)*

Το χαρακτηριστικό style μπορεί επίσης να χρησιμοποιηθεί για να υποδείξει το πλάτος και το ύψος. Εμποδίζει τα φύλλα του style να αλλάξουν το μέγεθος της εικόνας κατά λάθος, επειδή το ενσωματωμένο χαρακτηριστικό style έχει την υψηλότερη προτεραιότητα.

Χρήση του χαρακτηριστικού της HTML5 για εικόνες

Κάποιες φορές, η κλιμάκωση μιας εικόνας προς τα πάνω ή προς τα κάτω για να ταιριάζει σε διαφορετικές συσκευές (ή μεγέθη οθόνης) δεν λειτουργεί όπως αναμένεται. Επιπλέον, η μείωση των διαστάσεων της εικόνας χρησιμοποιώντας τα χαρακτηριστικά width και height δεν μειώνει το αρχικό μέγεθος αρχείου. Για την επίλυση αυτών των

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

προβλημάτων, η HTML5 εισήγαγε την ετικέτα <picture> που επιτρέπει τον καθορισμό πολλαπλών εκδοχών μιας εικόνας που στοχεύουν σε διαφορετικούς τύπους συσκευών.

Το στοιχείο <picture> περιέχει μηδέν ή περισσότερα στοιχεία <Πηγή>, καθένα από τα οποία αναφέρεται σε διαφορετική πηγή εικόνας και ένα στοιχείο <img> στο τέλος. Ομοίως, κάθε στοιχείο <Πηγή> έχει το χαρακτηριστικό media που καθορίζει μια συνθήκη πολυμέσων που χρησιμοποιείται από το πρόγραμμα περιήγησης για να καθορίσει πότε πρέπει να χρησιμοποιηθεί μια συγκεκριμένη πηγή.

Χάρτες Εικόνων

Ένας χάρτης εικόνων μας επιτρέπει να ορίσουμε τα κομβικά σημεία (hotspot) σε μια εικόνα που λειτουργεί ακριβώς όπως ένας υπερσύνδεσμος. Η βασική ιδέα πίσω από τη δημιουργία ενός χάρτη εικόνων είναι να δώσουμε έναν απλό τρόπο σύνδεσης διαφόρων τμημάτων μιας εικόνας χωρίς να τη διαιρέσουμε σε ξεχωριστά αρχεία εικόνων. Για παράδειγμα, ένας χάρτης μιας χώρας μπορεί να έχει κάθε πόλη συνδεδεμένη με περισσότερες πληροφορίες σχετικά με αυτήν την πόλη.

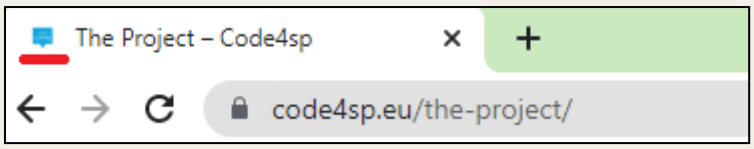
Το παρακάτω παράδειγμα είναι αρκετά ακριβές σχετικά με αυτά τα χαρακτηριστικά: https://www.tutorialrepublic.com/codelab.php?topic=html&file=image-maps

Το χαρακτηριστικό name της ετικέτας <map> χρησιμοποιείται για την αναφορά του χάρτη από την ετικέτα <img> χρησιμοποιώντας το χαρακτηριστικό usemap της. Η ετικέτα <area> χρησιμοποιείται μέσα στο στοιχείο <map> για να ορίσει τις περιοχές με δυνατότητα κλικ σε μια εικόνα. Οποιοσδήποτε αριθμός σημείων με δυνατότητα κλικ μπορεί να οριστεί σε μια εικόνα.

Εικονίδιο Συντόμευσης στην HTML (Favicon)

Το favicon είναι ένα εικονίδιο που εμφανίζεται στα αριστερά του τίτλου της σελίδας στην καρτέλα του προγράμματος περιήγησης:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 42 *– Favoticon του Code4SP (Πηγή: Συντάκτης)*

Για να προσθέσετε ένα favicon σε έναν ιστότοπο, θα πρέπει να αποθηκεύσετε μια εικόνα favicon στον βασικό κατάλογο του διακομιστή ιστού. Ένας άλλος τρόπος είναι να δημιουργήσετε έναν φάκελο στον βασικό κατάλογο που ονομάζεται εικόνες και, στη συνέχεια, να αποθηκεύσετε την εικόνα favicon σε αυτόν τον φάκελο. Ένα κοινό όνομα για μια εικόνα favicon είναι «favicon.ico».

Στη συνέχεια, θα πρέπει να προσθέσετε ένα στοιχείο <link> στοιχείο θ στο αρχείο «index.html», και μετά το στοιχείο <title>, ως εξής:

```
<!DOCTYPE html>
<html>
<head>
  <title>Code4SP</title>
  <link rel="icon" type="image/x-icon" href="/images/favicon.ico">
</head>
<body>
<h1>Our project</h1>
<p>Co-funded by the E+ fund of the EC.</p>
</body>
</html>
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Δημιουργία πινάκων στην HTML

Οι πίνακες στην HTML επιτρέπουν τη διάταξη των δεδομένων σε γραμμές και στήλες. Χρησιμοποιούνται γενικά για την εμφάνιση δεδομένων ενός πίνακα όπως καταχωρίσεις προϊόντων, στοιχεία πελατών, οικονομικές αναφορές κ.λπ.

Ένας πίνακας μπορεί να δημιουργηθεί χρησιμοποιώντας το στοιχείο <table>. Μέσα στο στοιχείο <table>, τα στοιχεία <tr> μπορούν να χρησιμοποιηθούν για τη δημιουργία σειρών, ενώ για τη δημιουργία στηλών μέσα σε μια γραμμή, μπορούν να χρησιμοποιηθούν τα στοιχεία <td>. Ένα κελί μπορεί να οριστεί ως κεφαλίδα για ένα σύνολο κελιών ενός πίνακα που χρησιμοποιούν το στοιχείο <th>.

Οι πίνακες δεν έχουν περιγράμματα από προεπιλογή. Η ιδιότητα border στην CSS μπορεί να χρησιμοποιηθεί για την προσθήκη περιγραμμάτων στους πίνακες. Επιπλέον, τα κελιά του πίνακα έχουν αρκετά μεγάλο μέγεθος έτσι ώστε να χωρούν τα προεπιλεγμένα περιεχόμενα. Για να προσθέσετε περισσότερο χώρο γύρω από το περιεχόμενο στα κελιά του πίνακα, μπορεί να χρησιμοποιηθεί η ιδιότητα padding στην CSS.

Τα περιγράμματα γύρω από τον πίνακα και τα κελιά τους διαχωρίζονται το ένα από το άλλο, από προεπιλογή. Ωστόσο μπορούν να συμπτυχθούν σε ένα περίγραμμα χρησιμοποιώντας την ιδιότητα border-collapse με το στοιχείο <table>. Επιπλέον, το κείμενο εντός των στοιχείων <th> εμφανίζεται με έντονη γραμματοσειρά, στοιχισμένο οριζόντια στο κέντρο του κελιού από προεπιλογή. Για να αλλάξετε την προεπιλεγμένη στοίχιση, μπορεί να χρησιμοποιηθεί η ιδιότητα CSS text-align. Για να αλλάξετε την προεπιλεγμένη στοίχιση, μπορείτε να χρησιμοποιήσετε την ιδιότητα στοίχισης κειμένου (text-align) στην CSS. Για να μπορέσουν να το κάνουν αυτό οι εκπαιδευόμενοι, θα ήταν καλύτερα πρώτα να φτάσουν στην ενότητα που καλύπτει την CSS. Τα περισσότερα χαρακτηριστικά του στοιχείου <table>, όπως border, cellpadding , cellspacing, width,

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

align, κ.λπ. για τις αλλαγές στην εμφάνιση ενός πίνακα, τα οποία υπήρχαν σε προηγούμενες εκδόσεις της HTML, έχουν απορριφθεί στην HTML5, επομένως θα πρέπει να αποφεύγονται. **Για την τροποποίηση της εμφάνισης ενός πίνακα, συστήνεται η χρήση της CSS.**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Creating Tables in HTML</title>
</head>
<body>
    <h2>Spotify Top Songs of 2021 (USA)</h2>
    <table>
        <tr>
            <th>No.</th>
            <th>Song - Band </th>
            <th>Lenght</th>
        </tr>
        <tr>
            <td>1</td>
            <td>drivers licence - Olivia Rodrigo</td>
            <td>4:02</td>
        </tr>
        <tr>
            <td>2</td>
            <td>MONTERO (Call me by your name) - Lil Nas X</td>
            <td>2:17</td>
        </tr>
        <tr>
            <td>3</td>
            <td>STAY - The Kid LAROI ft. Justin Bieber</td>
            <td>2:21</td>
        </tr>
    </table>
</body>
</html>
```

Spotify Top Songs of 2021 (USA)

No.	Song - Band	Lenght
1	drivers licence - Olivia Rodrigo	4:02
2	MONTERO (Call me by your name) - Lil Nas X	2:17
3	STAY - The Kid LAROI ft. Justin Bieber	2:21

Εικόνα 43 *– Ένας βασικός πίνακας (Πηγή: Συντάκτης)*

Επέκταση πολλαπλών γραμμών και στηλών.

Η επέκταση επιτρέπει την επέκταση σειρών και στηλών ενός πίνακα σε πολλές άλλες σειρές και στήλες. Συνήθως, ένα κελί σε έναν πίνακα δεν μπορεί να περάσει πάνω ή κάτω από άλλο κελί του πίνακα. Ωστόσο, τα χαρακτηριστικά rowspan ή colspan μπορούν να χρησιμοποιηθούν για την επέκταση πολλαπλών σειρών ή στηλών σε έναν πίνακα.

Ομοίως, το χαρακτηριστικό rowspan μπορεί να χρησιμοποιηθεί για τη δημιουργία ενός κελιού που εκτείνεται σε περισσότερες από μία σειρές, ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Span Multiple Rows in an HTML Table</title>
    <style>
        table {
            width: 300px;
            border-collapse: collapse;
        }
        table, th, td {
            border: 1px solid black;
        }
        th, td {
            padding: 10px;
        }
    </style>
</head>
<body>
    <h2>Spanning Rows</h2>
    <table>
        <tr>
            <th>Name:</th>
            <td>John Carter</td>
        </tr>
        <tr>
            <th rowspan="2">Phone:</th>
            <td>55577854</td>
        </tr>
        <tr>
            <td>55577855</td>
        </tr>
    </table>
</body>
</html
```

Spanning Rows

Name:	John Carter
Phone:	55577854
	55577855

Εικόνα 44 *– Το χαρακτηριστικό rowspan (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-tables.php)*

Λεζάντες στους πίνακες

Μια λεζάντα (ή τίτλος) ενός πίνακα μπορεί να δημιουργηθεί χρησιμοποιώντας το στοιχείο <caption>. Αυτό το στοιχείο θα πρέπει να τοποθετηθεί αμέσως μετά την ετικέτα <table> (στην αρχή). Η λεζάντα εμφανίζεται στην κορυφή του πίνακα από προεπιλογή, ωστόσο αυτό μπορεί να αλλάξει χρησιμοποιώντας την ιδιότητα caption-side της CSS.

```
<!DOCTYPE html>
<html>
   <body>
      <table border=1>
       <caption> WIKITECHY WEBSITE </caption>
       <tr>
           <th>Firstname</th>
           <th>Lastname</th>
       </tr>
       <tr>
           <td>Wiki</td>
           <td>techy</td>
       </tr>
      </table>
   </body>
</html>
```

WIKITECHY WEBSITE

Firstname	Lastname
Wiki	techy

Εικόνα 45 *– Προσθήκη λεζάντας πίνακα (Πηγή: https://www.wikitechy.com)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Καθορισμός κεφαλίδας, κυρίως κειμένου και υποσέλιδου του πίνακα

Η HTML παρέχει ορισμένες ετικέτες όπως, <thead>, <tbody>, και <tfoot>, οι οποίες βοηθούν τους εκπαιδευόμενους να δημιουργήσουν πιο οργανωμένους πίνακες, χωρίζοντάς τους μεταξύ κεφαλίδας, κυρίως κειμένου και υποσέλιδου, με αυτή τη σειρά.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Table with a Header, Footer and Body</title>
    <style>
        table {
            width: 300px;
            border-collapse: collapse;
        }
        table, th, td {
            border: 1px solid black;
        }
        th, td {
            padding: 10px;
            text-align: left;
        }
    </style>
</head>
<body>
    <table>
        <thead>
            <tr>
                <th>Items</th>
                <th>Expenditure</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>Stationary</td>
                <td>2,000</td>
            </tr>
            <tr>
                <td>Furniture</td>
                <td>10,000</td>
            </tr>
        </tbody>
        <tfoot>
            <tr>
                <th>Total</th>
                <td>12,000</td>
            </tr>
        </tfoot>
    </table>
</body>
</html
```

Items	Expenditure
Stationary	2,000
Furniture	10,000
Total	12,000

Εικόνα 45 *– Προσθήκη λεζάντας σε έναν πίνακα (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-tables.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Λίστες στην HTML

Οι λίστες HTML εφαρμόζονται για την παρουσίαση πληροφοριών με καλοσχηματισμένο και σημασιολογικό τρόπο. Μέσα σε αυτά, μπορεί κανείς να προσθέσει κείμενο, εικόνες, συνδέσμους, αλλαγές γραμμής κ.λπ. Υπάρχουν τρεις διαφορετικοί τύποι λιστών σε HTML και ο καθένας έχει συγκεκριμένο σκοπό και σημασία:

- **Α) Μη Διατεταγμένες λίστες** — Χρησιμοποιείται για τη δημιουργία λίστας σχετικών στοιχείων, χωρίς συγκεκριμένη σειρά.
- **Β) Διατεταγμένες λίστες** — Χρησιμοποιείται για τη δημιουργία λίστας σχετικών στοιχείων, με συγκεκριμένη σειρά.
- **Γ) Λίστες περιγραφών** — Χρησιμοποιούνται για τη δημιουργία λίστας με όρους και των περιγραφών τους.

Α) Μη Διατεταγμένες Λίστες

Μια Μη Διατεταγμένη Λίστα δημιουργείται χρησιμοποιώντας το στοιχείο <ul> και κάθε στοιχείο της λίστας ξεκινά με το στοιχείο <li>. Σημειώνεται με κουκκίδες, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Reasons why you should travel by train</title>
</head>
<body>
    <h2>Reasons why you should travel by train</h2>
    <ul>
        <li>It is less expensive</li>
        <li>It is eco-friendly</li>
        <li>You can enjoy the view</li>
    </ul>
</body>
</html>
```

Reasons why you should travel by train

- It is less expensive
- It is eco-friendly
- You can enjoy the view

Εικόνα 46 *– Μη διατεταγμένες λίστες (Πηγή: Συντάκτης)*

Β) Διατεταγμένες Λίστες

Μια Διατεταγμένη Λίστα δημιουργείται χρησιμοποιώντας το στοιχείο <ol> στοιχείο και κάθε στοιχείο της λίστας ξεκινά με το στοιχείο <li> στοιχείο. Οι διατεταγμένες

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

λίστες χρησιμοποιούνται όταν η σειρά των στοιχείων της λίστας παίζει καθοριστικό ρόλο. Σημειώνεται με αριθμούς, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Ordered List</title>
</head>
<body>
    <h2>How to cook your own veggie burger</h2>
    <ol>
        <li>Dump your ground meat into a bowl. (We go for ground meat with
around 20% fat.) Season it with salt, pepper, and whatever else you want; you can
add spices, perhaps, or Worcestershire sauce, or shallots, or chiles.</li>
        <li>Shape your burgers into patties, using your thumb to make an
indentation in the center; this will keep the burgers from puffing up. Keep in
mind that the burgers will shrink up a bit once you cook them, so make your
patties a bit bigger than you want them later.</li>
        <li>Oil your grill or a cast-iron pan, and grill or sear those patties.
(How many times to flip them is up for debate -- but when I'm grilling, I flip
once so I can get get those nice grill marks.) Cook them until your desired
doneness (around 125-130ºF for medium rare, around 1 minute per side for each
inch of thickness). But before you take them off the grill...</li>
      <li>...add your cheese and toast your buns. Let the cheese melt while the
burgers are still on the grill; to speed things up, you can close the cover.
</li>
<li>Once your burgers iare finished cooking, and your cheese is melty and your
buns are nicely charred, throw some condiments and toppings on those burgers.
Anything goes. (Really, anything goes.) Bite into it and let those juices run
down your chin, and rejoice that it's summer. And then make another round,
because now you know how.</li>
    </ol>
</body>
</html>
```

How to cook your own veggie burger

1. Dump your ground meat into a bowl. (We go for ground meat with around 20% fat.) Season it with salt, pepper, and whatever else you want; you can add spices, perhaps, or Worcestershire sauce, or shallots, or chiles.
2. Shape your burgers into patties, using your thumb to make an indentation in the center; this will keep the burgers from puffing up. Keep in mind that the burgers will shrink up a bit once you cook them, so make your patties a bit bigger than you want them later.
3. Oil your grill or a cast-iron pan, and grill or sear those patties. (How many times to flip them is up for debate -- but when I'm grilling, I flip once so I can get get those nice grill marks.) Cook them until your desired doneness (around 125-130ºF for medium rare, around 1 minute per side for each inch of thickness). But before you take them off the grill...
4. ...add your cheese and toast your buns. Let the cheese melt while the burgers are still on the grill; to speed things up, you can close the cover.
5. Once your burgers iare finished cooking, and your cheese is melty and your buns are nicely charred, throw some condiments and toppings on those burgers. Anything goes. (Really, anything goes.) Bite into it and let those juices run down your chin, and rejoice that it's summer. And then make another round, because now you know how.

Εικόνα 47 *– Διατεταγμένες λίστες (Πηγή: Συντάκτης)*

Γ) Λίστες περιγραφής

Μια λίστα περιγραφής είναι μια λίστα στοιχείων με την περιγραφή ή τον ορισμό κάθε στοιχείου.

Η λίστα περιγραφής δημιουργείται χρησιμοποιώντας το στοιχείο <dl>. Το στοιχείο <dl> χρησιμοποιείται σε συνδυασμό με το στοιχείο <dt> που προσδιορίζει έναν όρο και το στοιχείο <dd> που προσδιορίζει τον ορισμό του όρου.

Τα προγράμματα περιήγησης συνήθως αποδίδουν τις λίστες ορισμών τοποθετώντας τους όρους και τους ορισμούς σε ξεχωριστές γραμμές, όπου οι ορισμοί του όρου έχουν ελαφρά εσοχή.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Description or Definition List</title>
</head>
<body>
    <h2>What are bread and coffee?</h2>
    <dl>
        <dt>Bread</dt>
        <dd>A baked food made of flour.</dd>
        <dt>Coffee</dt>
        <dd>A drink made from roasted coffee beans.</dd>
    </dl>
</body>
</html>
```

What are bread and coffee?

Bread
A baked food made of flour.
Coffee
A drink made from roasted coffee beans.

Εικόνα 48 *– Λίστες περιγραφών (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Φόρμες στην HTML

Οι Φόρμες στην HTML στοχεύουν στη συλλογή διαφορετικών τύπων εισροών χρηστών, όπως στοιχεία επικοινωνίας (όνομα, email, αριθμός τηλεφώνου, τραπεζικός λογαριασμός κ.λπ.). Οι φόρμες περιλαμβάνουν ειδικά στοιχεία γνωστά ως στοιχεία ελέγχου, όπως πλαίσιο εισαγωγής, πλαίσια ελέγχου, κουμπιά επιλογής, κουμπιά υποβολής κ.λπ. Για το σκοπό αυτό, οι χρήστες συμπληρώνουν μια φόρμα με αυτά τα δεδομένα, μέσω επιλογής κειμένου ή πλαισίων, υποβάλλοντάς την αργότερα σε έναν διακομιστή ιστού για την επεξεργασία αυτών των δεδομένων.

Η ετικέτα <form> χρησιμοποιείται για τη δημιουργία μιας φόρμας HTML. Ένα απλό παράδειγμα μιας φόρμας σύνδεσης θα ήταν το εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Simple HTML Form</title>
</head>
<body>
    <form action="/examples/actions/confirmation.php" method="post">
        <label>Username: <input type="text" name="username"></label>
        <label>Password: <input type="password" name="userpass"></label>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Username: Password: Submit

Εικόνα 49 *– Παράδειγμα μιας φόρμας σύνδεσης (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-forms.php)*

Υπάρχουν διάφοροι τύποι στοιχείων ελέγχου που πρέπει να εφαρμόζονται σε μια φόρμα HTML, με τα πιο συνηθισμένα να είναι **τα στοιχεία εισόδου**. Προσδιορίζουν διάφορους τύπους πεδίων εισαγωγής χρήστη, ανάλογα με το χαρακτηριστικό type. Τα στοιχεία εισαγωγής μπορεί να είναι **πεδία κειμένου, πεδία κωδικού πρόσβασης, πλαίσια ελέγχου, κουμπιά υποβολής, κουμπιά επαναφοράς, πλαίσια επιλογής αρχείων**, καθώς και αρκετοί νέοι τύποι εισαγωγής που εισάγονται στην HTML5 (μπορείτε να τους ρίξετε μια ματιά εδώ).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τα **πεδία κειμένου** είναι τα πεδία που επιτρέπουν στους χρήστες να προσθέτουν κείμενο. Αυτά δημιουργούνται χρησιμοποιώντας ένα στοιχείο <input>, του οποίου το χαρακτηριστικό type έχει μια τιμή text. Θα πρέπει να σημειωθεί ότι η ετικέτα <label> χρησιμοποιείται για την αναγνώριση των ετικετών για τα στοιχεία <input>. Εάν ο διαχειριστής του ιστοχώρου θέλει οι χρήστες να εισάγουν πολλές γραμμές, θα πρέπει να προστεθεί η ετικέτα <textarea> αντί της <label>.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Text Input Field</title>
</head>
<body>
    <form>
        <label for="user-name">Login:</label>
        <input type="text" name="username" id="user-name">
    </form>
</body>
</html>
```

Login:

Εικόνα 50 *– Παράδειγμα πεδίου κειμένου (Πηγή: Συντάκτης)*

Τα **πεδία κωδικού πρόσβασης** είναι σαν πεδία κειμένου, με τη μόνη διαφορά να είναι ότι οι χαρακτήρες σε ένα πεδίο κωδικού πρόσβασης είναι κρυμμένοι, και επομένως εμφανίζονται ως αστερίσκοι ή κουκκίδες. Επιπλέον, αυτή η διαδικασία εμποδίζει κάποιον τρίτο από το να διαβάσει τον κωδικό πρόσβασης στην οθόνη. Αυτό είναι επίσης ένα στοιχείο ελέγχου εισαγωγής κειμένου μιας γραμμής που δημιουργείται χρησιμοποιώντας ένα στοιχείο <input> του οποίου το χαρακτηριστικό type έχει μια τιμή password.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Password Input Field</title>
</head>
<body>
    <form>
        <label for="user-pwd">Password:</label>
        <input type="password" name="user-password" id="user-pwd">
    </form>
</body>
</html>
```

Password:

Εικόνα 51 *– Παράδειγμα πεδίου κωδικού πρόσβασης (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-forms.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τα **κουμπιά επιλογής** εφαρμόζονται για να επιτρέπουν στον χρήστη να επιλέξει ακριβώς μία επιλογή από ένα προκαθορισμένο σύνολο επιλογών. Δημιουργείται χρησιμοποιώντας ένα στοιχείο <input> του οποίου το χαρακτηριστικό type έχει την τιμή του κουμπιού επιλογής.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Radio Buttons</title>
</head>
<body>
    <form>
        <input type="radio" name="Civil Status" value="single" id="single">
        <label for="single">Single</label>
        <input type="radio" name="Civil Status" value="married" id="married">
        <label for="married">Married</label>
      <input type="radio" name="Civil Status" value="other" id="other">
        <label for="other">Other</label>
    </form>
</body>
</html>
```

Single Married Other

Εικόνα 52 *– Κώδικας για τη ρύθμιση των κουμπιών επιλογής (Πηγή: Συντάκτης)*

Τα **πλαίσια ελέγχου** παρέχουν στον χρήστη μία ή περισσότερες επιλογές από ένα προκαθορισμένο σύνολο επιλογών. Δημιουργείται χρησιμοποιώντας ένα στοιχείο <input> του οποίου το χαρακτηριστικό type έχει την τιμή του checkbox. Αν κάποιος προτιμά να δημιουργήσει ένα πλαίσιο ελέγχου (ή κουμπί επιλογής) που να είναι προεπιλεγμένο, πρέπει απλώς να προσθέσει το χαρακτηριστικό checked στο στοιχείο εισαγωγής (<input type="checkbox" checked>).

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Checkboxes</title>
</head>
<body>
    <form>
        <input type="checkbox" name="sports" value="football" id="football">
        <label for="football">Football</label>
        <input type="checkbox" name="sports" value="handball" id="handball">
        <label for="handball">Handball</label>
        <input type="checkbox" name="sports" value="basketball" id="basketball">
        <label for="basketball">Basketball</label>
    </form>
</body>
</html>
```

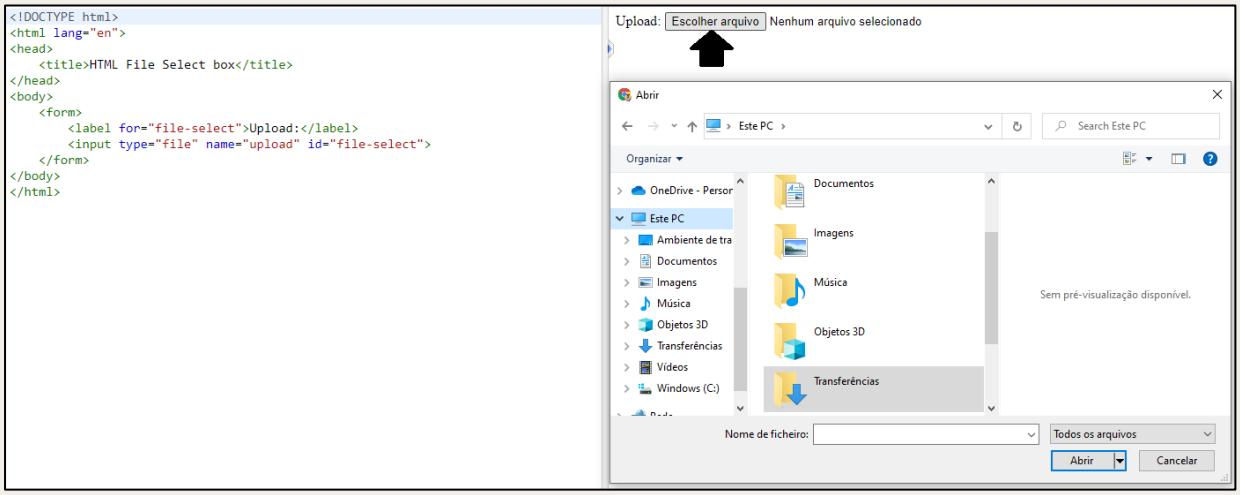
Football Handball Basketball

Εικόνα 53 *– Κωδικός για τη ρύθμιση πλαισίων ελέγχου (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

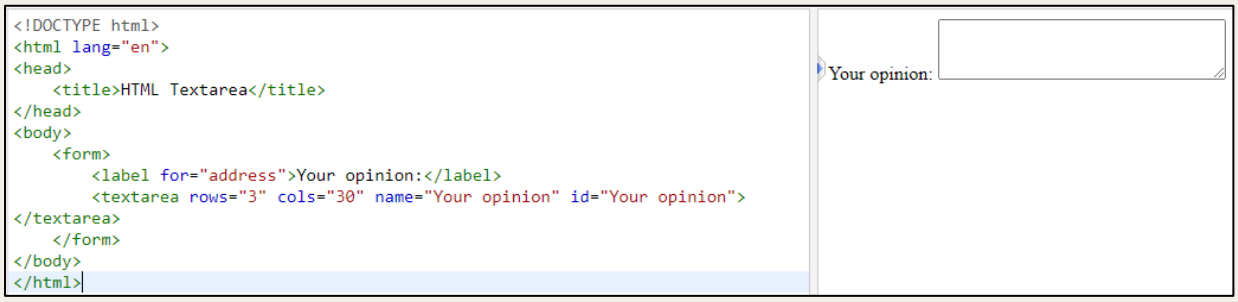
Τα **πλαίσια επιλογής αρχείου** επιτρέπουν σε έναν χρήστη να αναζητήσει ένα τοπικό αρχείο και να το στείλει ως συνημμένο με τα δεδομένα της φόρμας. Π.χ., το Google Chrome παρέχει ένα πεδίο επιλογής για την εισαγωγή αρχείου με το κουμπί «Αναζήτηση» που επιτρέπει στους χρήστες να επιλέξουν ένα αρχείο από τον σκληρό τους δίσκο.

Τα πλαίσια επιλογής αρχείου δημιουργούνται επίσης χρησιμοποιώντας ένα στοιχείο <input>, του οποίου η τιμή του χαρακτηριστικού type έχει οριστεί σε file.



Εικόνα 54 *– Κώδικας για τη ρύθμιση των πλαισίων επιλογής αρχείων (Πηγή: Συντάκτης)*

Το Textarea μπορεί να οριστεί ως ένα στοιχείο ελέγχου εισαγωγής κειμένου πολλαπλών γραμμών που επιτρέπει την εισαγωγή περισσότερων από μίας γραμμών κειμένου. Αυτά τα στοιχεία ελέγχου δημιουργούνται χρησιμοποιώντας ένα στοιχείο <textarea> , ως εξής:



Εικόνα 55 *– Κωδικός για τη ρύθμιση ενός στοιχείου ελέγχου εισαγωγής κειμένου πολλαπλών γραμμών (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τα **πλαίσια επιλογής** είναι αναπτυσσόμενες λίστες επιλογών στις οποίες ένας χρήστης μπορεί να επιλέξει μία ή περισσότερες επιλογές από ένα αναπτυσσόμενο μενού. Δημιουργούνται με τη χρήση του στοιχείου <select> και του στοιχείου <option>. Τα στοιχεία <option> εντός του στοιχείου <select > ορίζουν το κάθε στοιχείο της λίστας.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Select Box</title>
</head>
<body>
    <form>
        <label for="country">Country:</label>
        <select name="country" id="country">
            <option value="Portugal">Portugal</option>
            <option value="Cyprus">Cyprus</option>
            <option value="Greece">Greece</option>
        </select>
    </form>
</body>
</html>
```

Country: Portugal
Portugal
Cyprus
Greece

Εικόνα 56 *– Κώδικας για τη ρύθμιση των πλαισίων επιλογής αρχείων (Πηγή: Συντάκτης)*

Τα **κουμπιά υποβολής και επαναφοράς** είναι πολύ συνηθισμένα στους περισσότερους ιστότοπους. Τα κουμπιά υποβολής χρησιμοποιούνται για την αποστολή δεδομένων (φόρμας) σε έναν διακομιστή ιστού, ενώ τα κουμπιά επαναφοράς δημιουργούνται για την επαναφορά της φόρμας στις προεπιλεγμένες τιμές. Όταν ο χρήστης κάνει κλικ στο κουμπί υποβολή, τα δεδομένα της φόρμας αποστέλλονται στο αρχείο που καθορίζεται στο χαρακτηριστικό action της φόρμας για την επεξεργασία των δεδομένων που υποβλήθηκαν.

Τα κουμπιά υποβολής ή επαναφοράς μπορούν επίσης να δημιουργηθούν χρησιμοποιώντας το στοιχείο </button>. Έχουν τον ίδιο σκοπό με τα κουμπιά που δημιουργούνται με το στοιχείο <input>. Ωστόσο, προσφέρουν περισσότερες δυνατότητες απόδοσης, καθώς επιτρέπουν την ενσωμάτωση άλλων στοιχείων της HTML.

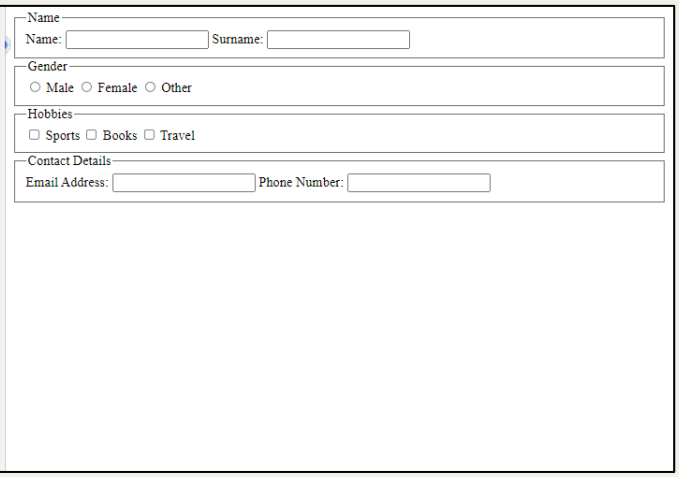
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>HTML Submit and Reset Buttons</title>
</head>
<body>
    <form action="/examples/html/action.php" method="post">
        <label for="first-name">First Name:</label>
        <input type="text" name="first-name" id="first-name">
        <input type="submit" value="Submit">
        <input type="reset" value="Reset">
    </form>
</body>
</html>
```

Εικόνα 57 *– Κωδικός για τη ρύθμιση των κουμπιών υποβολής και επαναφοράς (Πηγή: Συντάκτης)*

Τα **στοιχεία ομαδοποίησης ελέγχου φόρμας** είναι ένα εξαιρετικό εργαλείο για τους χρήστες που τους βοηθούν να εντοπίσουν ένα στοιχείο ελέγχου, καθιστώντας τη φόρμα πιο εύκολα προσβάσιμη. Το στοιχείο < legend > είναι το κλειδί για τη δημιουργία λογικά σχετικών στοιχείων ελέγχου, όπως φαίνεται στην παρακάτω εικόνα:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Grouping Form Controls in HTML</title>
</head>
<body>
    <form>
        <fieldset>
            <legend>Name</legend>
            <label>Name: <input type="text" name="firstname"></label>
            <label>Surname: <input type="text" name="lastname"></label>
        </fieldset>
        <fieldset>
            <legend>Gender</legend>
            <label><input type="radio" name="gender" value="male"> Male</label>

            <label><input type="radio" name="gender" value="female"> Female</label>
          <label><input type="radio" name="gender" value="other"> Other</label>
        </fieldset>
        <fieldset>
            <legend>Hobbies</legend>
            <label><input type="checkbox" name="hobbies" value="sports"> Sports</label>
            <label><input type="checkbox" name="hobbies" value="culture"> Books</label>
            <label><input type="checkbox" name="hobbies" value="leisure"> Travel</label>
        </fieldset>
        <fieldset>
            <legend>Contact Details</legend>
            <label>Email Address: <input type="email" name="email"></label>
            <label>Phone Number: <input type="text" name="phone"></label>
        </fieldset>
    </form>
</body>
</html>
```



Εικόνα 58 *– Κώδικας για τη ρύθμιση των στοιχείων ελέγχου φορμών ομαδοποίησης (Πηγή: Συντάκτης, ρίξτε μια ματιά εδώ για καλύτερο ορισμό)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ακολουθεί μια λίστα με τα πιο συνηθισμένα χαρακτηριστικά χρησιμοποιούνται στις φόρμες:

Χαρακτηριστικό	Περιγραφή
name	Καθορίζει το όνομα της φόρμας.
action	Καθορίζει τη διεύθυνση URL του προγράμματος ή του αρχείου εντολών στον διακομιστή Παγκόσμιου Ιστού που θα χρησιμοποιηθεί για την επεξεργασία των πληροφοριών που υποβάλλονται μέσω της φόρμας.
method	Καθορίζει τη μέθοδο HTTP που χρησιμοποιείται για την αποστολή των δεδομένων στον διακομιστή Παγκόσμιου Ιστού από το πρόγραμμα περιήγησης. Η τιμή μπορεί να είναι είτε get (προεπιλεγμένη) είτε post.
target	Καθορίζει πού θα εμφανίζεται η απάντηση που λαμβάνεται μετά την υποβολή της φόρμας. Πιθανές τιμές είναι _blank, _self, _parent και _top.
enctype	Καθορίζει τον τρόπο με τον οποίο θα πρέπει να κωδικοποιούνται τα δεδομένα της φόρμας κατά την υποβολή της φόρμας στον διακομιστή. Ισχύει μόνο όταν η τιμή του χαρακτηριστικού method είναι post.

***Πίνακας 2** – Λίστα χαρακτηριστικών στοιχείων φόρμας που χρησιμοποιούνται συχνά (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-forms.php)*

Περισσότερες πληροφορίες σχετικά με άλλα χαρακτηριστικά μπορείτε να βρείτε εδώ.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ενσωματωμένο πλαίσιο iFrame στην HTML

Ένα iframe (*ή ενσωματωμένο πλαίσιο*) χρησιμοποιείται για την έκθεση εξωτερικών παραγόντων μέσα σε μια ιστοσελίδα, συμπεριλαμβανομένων άλλων ιστοσελίδων. Ένα iframe λειτουργεί σχεδόν σαν ένα μίνι πρόγραμμα περιήγησης ιστού μέσα σε ένα κανονικό πρόγραμμα περιήγησης ιστού. Ομοίως, το περιεχόμενο μέσα σε ένα iframe εμφανίζεται χωριστά από τα προσκείμενα στοιχεία.

Η βασική σύνταξη για την προσθήκη αυτής της δυνατότητας σε μια ιστοσελίδα είναι η εξής:

```
<iframe src="URL"></iframe>
```

Η διεύθυνση URL που καθορίζεται στο χαρακτηριστικό src υποδεικνύει τη θέση ενός εξωτερικού παράγοντα ή μιας ιστοσελίδας.

Εικόνα 59 *– Κώδικας για τη ρύθμιση ενός ενσωματωμένου πλαισίου (Πηγή: Συντάκτης)*

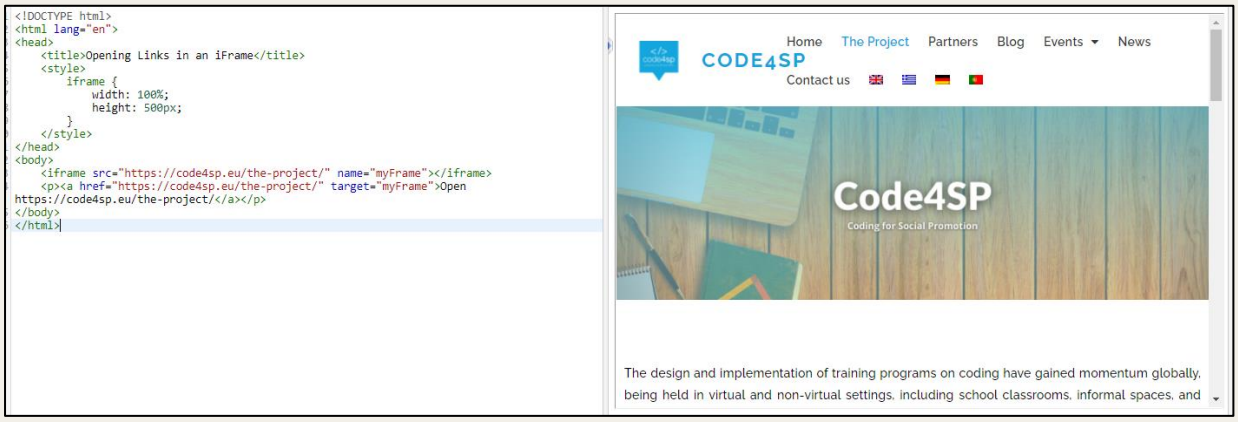
Το **ύψος** και το **πλάτος** ενός iframe μπορούν να καθοριστούν εφαρμόζοντας τον κώδικα που φαίνεται στην Εικόνα παρακάτω. Οι τιμές των χαρακτηριστικών width (πλάτος) και height (ύψος) προκαθορίζονται σε pixel, αλλά οι χρήστες μπορούν επίσης να ορίσουν αυτές τις τιμές σε ποσοστά, όπως 50%, 100%, κ.λπ.. Το προεπιλεγμένο πλάτος ενός iframe είναι 300 pixel, ενώ το προεπιλεγμένο ύψος είναι 150 pixel.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εικόνα 60 *– Κωδικός για τη ρύθμιση ύψους και πλάτους ενός inline frame (Πηγή: Συγγραφέας)*

Όπως φαίνεται από την εικόνα, το iframe έχει ένα περίγραμμα γύρω του που έχει καθοριστεί από προεπιλεγμένες ρυθμίσεις. Ο καλύτερος τρόπος για να το τροποποιήσετε ή να το αφαιρέσετε, είναι να χρησιμοποιήσετε την ιδιότητα border στην CSS (που θα διδαχθείτε στην ενότητα CSS).

Ένα iframe μπορεί επίσης να χρησιμοποιηθεί ως στοχευμένος σύνδεσμος για τους υπερσυνδέσμους. Μπορεί να ονομαστεί χρησιμοποιώντας το χαρακτηριστικό name. Αυτό σημαίνει ότι όταν κάνετε κλικ σε έναν σύνδεσμο με ένα χαρακτηριστικό target (με αυτή την ονομασία ορισμένη ως τιμή), η συνδεδεμένη πηγή θα ανοίξει στο ίδιο ενσωματωμένο πλαίσιο, όπως φαίνεται στην εικόνα:



Εικόνα 61 *– Χρήση του iframe ως στοχευμένου συνδέσμου (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

2.2. Προηγμένες έννοιες της HTML

Doctypes στην HTML

Το Document Type Declaration (DOCTYPE) είναι μια οδηγία προς το πρόγραμμα περιήγησης ιστού σχετικά με την έκδοση της γλώσσας σήμανσης στην οποία δημιουργείται μια ιστοσελίδα. Εμφανίζεται στην κορυφή μιας ιστοσελίδας πριν από όλα τα άλλα στοιχεία. Σύμφωνα με τις προδιαγραφές της HTML, κάθε έγγραφο HTML απαιτεί μια έγκυρη δήλωση τύπου εγγράφου (doctype) για να διασφαλιστεί ότι οι ιστοσελίδες έχουν την κατάλληλη μορφή. Η δήλωση doctype είναι συνήθως το πρώτο πράγμα που ορίζεται σε ένα έγγραφο HTML (ακόμα και πριν από την ετικέτα αρχής <html>). Ωστόσο, η ίδια η δήλωση doctype δεν είναι ετικέτα της HTML.

Το DOCTYPE στην HTML5 είναι πολύ σύντομο, συνοπτικό και χωρίς διάκριση πεζών-κεφαλαίων: <!DOCTYPE html> .

Η ακόλουθη σήμανση μπορεί να χρησιμοποιηθεί ως πρότυπο για τη δημιουργία ενός νέου εγγράφου HTML5:

```
<!DOCTYPE html> <html lang="en"> <head> <meta charset="utf-8">
<title><!-- Insert your title here --></title> </head> <body> <!--
Insert your content here --> </body> </html>
```

Διατάξεις στην HTML (Layouts)

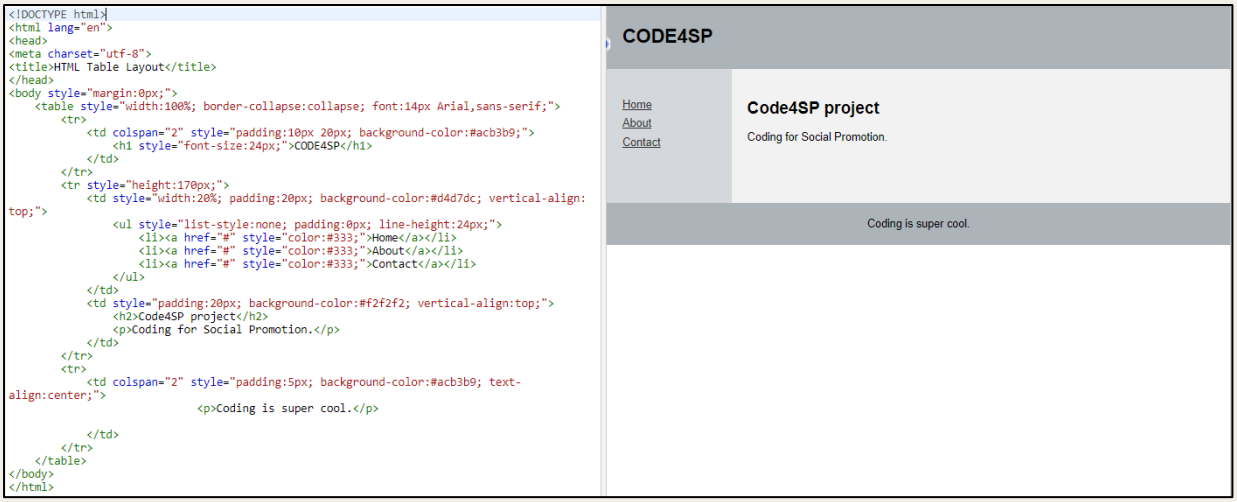
Υπάρχουν διάφορες μέθοδοι για τη δημιουργία της διάταξης μιας ιστοσελίδας, την τοποθέτηση των διαφόρων στοιχείων που συνθέτουν μια ιστοσελίδα με καλά δομημένο τρόπο, δίνοντας ταυτόχρονα μια ελκυστική εμφάνιση στον ιστότοπο. Οι περισσότεροι ιστότοποι συνήθως εμφανίζουν το περιεχόμενό τους σε πολλές σειρές και στήλες, διαμορφωμένες σαν περιοδικό για να παρέχουν στους χρήστες μια καλύτερη ατμόσφαιρα ανάγνωσης και γραφής. Αυτό μπορεί να επιτευχθεί εύκολα χρησιμοποιώντας τις ετικέτες της HTML, όπως π.χ< table > ,< div > ,< header > ,<

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

footer>,< section > , κ.λπ. και συνδυάζοντας ορισμένα στυλ της CSS με τις ετικέτες αυτές.

Ο απλούστερος τρόπος για τη δημιουργία διατάξεων στην HTML επιτυγχάνεται με το σχεδιασμό πινάκων. Όπως φαίνεται στις προηγούμενες ενότητες, αυτό συνήθως περιλαμβάνει τη διαδικασία τοποθέτησης περιεχομένων (κείμενο, εικόνες κ.λπ.) σε σειρές και στήλες.

Η παρακάτω διάταξη περιέχει έναν πίνακα HTML με τρεις σειρές και δύο στήλες. Θα πρέπει να σημειωθεί ότι η πρώτη και η τελευταία σειρά εκτείνονται ενώ χρησιμοποιούν και οι δύο το χαρακτηριστικό colspan. Αξίζει να αναφερθεί ότι η μέθοδος που χρησιμοποιείται για τη δημιουργία διάταξης σε αυτό το παράδειγμα, αν και δεν είναι λάθος, δεν συνιστάται. Οι πίνακες και τα ενσωματωμένα στυλ για τη δημιουργία διατάξεων θα πρέπει να αποφεύγονται. Οι διατάξεις που δημιουργούνται με τη χρήση πινάκων συχνά αποδίδονται πολύ αργά. **Οι πίνακες πρέπει να χρησιμοποιούνται μόνο για την εμφάνιση δεδομένων του πίνακα.** Για τη δημιουργία τέτοιων διατάξεων, **συνίσταται η χρήση των ιδιοτήτων επίπλευσης της CSS (float techniques)**. Οι χρήστες θα μάθουν περισσότερα για αυτό το εργαλείο αργότερα.



Εικόνα 62 *– Βασική διάταξη HTML (Πηγή: Συντάκτης, προσαρμοσμένο από το Tutorial Republic. Κάντε κλικ εδώ για καλύτερη ανάλυση)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στην HTML5 έχουν εισαχθεί νέα δομικά στοιχεία, όπως το <header>, <footer>, <nav>, <section>, κ.λπ., έτσι ώστε να μπορεί ο χρήστης να εντοπίζει τα διαφορετικά μέρη μιας ιστοσελίδας πιο εύκολα. Αυτό το παράδειγμα εφαρμόζει τα νέα δομικά στοιχεία HTML5 για τη δημιουργία της ίδιας διάταξης που δημιουργήθηκε στην *Εικόνα 62*.

Για να μάθετε περισσότερα σχετικά με τις ετικέτες που αναφέρονται πιο πάνω, μπορείτε να επισκεφτείτε αυτόν τον σύνδεσμο.

Επικεφαλίδα στην HTML (Head)

Το στοιχείο head είναι η υποδοχή για όλα τα στοιχεία της επικεφαλίδας, τα οποία παρέχουν επιπλέον πληροφορίες για το έγγραφο ή παραπομπή σε άλλους πόρους, που απαιτούνται για τη σωστή απόδοση του εγγράφου. Παρουσιάζει τις ιδιότητες του εγγράφου, όπως τον τίτλο, την παράδοση μετα-πληροφοριών, όπως το σύνολο χαρακτήρων, δείχνει στο πρόγραμμα περιήγησης πού να βρει τα φύλλα στυλ ή τα σενάρια που επιτρέπουν την διαδραστική επέκταση του εγγράφου HTML.

Τα στοιχεία HTML που μπορούν να χρησιμοποιηθούν εντός του στοιχείου <head> είναι: <title>, <base>, <link>, <style>, <meta>, <script> και <noscript>.

Το στοιχείο title στην HTML (τίτλος)

Το στοιχείο <title>προσδιορίζει τον τίτλο του εγγράφου και απαιτείται μόνο ένα σε όλα τα έγγραφα HTML/XHTML για την παραγωγή ενός έγκυρου εγγράφου. Πρέπει να τοποθετηθεί μέσα στο στοιχείο < head >. Το στοιχείο <title> περιλαμβάνει απλό κείμενο και περιεχόμενο και ενδέχεται να μην περιλαμβάνει άλλες ετικέτες σήμανσης. Μπορεί να χρησιμοποιηθεί για διαφορετικές λειτουργίες όπως:

- Για να εμφανιστεί ένας τίτλος στη γραμμή τίτλου του προγράμματος περιήγησης και στη γραμμή εργασιών.
- Για να δοθεί ένας τίτλος στη σελίδα για όταν προστίθεται στα αγαπημένα ή στους σελιδοδείκτες.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Για να δοθεί ένας τίτλος στη σελίδα που θα εμφανίζεται σε αποτελέσματα μηχανών αναζήτησης (π.χ. αναζήτηση Google).

Θα πρέπει να είναι **σύντομο** και **συγκεκριμένο** για το περιεχόμενο του εγγράφου, καθώς οι μηχανές αναζήτησης δίνουν ιδιαίτερη προσοχή στις λέξεις που υπάρχουν στον τίτλο, γι' αυτό ιδανικά θα έχει 65 χαρακτήρες.

Ένας τίτλος σε ένα έγγραφο HTML θα πρέπει να εμφανίζεται ως εξής:

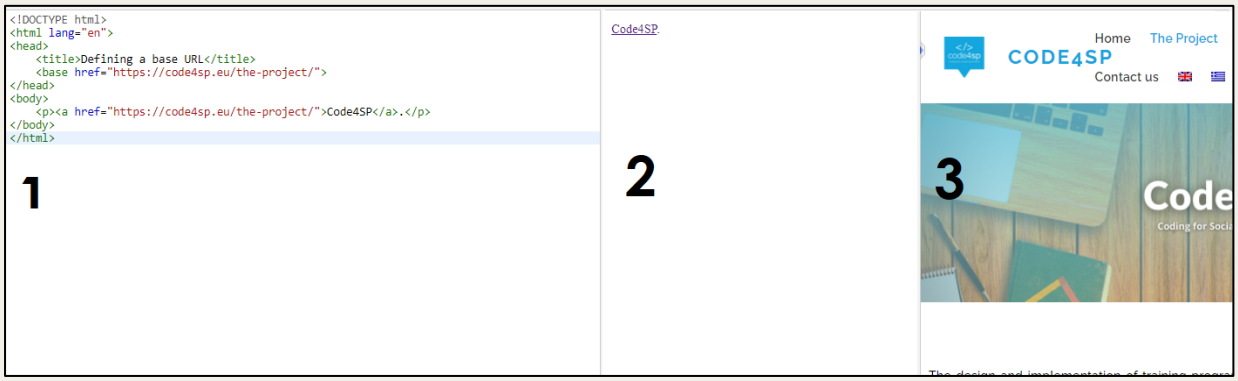
```
<!DOCTYPE html> <html lang="en"> <head> <title>A simple HTML document</title> </head> <body> <p>Hello World! </p> </body> </html>
```

Το στοιχείο base στην HTML (βασική διεύθυνση URL)

Το στοιχείο < base > στην HTML χρησιμοποιείται για τον προσδιορισμό μιας βασικής διεύθυνσης URL για όλους τους σχετικούς συνδέσμους που περιέχονται στο έγγραφο. Για παράδειγμα, μας δίνει τη δυνατότητα να ορίσουμε τη βασική διεύθυνση URL μία φορά στο επάνω μέρος της σελίδας και, στη συνέχεια, όλοι οι επόμενοι σχετικοί σύνδεσμοι θα χρησιμοποιήσουν αυτήν τη διεύθυνση URL ως σημείο εκκίνησης, ως εξής:

```
<!DOCTYPE html> <html lang="en"> <head> <title>Defining a base URL</title> <base href=" https://code4sp.eu/the-project/ "> </head> <body> <p><a href=" https://code4sp.eu/the-project/ ">HTML Head</a>. </p> </body> </html>
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

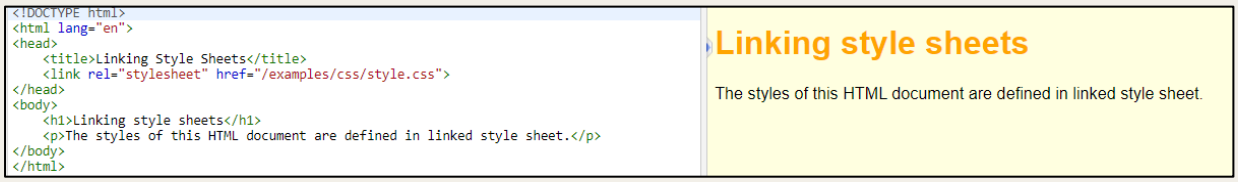


***Εικόνα 63** – Στοιχείο HTML base (Πηγή: Συντάκτης)*

Το στοιχείο < base > στην HTML πρέπει να βρίσκεται πριν από οποιοδήποτε στοιχείο που ανήκει σε έναν εξωτερικό πόρο. Η HTML επιτρέπει μόνο ένα βασικό στοιχείο για κάθε έγγραφο.

Το στοιχείο link στην HTML

Το στοιχείο <link> περιγράφει τη συσχέτιση μεταξύ του τρέχοντος εγγράφου και ενός εξωτερικού εγγράφου ή πόρου. Μια κοινή χρήση του στοιχείου <link> είναι η σύνδεση με εξωτερικά φύλλα στυλ.



***Εικόνα 63** – Στοιχείο base HTML (Πηγή: https://www.tutorialrepublic.com/codelab.php?topic=html&file= linking-style-sheet)*

Αξίζει να σημειωθεί ότι ένα έγγραφο στην HTML με το στοιχείο < head > μπορεί να περιλαμβάνει οποιοδήποτε αριθμό στοιχείων < link>. Το στοιχείο <link> έχει χαρακτηριστικά, αλλά όχι περιεχόμενα.

Το στοιχείο style στην HTML

Το στοιχείο <style> εφαρμόζεται για να περιγράψει ενσωματωμένες πληροφορίες σε ένα έγγραφο HTML. Οι κανόνες στυλ μέσα στο στοιχείο <style> υποδεικνύουν πως πρέπει

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

να αποδίδονται τα στοιχεία HTML σε ένα πρόγραμμα περιήγησης. Ένα ενσωματωμένο φύλλο στυλ θα πρέπει να χρησιμοποιείται όταν ένα μεμονωμένο έγγραφο έχει μοναδικό στυλ. Σε περίπτωση που το ίδιο φύλλο στυλ χρησιμοποιείται σε διάφορα έγγραφα, τότε ένα εξωτερικό φύλλο στυλ θα ήταν πιο κατάλληλο.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Code4SP</title>
    <style>
        body { background-color: Blue; }
        h1 { color: white; }
        p { color: white; }
    </style>
</head>
<body>
    <h1>CODE4SP</h1>
    <p>Coding for social promotion.</p>
</body>
</html>
```

CODE4SP

Coding for social promotion.

Εικόνα 64 *– Κωδικοποίηση διαφόρων στοιχείων style (Πηγή: Συντάκτης)*

Το στοιχείο meta στην HTML

Το στοιχείο < meta > παρέχει μεταδεδομένα σχετικά με το έγγραφο HTML, το οποίο είναι ένα σύνολο δεδομένων που περιγράφει και παρέχει πληροφορίες για άλλα δεδομένα. Οι ετικέτες < meta > εμφανίζονται πάντα μέσα στο στοιχείο < head > και χρησιμοποιούνται συνήθως για τον καθορισμό του συνόλου χαρακτήρων, της περιγραφής σελίδας, των λέξεων-κλειδιών, του δημιουργού του εγγράφου και των ρυθμίσεων του παραθύρου προβολής.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Code4SP</title>
    <meta charset="utf-8">
    <meta name="author" content="CODE4SP project team">
</head>
<body>
    <h1>Code4SP</h1>
    <p>Coding for social promotion.</p>
</body>
</html>
```

Code4SP

Coding for social promotion.

Εικόνα 64 *– Το στοιχείο meta (Πηγή: Συντάκτης)*

Όπως φαίνεται πιο πάνω, οι ετικέτες <meta> περιλαμβάνουν πληροφορίες για μια ιστοσελίδα. Δεν είναι ορατό στο πρόγραμμα περιήγησης (μπορεί όμως να αναλυθεί από μηχανή). Τα μεταδεδομένα χρησιμοποιούνται από προγράμματα περιήγησης (για τον

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

τρόπο εμφάνισης περιεχομένου ή επαναφόρτωσης της σελίδας), μηχανές αναζήτησης (λέξεις-κλειδιά) και άλλες υπηρεσίες Ιστού. Επιπλέον, υπάρχει μια τεχνική που επιτρέπει στους σχεδιαστές ιστοσελίδων να κυριαρχούν στη **θύρα προβολής (viewport)**, μέσω της ετικέτας < meta >. Η θύρα προβολής είναι περιοχή μιας ιστοσελίδας που είναι ορατή από τον χρήστη. Διαφέρει από συσκευή σε συσκευή (θα είναι μεγαλύτερη σε μια οθόνη υπολογιστή από μια οθόνη κινητού τηλεφώνου).

Το ακόλουθο στοιχείο < meta > θα πρέπει να περιλαμβάνεται σε όλες τις ιστοσελίδες, καθώς θα δώσει στο πρόγραμμα περιήγησης οδηγίες για το πώς να υποθέσει τις διαστάσεις και την κλιμάκωση της σελίδας:
<meta name="viewport" content="width=device-width, initial-scale=1.0">
Το σημείο width=device-width ορίζει το πλάτος της σελίδας ώστε να ακολουθεί το πλάτος οθόνης της συσκευής (το οποίο θα διαφέρει ανάλογα με την οθόνη). Το σημείο initial-scale=1.0 ορίζει το αρχικό επίπεδο ζουμ όταν η σελίδα φορτώνεται αρχικά από το πρόγραμμα περιήγησης.
Η διαφορά μεταξύ ιστοσελίδων που έχουν την ετικέτα <meta> στο παράθυρο προβολής με αυτές που δεν έχουν, φαίνεται ξεκάθαρα στο ακόλουθο παράδειγμα:

Εικόνα 65 *– Παραδείγματα με και χωρίς ετικέτες meta στο παράθυρο προβολής (Πηγή: https://www.w3schools.com/tags/tag_meta.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Το στοιχείο script στην HTML

Το στοιχείο < script > χρησιμοποιείται για να καθορίζει το σενάριο από την πλευρά του προγράμματος-πελάτη, όπως το JavaScript σε έγγραφα HTML.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Adding JavaScript</title>
    <script>
        document.write("<h1>Hello CODE4SP learners!</h1>")
    </script>
</head>
<body>
    <p>The above heading is inserted in this document by JavaScript.</p>
</body>
</html>
```

Hello CODE4SP learners!

The above heading is inserted in this document by JavaScript.

Εικόνα 65 *– Το στοιχείο script (Πηγή: Συντάκτης)*

Δουλεύοντας με σενάριο από την πλευρά του προγράμματος-πελάτη

Η δέσμη ενεργειών από την πλευρά του προγράμματος-πελάτη συνδέεται με τον τύπο των προγραμμάτων υπολογιστή που εκτελούνται από το πρόγραμμα περιήγησης του χρήστη. **Η JavaScript (JS)** είναι η πιο διαδεδομένη γλώσσα προγραμματισμού σεναρίων από την πλευρά του προγράμματος πελάτη στον ιστό. Το στοιχείο <script> χρησιμοποιείται για να ενσωματώσει την JavaScript σε ένα έγγραφο HTML έτσι ώστε να προσθέσει τη δυνατότητα διαδραστικότητας σε ιστοσελίδες και για να δημιουργήσει μια εμπειρία πιο φιλική προς τον χρήστη. Μερικές από τις πιο κοινές χρήσεις της JavaScript είναι η επικύρωση φόρμας, η δημιουργία μηνυμάτων ειδοποίησης, η δημιουργία συλλογής εικόνων, η εμφάνιση περιεχομένου απόκρυψης, ο χειρισμός Μοντέλου Αντικειμένου Εγγράφου (DOM) κ.λπ.

Η JavaScript μπορεί να **ενσωματωθεί** με δύο τρόπους:

1. Απευθείας μέσα στη σελίδα HTML ή
2. Να τοποθετηθεί σε ένα εξωτερικό αρχείο σεναρίου και να προστεθεί στη σελίδα HTML.

Σημείωση: και οι δύο τεχνικές χρησιμοποιούν το στοιχείο <script>.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για να ενσωματώσει ο χρήστης την JS σε ένα αρχείο HTML, θα πρέπει να προσθέσει τον κώδικα ως περιεχόμενο του στοιχείου < script> ακολουθώντας το παράδειγμα που φαίνεται στην *Εικόνα 66*. Κατά προτίμηση, τα στοιχεία του σεναρίου θα πρέπει να τοποθετούνται στο τέλος της σελίδας, πριν από την ετικέτα τέλους (</body >), επειδή όταν το πρόγραμμα περιήγησης συναντά ένα σενάριο, διακόπτει την απόδοση της υπόλοιπης σελίδας μέχρι να αποδομήσει το σενάριο που μπορεί να επηρεάσει δραστικά την απόδοση του ιστότοπου.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Embedding JavaScript</title>
</head>
<body>
    <div id="greet"></div>
    <script>
        document.getElementById("greet").innerHTML = "Code4SP";
    </script>
</body>
</html>
```

Εικόνα 66 *– Ενσωμάτωση της JS σε έγγραφο HTML (Πηγή: Συντάκτης)*

Οι κώδικες JavaScript μπορούν επίσης να τοποθετηθούν σε ένα ξεχωριστό αρχείο, προσθέτοντας μια επέκταση .js, ενώ την αντιστοιχούν στο έγγραφο HTML χρησιμοποιώντας το χαρακτηριστικό src της ετικέτας <script>. Αυτό μπορεί να είναι ιδιαίτερα χρήσιμο εάν ο σχεδιαστής ιστοσελίδων θέλει να κάνει το ίδιο σενάριο διαθέσιμο για πολλά έγγραφα, έτσι ώστε να μην χρειάζεται να εκτελεί τις ίδιες ενέργειες κάθε φορά. Όταν υποδεικνύεται το χαρακτηριστικό src, το στοιχείο < script > θα πρέπει να είναι άδειο, έτσι ώστε ο σχεδιαστής της ιστοσελίδας να μην μπορεί να χρησιμοποιήσει το ίδιο στοιχείο < script > και για να ενσωματώσει την JavaScript και για να συνδέσει ένα αρχείο JavaScript σε ένα έγγραφο HTML.

Εάν ένα πρόγραμμα περιήγησης δεν υποστηρίζει σενάρια από την πλευρά του προγράμματος- πελάτη ή οι χρήστες έχουν απενεργοποιήσει την JS στο πρόγραμμα

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

περιήγησής τους, μπορεί να χρησιμοποιηθεί το στοιχείο <no script> για την παροχή ενός εναλλακτικού περιεχομένου. Αυτό το στοιχείο μπορεί να περιλαμβάνει οποιαδήποτε στοιχεία HTML, εκτός από το <script>, καθώς αυτό μπορεί να συμπεριληφθεί στο στοιχείο <body> μιας σελίδας HTML.

Οντότητες στην HTML

Οι περισσότεροι μαθητές σίγουρα θα είναι περίεργοι για το πώς να εμφανίζουν ειδικούς χαρακτήρες και σύμβολα στις διαδικασίες προγραμματισμού τους. Ως εκ τούτου, αυτό το υποκεφάλαιο έχει σκοπό να εξηγήσει πώς μπορούν να επιτύχουν κάτι τέτοιο.

Πρώτον, είναι σημαντικό να κατανοήσουμε τι είναι μια οντότητα HTML. Όπως έγινε αντιληπτό στα προηγούμενα κεφάλαια, ορισμένοι χαρακτήρες είναι αρκετά περιορισμένοι στην HTML. Για παράδειγμα, δεν μπορούν να χρησιμοποιηθούν σύμβολα όπως «<» ή «>», καθώς το πρόγραμμα περιήγησης θα μπορούσε να τα μπερδέψει ως σήμανση. Επιπλέον, ορισμένοι χαρακτήρες δεν είναι διαθέσιμοι στο πληκτρολόγιο (π.χ. το σύμβολο πνευματικών δικαιωμάτων).

Αυτοί οι ειδικοί χαρακτήρες μπορούν να εμφανιστούν αντικαθιστώντας τους απλά με τις οντότητες χαρακτήρων (ή απλώς οντότητες) και στη συνέχεια λύνοντας τα προαναφερθέντα προβλήματα. Παρακάτω ακολουθεί μια λίστα με τις πιο συχνά χρησιμοποιούμενες οντότητες στην HTML:

Αποτέλεσμα	Περιγραφή	Όνομα Οντότητας	Αριθμητική αναφορά
	μη χωριζόμενο κενό		
<	Λιγότερο από	<	<
>	μεγαλύτερο από	>	>

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

&	συμπλεκτικό σύμβολο	&amp;	&#38;
"	Εισαγωγικά	&quot;	&#34;
'	απόστροφος	&apos;	&#39;
¢	σεντ	&cent;	&#162;
£	λίρα	&pound;	&#163;
¥	ιαπωνικό γιεν	&yen;	&#165;
€	ευρώ	&euro;	&#8364;
©	πνευματικά δικαιώματα	&copy;	&#169;
®	σήμα κατατεθέν	&reg;	&#174;
™	εμπορικό σήμα	&trade;	&#8482;

Πίνακας 3 *– Οι πιο συχνές οντότητες στην HTML (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-entities.php)*

Οι αναφορές με αριθμητικούς χαρακτήρες μπορούν επίσης να χρησιμοποιηθούν ως εναλλακτική στα ονόματα οντοτήτων, ειδικά επειδή έχουν ισχυρότερη υποστήριξη στα προγράμματα περιήγησης και μπορούν να χρησιμοποιηθούν για τον καθορισμό οποιουδήποτε χαρακτήρα Unicode, ωστόσο οι οντότητες περιορίζονται σε μια υποομάδα αυτού του χαρακτήρα.

Διεύθυνση URL στην HTML

URL: πόσες φορές έχει εμφανιστεί αυτή η συντομογραφία σε εικονικά περιβάλλοντα; Τα αρχικά της αντιπροσωπεύουν το Uniform ReΠηγή Locator (διεύθυνση ιστοσελίδας) και λειτουργεί ως παγκόσμια διεύθυνση εγγράφων και άλλων πόρων στον Παγκόσμιο Ιστό.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στόχος του είναι να προσδιορίσει τη θέση ενός εγγράφου και άλλων πόρων που είναι διαθέσιμοι στο διαδίκτυο, καθορίζοντας παράλληλα τον μηχανισμό πρόσβασης σε αυτό χρησιμοποιώντας ένα πρόγραμμα περιήγησης Ιστού. Για παράδειγμα, το https://code4sp.eu/ είναι μια διεύθυνση URL.

Η γενική σύνταξη των διευθύνσεων URL μπορεί να περιγραφεί ως εξής: `scheme://host:port/path?query-string#fragment-id`

Οι διευθύνσεις URL έχουν γραμμική δομή και αποτελούνται από τα ακόλουθα στοιχεία:

- **Όνομα σχήματος *(Scheme Name)***— Το σχήμα αναγνωρίζει το πρωτόκολλο που θα χρησιμοποιηθεί για την πρόσβαση σε έναν πόρο στο Διαδίκτυο. Τα ονόματα των σχημάτων ακολουθούνται από τους τρεις χαρακτήρες://(άνω και κάτω τελεία και δύο κάθετοι). Τα πρωτόκολλα που χρησιμοποιούνται περισσότερο είναι τα http://, https://, ftp://και mailto://.
- **Όνομα κεντρικού υπολογιστή *(Host Name)*** — προσδιορίζει τον κεντρικό υπολογιστή όπου βρίσκεται ο πόρος. Ένα όνομα κεντρικού υπολογιστή είναι ένα όνομα τομέα που δίνεται σε έναν κεντρικό υπολογιστή. Αυτός είναι συνήθως ένας συνδυασμός του τοπικού ονόματος του κεντρικού υπολογιστή με το όνομα του τομέα ανώτατου επιπέδου του. Για παράδειγμα, το www.code4sp.eu/ αποτελείται από το όνομα μηχανής του κεντρικού υπολογιστή www και το όνομα τομέα code4sp.eu.
- **Αριθμός θύρας (Port Number)**— Οι διακομιστές συχνά παρέχουν περισσότερους από έναν τύπους υπηρεσιών, επομένως πρέπει να ενημερώνονται για το ποια υπηρεσία ζητείται. Αυτά τα αιτήματα γίνονται με αριθμό θύρας. Οι γνωστοί αριθμοί θυρών για μια υπηρεσία συνήθως παραλείπονται από τη διεύθυνση URL. Για παράδειγμα, η υπηρεσία διαδικτύου HTTP εκτελείται από προεπιλογή στη θύρα 80, ενώ το HTTPS εκτελείται από προεπιλογή στη θύρα 443.
- **Μονοπάτι (Path)** — προσδιορίζει τον συγκεκριμένο πόρο εντός του κεντρικού υπολογιστή στον οποίο ο χρήστης θέλει να έχει πρόσβαση. Για παράδειγμα, /html/html-url.php,/news/technology/, κ.λπ.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- **Ερώτημα (Query String)** — περιέχει δεδομένα που πρέπει να μεταβιβαστούν σε σενάρια από την πλευρά του διακομιστή, που εκτελούνται στον διακομιστή Ιστού. Για παράδειγμα, παράμετροι για μια αναζήτηση. Το ερώτημα, το οποίο προηγείται από ένα ερωτηματικό του λατινικού αλφαβήτου (?), είναι συνήθως μια συμβολοσειρά ζευγών ονομάτων και τιμών που χωρίζονται με συμπλεκτικό σύμβολο (&), για παράδειγμα, ?first_name=John&last_name=Corner, q=mobile+phone, και ούτω καθεξής.
- **Αναγνωριστικό Τμήματος (Fragment Identifier)** — καθορίζει μια τοποθεσία μέσα στη σελίδα. Το πρόγραμμα περιήγησης μπορεί να μετακινήσει τη σελίδα για να εμφανίσει αυτό το τμήμα της σελίδας. Το αναγνωριστικό τμήματος που εισάγεται με το σύμβολο (#) είναι το προαιρετικό τελευταίο μέρος μιας διεύθυνσης URL ενός εγγράφου.

Κωδικοποίηση της διεύθυνσης URL στην HTML

Γνωρίζουμε ότι μερικές φορές τα δεδομένα δεν μεταδίδονται με ασφάλεια μέσω του Διαδικτύου. Αυτό συμβαίνει κυρίως επειδή οι διευθύνσεις URL δεν είναι πλήρως ή κωδικοποιημένες με ακρίβεια, κάτι που μπορεί να προκαλέσει ορισμένες παρεξηγήσεις μεταξύ των χρηστών του Διαδικτύου.

Σύμφωνα με το RFC 3986, οι χαρακτήρες σε μια διεύθυνση URL περιορίζονται μόνο σε ένα καθορισμένο σύνολο δεσμευμένων και μη δεσμευμένων χαρακτήρων US-ASCII. Δεν επιτρέπονται άλλοι χαρακτήρες σε μια διεύθυνση URL (γι' αυτό ορισμένοι λατινικοί και κυριλλικοί χαρακτήρες δεν εμφανίζονται στις διευθύνσεις URL). Ωστόσο, μια διεύθυνση URL συχνά περιλαμβάνει χαρακτήρες εκτός του συνόλου χαρακτήρων US-ASCII, επομένως πρέπει να μετατραπούν σε έγκυρη μορφή US-ASCII για παγκόσμια διαλειτουργικότητα. Η κωδικοποίηση μιας διεύθυνσης URL είναι μια διαδικασία μετατροπής πληροφοριών URL έτσι ώστε να μπορούν να μεταδοθούν με ασφάλεια μέσω του Διαδικτύου.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για να καταγραφεί το μεγάλο εύρος των χαρακτήρων που χρησιμοποιούνται παγκοσμίως, ακολουθείται μια μέθοδος που αποτελείται από δύο βήματα:

- Πρώτον, τα δεδομένα κωδικοποιούνται σύμφωνα με την κωδικοποίηση χαρακτήρων UTF-8.
- Στη συνέχεια, μόνο εκείνα τα byte που δεν αντιστοιχούν σε χαρακτήρες στο μη δεσμευμένο σύνολο θα πρέπει να κωδικοποιούνται επί τοις εκατό όπως % HH, όπου HH είναι η δεκαεξαδική τιμή του byte.

Για παράδειγμα, δείτε αυτό το δημοφιλές πορτογαλικό ρητό:

“Quem vê caras, não vê corações.” ["Πρόσωπα που βλέπουμε, καρδιές που δεν ξέρουμε."]

Αυτή η πρόταση θα κωδικοποιηθεί ως εξής:

Quem%20v%C3%AA%20caras%2C%20n%C3%A3o%20v%C3%AA%20cora%C3%A7%C3%B5es.

Το Ç, ç (c-cedilla) είναι γράμμα του λατινικού αλφαβήτου, καθώς και οι τόνοι που χρησιμοποιούνται (~ και ^).

Ορισμένοι χαρακτήρες περιορίζονται στη χρήση σε μια διεύθυνση URL, καθώς μπορεί (ή όχι) να ορίζονται ως οριοθέτες από τη γενική σύνταξη σε ένα συγκεκριμένο σχήμα διεύθυνσης URL (για παράδειγμα, κάθετο προς τα εμπρός / χρήση χαρακτήρων για τον διαχωρισμό διαφορετικών τμημάτων μιας διεύθυνσης URL).

Οι δεσμευμένοι χαρακτήρες σε μια διεύθυνση URL είναι οι εξής:

!	#	$	&	'	(	)	*	+	,	/	:	;	=	?	@	[	]
%21	%23	%24	%26	%27	%28	%29	%2A	%2B	%2C	%2F	%3A	%3B	%3D	%3F	%40	%5B	%5D

***Εικόνα 67** – Δεσμευμένοι χαρακτήρες σε μια διεύθυνση URL (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-url-encode.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ωστόσο, υπάρχουν και χαρακτήρες που, παρόλο που επιτρέπονται σε μια διεύθυνση URL, δεν έχουν δεσμευμένο σκοπό, γι' αυτό ονομάζονται «μη δεσμευμένοι χαρακτήρες».

Αυτοί οι χαρακτήρες αποτελούνται από κεφαλαία και πεζά γράμματα, δεκαδικά ψηφία, παύλα, τελεία, κάτω παύλα και περισπωμένη. Ο παρακάτω πίνακας παραθέτει όλους τους μη δεσμευμένους χαρακτήρες σε μια διεύθυνση URL:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
0	1	2	3	4	5	6	7	8	9	-	_	.	~												

Εικόνα 68 *– Μη Δεσμευμένοι χαρακτήρες σε μια διεύθυνση URL (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html-url-encode.php)*

Για την κωδικοποίηση/αποκωδικοποίηση χαρακτήρων, οι χρήστες μπορούν να χρησιμοποιήσουν αυτόν τον μετατροπέα.

Η Επικύρωση στην HTML

Για να βεβαιωθούμε ότι ένας κώδικας HTML ακολουθεί τα τρέχοντα πρότυπα ιστού, χωρίς σφάλματα, είναι εξαιρετικά σημαντικό να καταλάβουμε πώς να τον επικυρώσουμε. Οι αρχάριοι συχνά κάνουν λάθη όταν γράφουν κώδικες HTML και οι λανθασμένοι ή μη τυπικοί κώδικες θα προκαλέσουν σίγουρα απροσδόκητα αποτελέσματα στον τρόπο εμφάνισης μιας ιστοσελίδας σε ένα πρόγραμμα περιήγησης.

Προκειμένου να αποφευχθεί αυτό, οι χρήστες μπορούν να δοκιμάσουν τους κώδικές ακολουθώντας τις επίσημες οδηγίες και τα πρότυπα που ορίζονται από την Επιτροπή του παγκόσμιου ιστού (W3C) για ιστοσελίδες σε HTML/XHTML. Υπάρχει ένα διαδικτυακό εργαλείο που ελέγχει αυτόματα τους κώδικες HTML και βρίσκει τυχόν προβλήματα/λάθη, όπως για παράδειγμα αν λείπουν ετικέτες τέλους ή τα εισαγωγικά πριν και μετά τα χαρακτηριστικά.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η διαδικασία επικύρωσης μιας ιστοσελίδας διασφαλίζει τον σεβασμό των κανόνων/προτύπων που ορίζονται από την W3C, επομένως είναι μια πολύ σημαντική διαδικασία. Μερικοί λόγοι για την επικύρωση μιας ιστοσελίδας είναι:

- Βοηθά στη δημιουργία ιστοσελίδων που είναι συμβατές με προγράμματα περιήγησης και προγράμματα που είναι ανεξάρτητα πλατφόρμας. Επίσης έχουν πολλές πιθανότητες να είναι συμβατά με τη μελλοντική έκδοση των προγραμμάτων περιήγησης και των προτύπων Ιστού.
- Οι ιστοσελίδες που συμμορφώνονται με τα πρότυπα αυξάνουν την ορατότητα των προγραμμάτων ανίχνευσης ιστού των μηχανών αναζήτησης, με αποτέλεσμα οι ιστοσελίδες σας να έχουν περισσότερες πιθανότητες να εμφανίζονται στα αποτελέσματα αναζήτησης.
- Μειώνει τα απροσδόκητα σφάλματα και κάνει τις ιστοσελίδες σας πιο προσβάσιμες στον επισκέπτη.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

2.3 Λειτουργίες στην HTML5

Σε αυτή την υποενότητα, θα επεξηγηθούν οι λειτουργίες στην πέμπτη (και τελευταία) κύρια έκδοση της HTML που προτείνει η W3C.

Νέοι τύποι εισαγωγής της HTML5

Η HTML5 εισάγει πολλούς νέους τύπους <input>, όπως ηλεκτρονικό ταχυδρομείο, ημερομηνία, ώρα, χρώμα, εύρος και ούτω καθεξής, με στόχο τη βελτίωση της εμπειρίας του χρήστη και τη βελτίωση της διαδραστικότητας των φορμών. Ωστόσο, εάν ένα πρόγραμμα περιήγησης αδυνατεί να αναγνωρίσει αυτούς τους νέους τύπους εισαγωγής, θα τους θεωρήσει κανονικό πλαίσιο κειμένου.

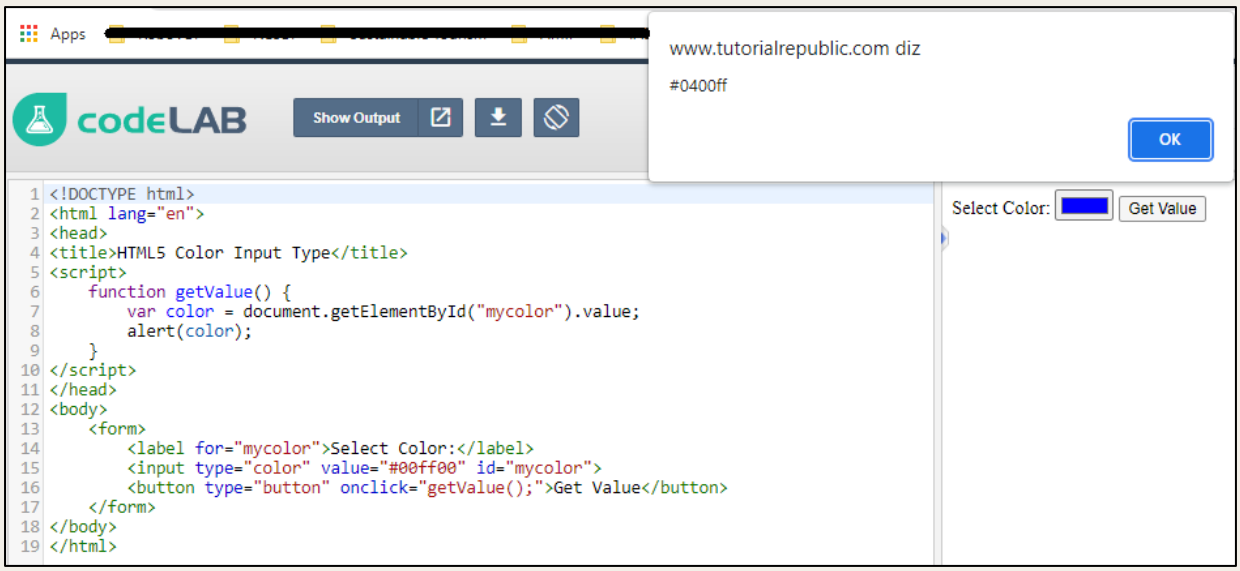
Ακολουθούν ορισμένοι νέοι τύποι εισαγωγής:

- χρώμα
- ημερομηνία
- ημερομηνία-ώρα-τοπική
- ηλεκτρονικό ταχυδρομείο
- μήνας
- αριθμός
- εύρος
- αναζήτηση
- τηλέφωνο
- ώρα
- διεύθυνση url
- εβδομάδα

Όσον αφορά την εισαγωγή χρώματος (color input), επιτρέπει την επιλογή ενός χρώματος από έναν επιλογέα χρωμάτων και δίνει πληροφορίες σχετικά με την τιμή χρώματος σε δεκαεξαδική μορφή (π.χ. #000000, που είναι μαύρο και είναι το προεπιλεγμένο χρώμα εάν ο χρήστης δεν καθορίσει μια τιμή), όπως φαίνεται στην *Εικόνα 69* παρακάτω.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

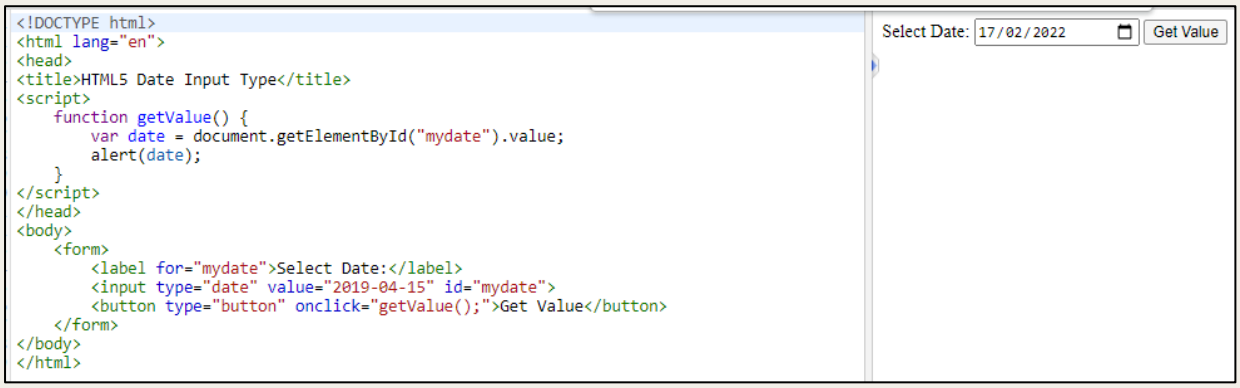
Θα πρέπει να σημειωθεί ότι η εισαγωγή χρώματος υποστηρίζεται σε όλα τα μεγάλα σύγχρονα προγράμματα περιήγησης (Firefox, Chrome, Opera, Safari (12.1+), Edge (14+)), αλλά δεν υποστηρίζεται από το Microsoft Internet Explorer και παλαιότερες εκδόσεις των προγραμμάτων περιήγησης Apple Safari.



Εικόνα 69 *– Επιλογή χρώματος χρησιμοποιώντας το color input (Πηγή: https://www.tutorialrepublic.com/codelab.php?topic=html5&file= color-input-type)*

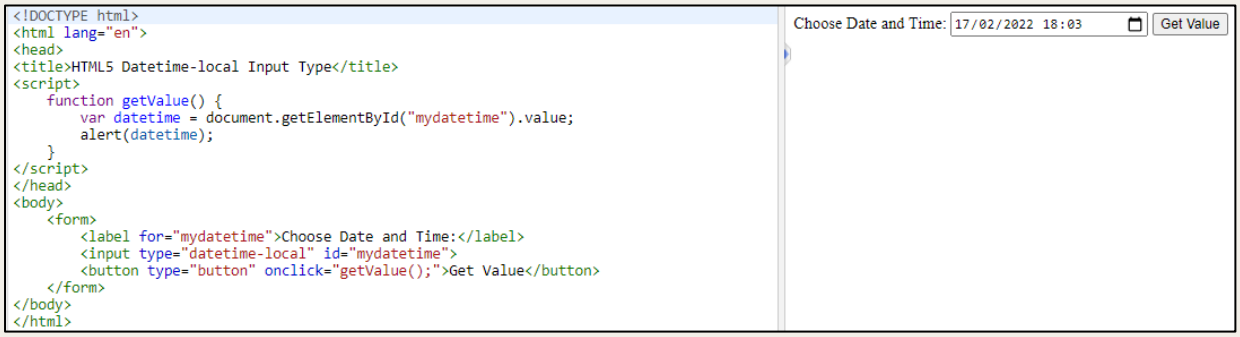
Ο τύπος εισαγωγής ημερομηνίας (date input) επιτρέπει στο χρήστη να επιλέξει μια ημερομηνία από ένα αναπτυσσόμενο ημερολόγιο, στο οποίο μπορεί να επιλέξει το έτος, το μήνα και την ημέρα (αλλά όχι την ώρα). Αυτή η δυνατότητα υποστηρίζεται επίσης από τα περισσότερα προγράμματα περιήγησης, εκτός από το Internet Explorer και το Safari.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 70 *– Ο τύπος εισαγωγής ημερομηνίας (Πηγή: https://www.tutorialrepublic.com/codelab.php?topic=html5&file= color-input-type)*

Η εισαγωγή τοπικής ώρας και ημερομηνίας (date-time local) δίνει τη δυνατότητα στο χρήστη να επιλέξει την τοπική ημερομηνία και ώρα, συμπεριλαμβανομένου του έτους, του μήνα και της ημέρας, καθώς και την ώρα σε ώρες και λεπτά. Αυτή η εισαγωγή υποστηρίζεται από το Safari, το Firefox και το IE, αλλά όχι από το Chrome, το Edge και το Opera.

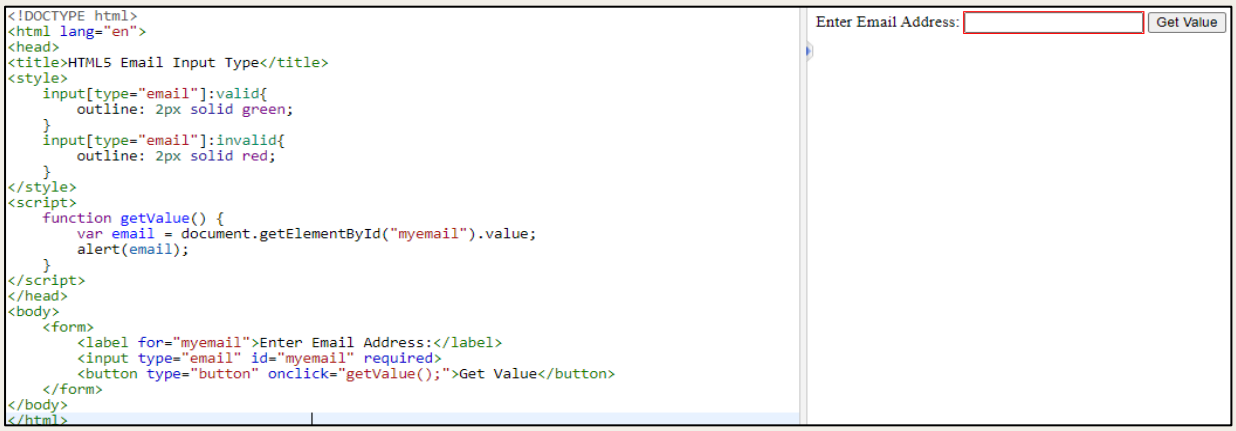


Εικόνα 71 *– Ο τύπος εισαγωγής τοπικής ημερομηνίας και ώρας (Πηγή: https://www.tutorialrepublic.com/codelab.php?topic=html5&file= color-input-type)*

Ο καλύτερος τύπος εισαγωγής για να εισάγουν οι χρήστες τι διεύθυνση ηλεκτρονικού ταχυδρομείου τους, είναι ο τύπος εισαγωγής email. Είναι σαν ένας συνηθισμένος τύπος εισαγωγής κειμένου, αλλά εάν εφαρμοστεί σε συνδυασμό με το χαρακτηριστικό required, τα προγράμματα περιήγησης μπορούν να αναζητήσουν τα μοτίβα για να διασφαλίσουν ότι θα εισαχθεί μια σωστά διαμορφωμένη διεύθυνση ηλεκτρονικού ταχυδρομείου. Το

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

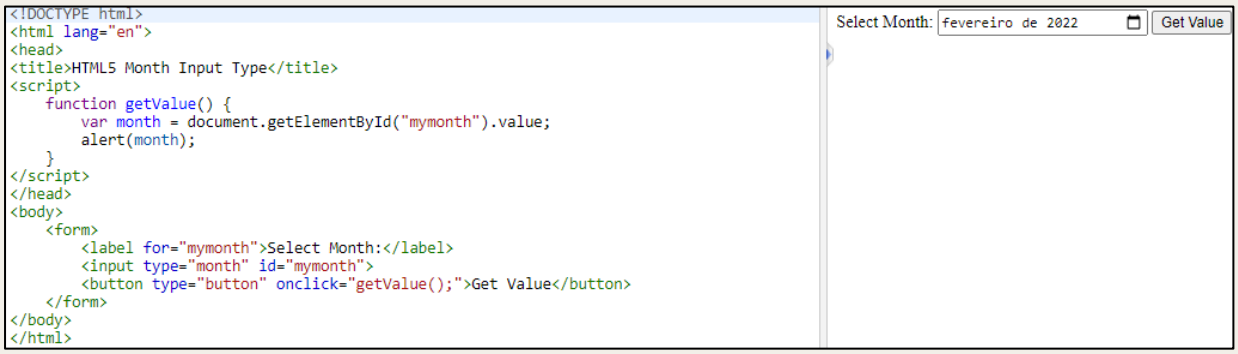
πεδίο εισαγωγής email μπορεί να διαμορφωθεί για διαφορετικές καταστάσεις επικύρωσης, όταν μια τιμή εισάγεται χρησιμοποιώντας τις ψευδοκλάσεις (pseudo-classes) :valid, :invalid ή :required. Αυτός ο τύπος εισαγωγής υποστηρίζεται από όλα τα μεγάλα προγράμματα περιήγησης.




```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Email Input Type</title>
<style>
    input[type="email"]:valid{
        outline: 2px solid green;
    }
    input[type="email"]:invalid{
        outline: 2px solid red;
    }
</style>
<script>
    function getValue() {
        var email = document.getElementById("myemail").value;
        alert(email);
    }
</script>
</head>
<body>
    <form>
        <label for="myemail">Enter Email Address:</label>
        <input type="email" id="myemail" required>
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Εικόνα 72 *– Ο τύπος εισαγωγής email (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Η εισαγωγή του μήνα (month input) είναι ένα πολύ παρόμοιο χαρακτηριστικό με τα προηγούμενα, καθώς επιτρέπει την επιλογή ενός μήνα και ενός έτους από ένα αναπτυσσόμενο ημερολόγιο (με 'EEEE' για το έτος και 'MM' για τον μήνα. Θα πρέπει να σημειωθεί ότι αυτό δεν υποστηρίζεται από τα προγράμματα περιήγησης Firefox, Safari και Internet Explorer. Υποστηρίζεται μόνο από τα προγράμματα περιήγησης Chrome, Edge και Opera.

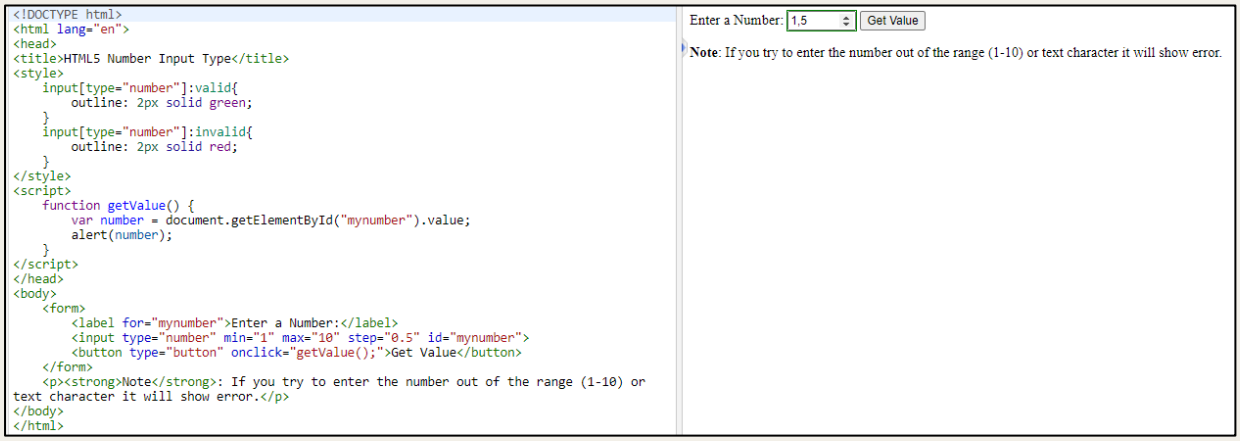



```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Month Input Type</title>
<script>
    function getValue() {
        var month = document.getElementById("mymonth").value;
        alert(month);
    }
</script>
</head>
<body>
    <form>
        <label for="mymonth">Select Month:</label>
        <input type="month" id="mymonth">
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Εικόνα 73 *– Ο τύπος εισαγωγής μήνα (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

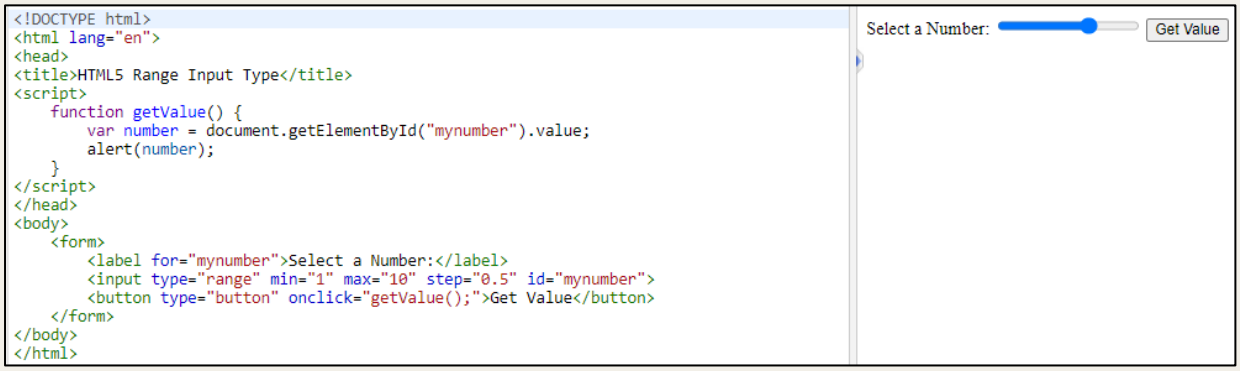
Όσον αφορά τον τύπο εισαγωγής αριθμού (number input), χρησιμοποιείται για την εισαγωγή μιας αριθμητικής τιμής. Ο σχεδιαστής ιστοσελίδων μπορεί επίσης να περιορίσει τον χρήστη έτσι ώστε να μπορεί να εισάγει μόνο αποδεκτές τιμές. Για να συμβεί αυτό, θα πρέπει να χρησιμοποιηθούν τα πρόσθετα χαρακτηριστικά min, max και step. Αυτή η δυνατότητα υποστηρίζεται από όλα τα μεγάλα προγράμματα περιήγησης ιστού.




```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Number Input Type</title>
<style>
    input[type="number"]:valid{
        outline: 2px solid green;
    }
    input[type="number"]:invalid{
        outline: 2px solid red;
    }
</style>
<script>
    function getValue() {
        var number = document.getElementById("mynumber").value;
        alert(number);
    }
</script>
</head>
<body>
    <form>
        <label for="mynumber">Enter a Number:</label>
        <input type="number" min="1" max="10" step="0.5" id="mynumber">
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
    <p><strong>Note</strong>: If you try to enter the number out of the range (1-10) or
text character it will show error.</p>
</body>
</html>
```

Εικόνα 74 *– Ο τύπος εισαγωγής αριθμού (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Ο τύπος εισαγωγής εύρους (range input) μπορεί να εφαρμοστεί για την καταχώριση μιας αριθμητικής τιμής εντός ενός συγκεκριμένου εύρους. Λειτουργεί πολύ παρόμοια με την εισαγωγή αριθμών που είδαμε παραπάνω, αν και παρουσιάζει έναν ευκολότερο τρόπο για την εισαγωγή ενός αριθμού. Αυτός ο τύπος εισόδου υποστηρίζεται από όλα τα μεγάλα προγράμματα περιήγησης.




```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Range Input Type</title>
<script>
    function getValue() {
        var number = document.getElementById("mynumber").value;
        alert(number);
    }
</script>
</head>
<body>
    <form>
        <label for="mynumber">Select a Number:</label>
        <input type="range" min="1" max="10" step="0.5" id="mynumber">
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Εικόνα 75 *– Ο τύπος εισαγωγής εύρους (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

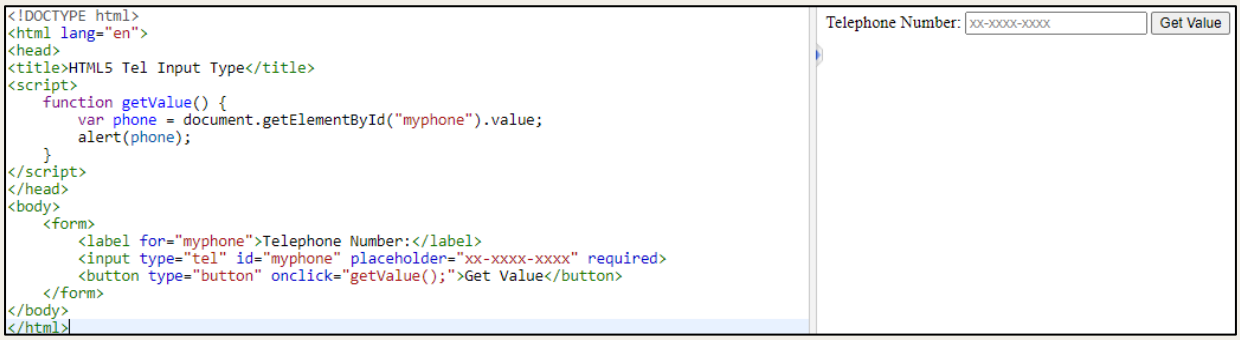
Ο τύπος εισαγωγής αναζήτησης (search input) είναι κατάλληλος για τη δημιουργία πεδίων εισαγωγής αναζήτησης. Αξίζει να αναφερθεί ότι, σε ορισμένα προγράμματα περιήγησης (δηλαδή, Chrome και Safari), μόλις ο χρήστης αρχίσει να πληκτρολογεί στο πλαίσιο αναζήτησης, εμφανίζεται ένας μικροσκοπικός σταυρός στη δεξιά πλευρά του πεδίου, ο οποίος μπορεί να χρησιμοποιηθεί για να καθαριστεί ολόκληρο το πεδίο αναζήτησης. Υποστηρίζεται από όλα τα μεγάλα προγράμματα περιήγησης.


```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Search Input Type</title>
<script>
    function getValue() {
        var term = document.getElementById("mysearch").value;
        alert(term);
    }
</script>
</head>
<body>
    <form>
        <label for="mysearch">Search Website:</label>
        <input type="search" id="mysearch" placeholder="Type something...">
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Εικόνα 76 *– Ο τύπος εισαγωγής αναζήτησης (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Ο τύπος εισαγωγής τηλεφώνου (tel input) είναι ιδιαίτερα χρήσιμος για την εισαγωγή ενός αριθμού τηλεφώνου. Καθώς τα προγράμματα περιήγησης δεν υποστηρίζουν την επικύρωση εισαγωγής τηλεφώνου από προεπιλογή, το χαρακτηριστικό placeholder μπορεί να χρησιμοποιηθεί για να βοηθήσει τους χρήστες να εισάγουν τη σωστή μορφή για τον αριθμό τηλεφώνου τους ή να υποδείξουν μια τυπική έκφραση για την επικύρωση της εισαγωγής του χρήστη εφαρμόζοντας το χαρακτηριστικό pattern. Αυτή η δυνατότητα δεν υποστηρίζεται από κανένα πρόγραμμα περιήγησης, καθώς οι αριθμοί τηλεφώνου ποικίλλουν πολύ παγκοσμίως.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)




```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Tel Input Type</title>
<script>
    function getValue() {
        var phone = document.getElementById("myphone").value;
        alert(phone);
    }
</script>
</head>
<body>
    <form>
        <label for="myphone">Telephone Number:</label>
        <input type="tel" id="myphone" placeholder="xx-xxxx-xxxx" required>
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Εικόνα 77 *– Ο τύπος εισαγωγής τηλεφώνου (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Όσον αφορά τον τύπο εισαγωγής ώρας (time input), μπορεί να χρησιμοποιηθεί για την εισαγωγή οποιασδήποτε δεδομένης ώρας (ώρες και λεπτά) και το πρόγραμμα περιήγησης μπορεί να χρησιμοποιήσει και τις δύο μορφές ωρών (12 ή 24 ωρών) για την εισαγωγή ώρας, ανάλογα με την περιοχή. Αυτός ο τύπος εισαγωγής δεν υποστηρίζεται από προγράμματα περιήγησης IE και Safari.


```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Time Input Type</title>
<script>
    function getValue() {
        var time = document.getElementById("mytime").value;
        alert(time);
    }
</script>
</head>
<body>
    <form>
        <label for="mytime">Select Time:</label>
        <input type="time" id="mytime">
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Εικόνα 78 *– Ο τύπος εισαγωγής ώρας (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Όσον αφορά τον τύπο εισαγωγής διεύθυνσης url (url input), μπορεί να χρησιμοποιηθεί για την εισαγωγή διευθύνσεων URL ή ιστοσελίδων. Το χαρακτηριστικό multiple μπορεί να χρησιμοποιηθεί για την εισαγωγή περισσότερων από μία διευθύνσεων URL. Επιπλέον, εάν το χαρακτηριστικό required δεν δύναται να χρησιμοποιηθεί, το πρόγραμμα περιήγησης θα εκτελέσει αυτόματα επικύρωση για να διασφαλίσει ότι μόνο

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

το κείμενο που πληροί την τυπική μορφή για τις διευθύνσεις URL θα μπει στο πλαίσιο εισαγωγής. Όλα τα μεγάλα προγράμματα περιήγησης υποστηρίζουν αυτόν τον τύπο εισαγωγής.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 URL Input Type</title>
<style>
    input[type="url"]:valid{
        outline: 2px solid green;
    }
    input[type="url"]:invalid{
        outline: 2px solid red;
    }
</style>
<script>
    function getValue() {
        var url = document.getElementById("myurl").value;
        alert(url);
    }
</script>
</head>
<body>
    <form>
        <label for="myurl">Enter Website URL:</label>
        <input type="url" id="myurl" required>
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
    <p><strong>Note</strong>: Enter URL in the form like https://www.google.com</p>
</body>
</html>
```

Enter Website URL: Get Value

Note: Enter URL in the form like https://www.google.com

Εικόνα 79 *– Ο τύπος εισαγωγής διεύθυνσης url (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Τέλος, ο τύπος εισαγωγής εβδομάδας (week input) επιτρέπει στο χρήστη να επιλέξει μια εβδομάδα και ένα έτος από ένα αναπτυσσόμενο ημερολόγιο. Αυτή η δυνατότητα δεν υποστηρίζεται από Firefox, Safari και IE, αλλά προς το παρόν υποστηρίζεται από τα προγράμματα περιήγησης Chrome, Edge και Opera.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Week Input Type</title>
<script>
    function getValue() {
        var week = document.getElementById("myweek").value;
        alert(week);
    }
</script>
</head>
<body>
    <form>
        <label for="myweek">Select Week:</label>
        <input type="week" id="myweek">
        <button type="button" onclick="getValue();">Get Value</button>
    </form>
</body>
</html>
```

Select Week: Semana --, ---- Get Value

Εικόνα 80 *– Ο τύπος εισαγωγής εβδομάδας (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο Καμβάς στην HTML5

Αυτή η εποενότητα θα είναι χρήσιμη για να μάθετε πώς να σχεδιάζετε γραφικά χρησιμοποιώντας το στοιχείο καμβά (canvas element) στην HTML5. Δημιουργήθηκε αρχικά από την Apple για τα γραφικά στοιχεία του πίνακα ελέγχου Mac OS και για την ενεργοποίηση γραφικών στο Safari. Επιπλέον, υιοθετήθηκε από άλλα προγράμματα περιήγησης, όπως το Firefox, το Google Chrome και το Opera.

Από προεπιλογή, το στοιχείο < canvas > έχει πλάτος 300px και ύψος 150px χωρίς περίγραμμα και περιεχόμενο. Ωστόσο, το προσαρμοσμένο πλάτος και ύψος μπορούν να καθοριστούν χρησιμοποιώντας την ιδιότητα ύψους και πλάτους της CSS.

Ο καμβάς είναι μια δισδιάστατη τετράπλευρη περιοχή. Οι συντεταγμένες της επάνω αριστερής γωνίας του καμβά είναι (0, 0), οι οποίες προσδιορίζονται ως προέλευση και οι συντεταγμένες της κάτω δεξιάς γωνίας είναι (*πλάτος καμβά, ύψος καμβά*), όπως φαίνεται με τη χρήση του διαδραστικού εργαλείου που είναι διαθέσιμο εδώ.

Για να σχεδιάσετε βασικές διαδρομές και σχήματα χρησιμοποιώντας το στοιχείο καμβά της HTML5 που μόλις είδαμε και τη JavaScript, μπορείτε να ρίξετε μια ματιά σε πολλά πρότυπα.

Ας δούμε πρώτα το βασικό πρότυπο για τη σχεδίαση διαδρομών και σχημάτων στον δισδιάστατο καμβά της HTML5:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Embedding Canvas Into HTML Pages</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        // draw stuff here
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```

Εικόνα 81 *– Το βασικό πρότυπο για τη σχεδίαση διαδρομών και σχημάτων στον δισδιάστατο καμβά της HTML5*

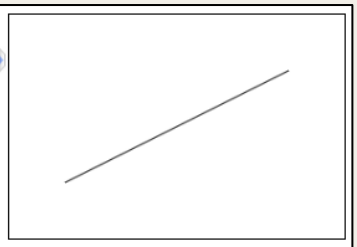
(Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)

Όλες οι σειρές εκτός από αυτές από το 7 έως το 11 είναι αρκετά απλές και κατανοητές. Η ανώνυμη συνάρτηση που έχει ενσωματωθεί στο συμβάν window.onload θα εκτελεστεί όταν φορτώσει η σελίδα. Μόλις φορτώσει η σελίδα, ο χρήστης μπορεί να έχει πρόσβαση στο στοιχείο <canvas> με τη μέθοδο document.getElementById(). Αργότερα, ορίζεται ένα πλαίσιο δισδιάστατου καμβά εισάγοντας το 2d στη μέθοδο getContext() του αντικειμένου του καμβά.

Το πρώτο βήμα για να σχεδιάσετε σε καμβά είναι να σχεδιάσετε μια **ευθεία γραμμή**. Οι πιο σημαντικές διαδικασίες που χρησιμοποιούνται για αυτό το σκοπό είναι οι moveTo(), lineTo() και το stroke(). Η μέθοδος moveTo() προσδιορίζει τη θέση του κέρσορα σχεδίασης στον καμβά, ενώ η μέθοδος lineTo () χρησιμοποιείται για τον καθορισμό των συντεταγμένων σημείου όπου τελειώνει η γραμμή και τέλος η μέθοδος stroke() χρησιμοποιείται για να κάνει τη γραμμή ορατή:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Drawing a Line on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.moveTo(50, 150);
        context.lineTo(250, 50);
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



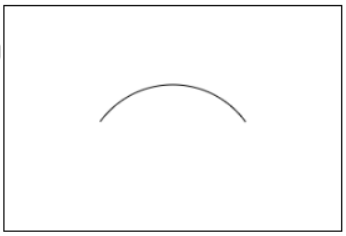
Εικόνα 82 *– Οι διαδικασίες moveTo(), lineTo() και stroke() (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Και πώς μπορούμε **να σχεδιάσουμε μια καμπύλη**; Μπορούμε να τη σχεδιάσουμε χρησιμοποιώντας απλά τη μέθοδο arc(), η οποία συντάσσεται ως εξής:

```
context.arc(centerX, centerY, radius, startingAngle, endingAngle, counterclockwise);
```

Στο παρακάτω παράδειγμα, σχεδιάστηκε μια καμπύλη σε καμβά εισάγοντας έναν κώδικα JavaScript:

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Drawing an Arc on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.arc(150, 150, 80, 1.2 * Math.PI, 1.8 * Math.PI, false);
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



Εικόνα 83 *– Σχεδιασμός καμπύλης με χρήση κώδικα JS (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

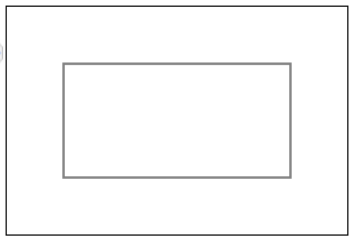
Η κατάλληλη μέθοδος για τον σχεδιασμό ορθογώνιων και τετράγωνων σχημάτων είναι η μέθοδος rect(). Συνεπάγεται τέσσερις παραμέτρους: τις χ, ψ θέσεις του ορθογωνίου, το πλάτος και το ύψος του. Η βασική σύνταξη της μεθόδου rect() είναι η εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
context.rect(x, y, width, height);
```

Για τον σχεδιασμό ορθογωνίου χρησιμοποιώντας έναν κώδικα JS:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Drawing a Rectangle on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.rect(50, 50, 200, 100);
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```

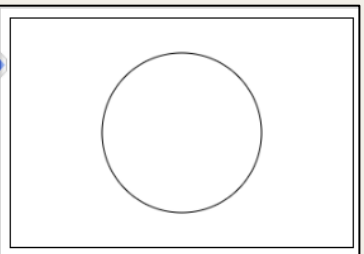


Εικόνα 84 *– Σχεδιασμός ορθογωνίου με χρήση κώδικα JS (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Σε αντίθεση με τη μέθοδο rect(), δεν υπάρχει συγκεκριμένη διαδικασία για τη σχεδίαση ενός κύκλου. Ωστόσο, αυτό μπορεί να επιτευχθεί δημιουργώντας μια πλήρως κλειστή καμπύλη, χρησιμοποιώντας τη μέθοδο arc(). Η σύνταξη για τη σχεδίαση ενός πλήρους κύκλου χρησιμοποιώντας τη μέθοδο arc() είναι η εξής:

```
context.arc(centerX, centerY, radius, 0, 2 * Math.PI, false);
```

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Drawing a Circle on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.arc(150, 100, 70, 0, 2 * Math.PI, false);
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



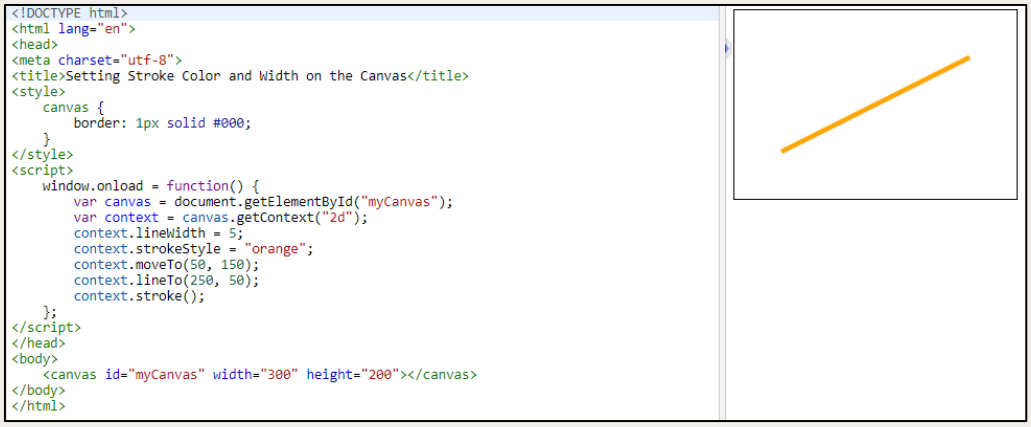
Εικόνα 85 *– Σχεδιάζοντας έναν κύκλο σε καμβά στην HTML5 (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Όσον αφορά το **στυλ και το χρώμα σχεδίασης**, το προεπιλεγμένο χρώμα είναι το μαύρο, με πάχος 1 pixel. Ωστόσο, αυτά τα χαρακτηριστικά μπορούν να αλλάξουν χρησιμοποιώντας τις ιδιότητες strokeStyle και lineWidth, όπως φαίνεται στην *Εικόνα 86*.

Στην *Εικόνα 87*, μπορείτε να δείτε τη δυνατότητα που έχετε να ορίσετε το άκρο του κέρσορα σχεδίασης των γραμμών χρησιμοποιώντας την ιδιότητα lineCap, με τρία διαθέσιμα στυλ: επίπεδο, στρογγυλό και τετράγωνο.

Μπορείτε επίσης να γεμίσετε με χρώμα μέσα στα σχήματα του καμβά χρησιμοποιώντας το fillStyle(). *Η Εικόνα 88* δείχνει πώς να προσθέσετε ένα χρώμα μέσα σε ένα ορθογώνιο σχήμα. Κατά τη σχεδίαση των σχημάτων σε καμβά, προτείνεται η χρήση της μεθόδου fill() πριν από τη μέθοδο stroke() για την κατάλληλη απόδοση του τρόπου σχεδίασης.

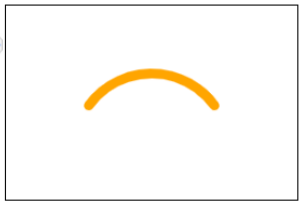


```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Setting Stroke Color and Width on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.lineWidth = 5;
        context.strokeStyle = "orange";
        context.moveTo(50, 150);
        context.lineTo(250, 50);
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```

Εικόνα 86 *– Ρύθμιση των στυλ και των χρωμάτων σχεδίασης χρησιμοποιώντας τις ιδιότητες strokeStyle και lineWidth (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

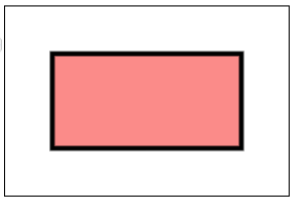
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Setting Stroke Cap Style on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.lineWidth = 10;
        context.strokeStyle = "orange";
        context.lineCap = "round";
        context.arc(150, 150, 80, 1.2 * Math.PI, 1.8 * Math.PI, false);
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



Εικόνα 87 *– Ρύθμιση του στυλ σχεδίασης γραμμών χρησιμοποιώντας την ιδιότητα lineCap (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Filling Color inside a Rectangle on the Canvas</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.rect(50, 50, 200, 100);
        context.fillStyle = "#FB8B89";
        context.fill();
        context.lineWidth = 5;
        context.strokeStyle = "black";
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



Εικόνα 88 *– Αρχειοθέτηση χρώματος μέσα στα σχήματα καμβά χρησιμοποιώντας το fillStyle() (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Μπορείτε επίσης να προσθέσετε χρώμα με βαθμιαία αλλαγή (μια ομαλή οπτική μετάβαση από το ένα χρώμα στο άλλο) μέσα στα σχήματα του καμβά. Υπάρχουν δύο τύποι βαθμιαίων αλλαγών εδώ: γραμμικές και ακτινικές.

Η βασική σύνταξη για τη δημιουργία μιας **γραμμικής διαβάθμισης** είναι:

var grd = context.createLinearGradient (startX, startY, endX, endY);

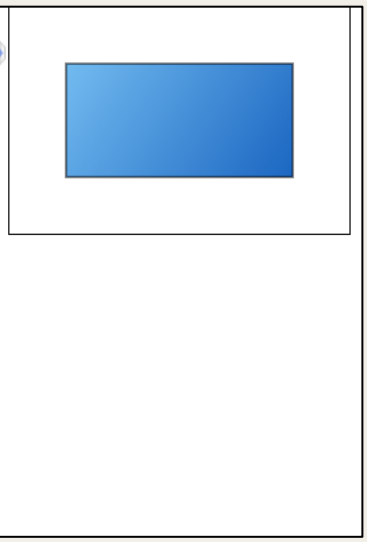
Η βασική σύνταξη για τη δημιουργία μιας **ακτινικής διαβάθμισης** είναι:

```
var grd = context.createLinearGradient(startX, startY, endX, endY);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η εικόνα 89 δείχνει πώς να γεμίσετε ένα χρώμα με **γραμμική διαβάθμιση** μέσα σε ένα ορθογώνιο χρησιμοποιώντας τη μέθοδο createLinearGradient() :

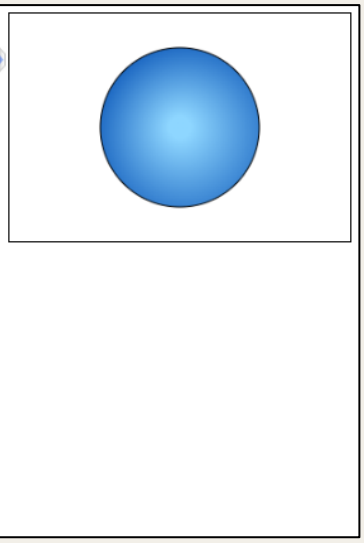
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Filling Linear Gradient inside Canvas Shapes</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.rect(50, 50, 200, 100);
        var grd = context.createLinearGradient(0, 0, canvas.width, canvas.height);
        grd.addColorStop(0, '#8ED6FF');
        grd.addColorStop(1, '#004CB3');
        context.fillStyle = grd;
        context.fill();
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



Εικόνα 89 *– Προσθήκη χρώματος με γραμμική διαβάθμιση μέσα σε ένα ορθογώνιο χρησιμοποιώντας τη μέθοδο createLinearGradient() (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Η Εικόνα 90 δείχνει πώς να γεμίσετε ένα χρώμα με ακτινική διαβάθμιση μέσα σε έναν κύκλο με τη μέθοδο createRadialGradient() :

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Filling Radial Gradient inside Canvas Shapes</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.arc(150, 100, 70, 0, 2 * Math.PI, false);
        var grd = context.createRadialGradient(150, 100, 10, 160, 110, 100);
        grd.addColorStop(0, '#8ED6FF');
        grd.addColorStop(1, '#004CB3');
        context.fillStyle = grd;
        context.fill();
        context.stroke();
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```



Εικόνα 90 *–Προσθήκη χρώματος με ακτινική διαβάθμιση μέσα σε έναν κύκλο με τη μέθοδο createRadialGradient() (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Επιπλέον, μπορείτε να **σχεδιάσετε κείμενο** σε καμβά (που περιέχει μόνο χαρακτήρες Unicode), να προσθέσετε **χρώμα και στοίχιση**, καθώς και πινελιές στο κείμενο

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

χρησιμοποιώντας τη μέθοδο strokeText(), η οποία θα χρωματίσει την περίμετρο του κειμένου αντί να το γεμίσει . Ωστόσο, εάν ο σχεδιαστής ιστοσελίδων επιθυμεί να ορίσει τόσο το χρώμα που θα γεμίσει αλλά και το χρώμα της περιμέτρου του κειμένου, μπορεί να χρησιμοποιήσει τις μεθόδους fillText () και strokeText () μαζί. Προτείνεται η χρήση της μεθόδου fillText () πριν από τη μέθοδο strokeText () για την ακριβή απόδοση του τρόπου σχεδίασης.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Adding Stroke to Canvas Text</title>
<style>
    canvas {
        border: 1px solid #000;
    }
</style>
<script>
    window.onload = function() {
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
        context.font = "bold 32px Arial";
        context.textAlign = "center";
        context.textBaseline = "middle";
        context.strokeStyle = "blue";
        context.strokeText("Code4SP", 150, 100);
    };
</script>
</head>
<body>
    <canvas id="myCanvas" width="300" height="200"></canvas>
</body>
</html>
```

Code4SP

Εικόνα 91 *– Σχεδιασμός κειμένου σε καμβά στην HTML5 (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php)*

Τα γραφικά SVG στην HTML5

Σε αυτή την υποενότητα θα αναλυθεί ο τρόπος χρήσης των στοιχείων SVG της HTML5 για τη σχεδίαση διανυσματικών γραφικών σε μια ιστοσελίδα. Για να γίνει αυτό, πρέπει πρώτα να ορίσουμε τι είναι τα στοιχεία **SVG**.

Το **SVG** σημαίνει Scalable Vector Graphics και είναι μια μορφή εικόνας που βασίζεται στην γλώσσα σήμανσης XML που χρησιμοποιείται για τον καθορισμό δισδιάστατων διανυσματικών γραφικών για τον Ιστό. Σε αντίθεση με τις εικόνες ράστερ (π.χ. .jpg, .gif, .png και άλλες δισδιάστατες μορφές), μια διανυσματική εικόνα μπορεί να κλιμακωθεί προς τα πάνω ή προς τα κάτω σε οποιοδήποτε βαθμό χωρίς να χάσει την ποιότητα της εικόνας. Οι διανυσματικές εικόνες αποτελούνται από μια σειρά σχημάτων που βασίζονται στα μαθηματικά, ενώ οι εικόνες ράστερ αποτελούνται από ένα σταθερό σύνολο κουκκίδων (pixel).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μια εικόνα SVG δημιουργείται χρησιμοποιώντας μια ακολουθία εντολών που συμβαδίζουν με το σχήμα XML (Schema), επομένως, οι **εικόνες SVG** μπορούν να δημιουργηθούν και να τροποποιηθούν με ένα πρόγραμμα επεξεργασίας κειμένου όπως το Notepad. Υπάρχουν πολλά πλεονεκτήματα από τη χρήση εικόνων SVG έναντι άλλων μορφών εικόνας, ως εξής:

- Μπορούν να αναζητηθούν, να ευρετηριαστούν, να δημιουργηθούν σενάρια και να συμπιεστούν.
- Μπορούν να δημιουργηθούν και να τροποποιηθούν χρησιμοποιώντας JavaScript σε πραγματικό χρόνο.
- Μπορούν να εκτυπωθούν με υψηλή ποιότητα σε οποιαδήποτε ανάλυση.
- Μπορούν να γίνουν κινούμενα χρησιμοποιώντας τα ενσωματωμένα στοιχεία κινούμενων γραφικών.
- Μπορούν να περιέχουν υπερσυνδέσμους προς άλλα έγγραφα.

Τα γραφικά SVG μπορούν να ενσωματωθούν απευθείας σε ένα έγγραφο χρησιμοποιώντας το στοιχείο <svg> στην HTML5 (βλ. *Εικόνα 92* παρακάτω).

Όλα τα μεγάλα σύγχρονα προγράμματα περιήγησης ιστού (Chrome, Firefox, Safari και Opera), καθώς και ο Internet Explorer 9 και η νεότερή του έκδοση είναι συμβατά με την ενσωματωμένη απόδοση των SVG.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Embedding SVG Into HTML Pages</title>
<style>
    svg {
        border: 1px solid black;
    }
</style>
</head>
<body>
    <svg width="300" height="200">
        <text x="10" y="20" style="font-size:14px;">
            Your browser support SVG.
        </text>
        Sorry, your browser does not support SVG.
    </svg>
</body>
</html>
```

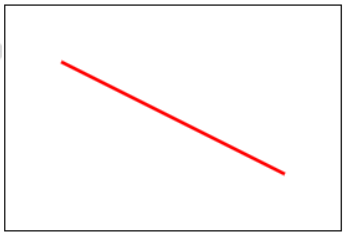
Your browser support SVG.

Εικόνα 92 *– Άμεση ενσωμάτωση γραφικών SVG χρησιμοποιώντας το στοιχείο svg (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Βασικές διαδρομές και σχήματα που βασίζονται σε διανύσματα στις ιστοσελίδες μπορούν να σχεδιαστούν χρησιμοποιώντας το στοιχείο <svg> της HTML5.

Η βασική διαδρομή για να δουλέψουμε με την SVG είναι να **σχεδιάσουμε μια ευθεία γραμμή**. Για να συμβεί αυτό, θα πρέπει να χρησιμοποιηθεί το στοιχείο <line> σε SVG. Όπως φαίνεται στην *Εικόνα 93*, τα χαρακτηριστικά x1, x2, y1 και y2 του στοιχείου σχεδιάζουν μια γραμμή από (x1,y1) έως (x2,y2).

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Create a Line with HTML5 SVG</title>
<style>
    svg {
        border: 1px solid black;
    }
</style>
</head>
<body>
    <svg width="300" height="200">
        <line x1="50" y1="50" x2="250" y2="150" style="stroke:red; stroke-width:3;" />
    </svg>
</body>
</html>
```



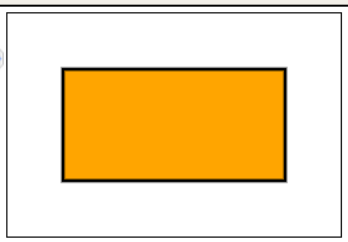
Εικόνα 93 *– Σχεδιάζοντας μια ευθεία γραμμή με το στοιχείο <line> σε SVG (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Ο καταλληλότερος τρόπος για τον σχεδιασμό **ορθογωνίων** και τετραγώνων, είναι το στοιχείο < rect > σε SVG. Τα χαρακτηριστικά x και y του στοιχείου < rect > καθορίζουν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

τις συντεταγμένες της επάνω αριστερής γωνίας του ορθογωνίου. Τα χαρακτηριστικά width και height καθορίζουν το πλάτος και το ύψος του σχήματος.

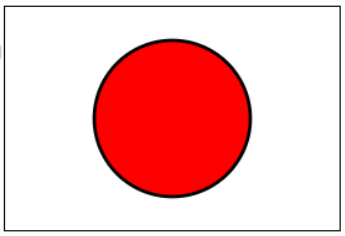
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Create a Rectangle with HTML5 SVG</title>
<style>
    svg {
        border: 1px solid black;
    }
</style>
</head>
<body>
    <svg width="300" height="200">
        <rect x="50" y="50" width="200" height="100" style="fill:orange; stroke:black;
stroke-width:3;" />
    </svg>
</body>
</html>
```



Εικόνα 94 *– Σχεδίαση ορθογωνίου με το στοιχείο <rect> SVG (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Για τον σχεδιασμό ενός **κύκλου,** το πιο κατάλληλο στοιχείο SVG είναι το στοιχείο <circle>. Τα χαρακτηριστικά cx και cy του στοιχείου <circle> καθορίζουν τις συντεταγμένες του κέντρου του κύκλου και το χαρακτηριστικό r προσδιορίζει την ακτίνα του κύκλου. Επίσης, εάν τα χαρακτηριστικά cx και cy απουσιάζουν ή δεν καθορίζονται, το κέντρο του κύκλου ορίζεται σε (0,0).

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Create a Circle with HTML5 SVG</title>
<style>
    svg {
        border: 1px solid black;
    }
</style>
</head>
<body>
    <svg width="300" height="200">
        <circle cx="150" cy="100" r="70" style="fill:red; stroke:black; stroke-width:3;"
/>
    </svg>
</body>
</html>
```



Εικόνα 94 *– Σχεδιασμός κύκλου με το στοιχείο <circle> της SVG (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Έχουμε επίσης τη δυνατότητα να **σχεδιάσουμε κείμενο** σε μορφή SVG. Το κείμενο σε SVG αποδίδεται ως ένα γραφικό, επομένως ο σχεδιαστής Ιστού μπορεί να χρησιμοποιήσει όλο τον μετασχηματισμό γραφικών σε αυτό, ενώ θα εξακολουθεί να λειτουργεί ως κείμενο, ώστε να μπορεί να επιλεγεί και να αντιγραφεί ως κείμενο από τον

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

χρήστη. Τα χαρακτηριστικά x και y του στοιχείου < text > προσδιορίζουν τη θέση της επάνω αριστερής γωνίας σε απόλυτες τιμές, αν και τα χαρακτηριστικά dx και dy υποδηλώνουν τη σχετική θέση.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Render Text with HTML5 SVG</title>
<style>
    svg {
        border: 1px solid black;
    }
</style>
</head>
<body>
    <svg width="300" height="200">
        <text x="20" y="30" style="fill:purple; font-size:22px;">
            Welcome to Our Website!
        </text>
        <text x="20" y="30" dx="0" dy="20" style="fill:navy; font-size:14px;">
            Here you will find lots of useful information.
        </text>
    </svg>
</body>
</html>
```

Welcome to Our Website!
Here you will find lots of useful information.

Εικόνα 95 *– Σχεδιασμός κειμένου με το στοιχείο <text> σε SVG (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Εναλλακτικά, οι σχεδιαστές ιστοσελίδων μπορούν να χρησιμοποιήσουν το στοιχείο <tspan> για να μορφοποιήσετε το κείμενο που περιλαμβάνεται σε ένα στοιχείο <text>. Το κείμενο περιλαμβάνεται σε ξεχωριστά tspan, αλλά μέσα στο ίδιο στοιχείο κειμένου μπορούν να επιλεγούν όλα ταυτόχρονα, όταν κάνετε κλικ και σύρετε για να επιλέξετε το κείμενο. Ωστόσο, τα κείμενα εντός ξεχωριστών στοιχείων <text> δεν μπορούν να επιλεχθούν ταυτόχρονα.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Rotate and Render Text with HTML5 SVG</title>
<style>
    svg {
        border: 1px solid black;
    }
</style>
</head>
<body>
    <svg width="300" height="200">
        <text x="30" y="15" style="fill:purple; font-size:22px; transform:rotate(30deg);">
            <tspan style="fill:purple; font-size:22px;">
                Welcome to Our Website!
            </tspan>
            <tspan dx="-230" dy="20" style="fill:navy; font-size:14px;">
                Here you will find lots of useful information.
            </tspan>
        </text>
    </svg>
</body>
</html>
```

Welcome to Our Website!
Here you will find lots of useful information.

Εικόνα 96 *– Σχεδιασμός κειμένου με το στοιχείο <tspan> SVG (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παρόλο που έχουμε δει τα νέα στοιχεία γραφικών <canvas> και <svg> στην HTML 5, αυτά τα δύο στοιχεία διαφέρουν αρκετά στη δημιουργία γραφικών υψηλής ποιότητας στο διαδίκτυο.

Στον παρακάτω πίνακα, συνοψίζονται οι διαφορές μεταξύ των δύο, για να βοηθήσει τους μαθητές να τα χρησιμοποιήσουν με τον κατάλληλο, αποτελεσματικό τρόπο:

SVG	Canvas
Βασίζεται σε διανύσματα (που αποτελούνται από σχήματα)	Βασίζεται σε ράστερ (που αποτελούνται από pixel)
Πολλά γραφικά στοιχεία, τα οποία γίνονται μέρος του δέντρου DOM της σελίδας	Μεμονωμένο στοιχείο με παρόμοια λειτουργία με το στοιχείο <img> . Το διάγραμμα canvas μπορεί να αποθηκευτεί σε μορφή PNG ή JPG
Τροποποιείται μέσω σεναρίου και CSS	Τροποποιείται μόνο μέσω σεναρίου
Καλές δυνατότητες απόδοσης κειμένου	Κακές δυνατότητες απόδοσης κειμένου
Έχει καλύτερη απόδοση με μικρότερο αριθμό αντικειμένων ή μεγαλύτερη επιφάνεια ή και τα δύο	Έχει καλύτερη απόδοση με μεγαλύτερο αριθμό αντικειμένων ή μικρότερη επιφάνεια ή και με τα δύο
Καλύτερη δυνατότητα κλιμάκωσης. Μπορούν να εκτυπωθούν σε υψηλή ποιότητα σε οποιαδήποτε ανάλυση. Δεν προκύπτει pixelation (παραμόρφωση των εικονοστοιχείων)	Μικρή δυνατότητα κλιμάκωσης. Δεν είναι κατάλληλο για εκτύπωση σε υψηλότερη ανάλυση. Ενδέχεται να προκύψει pixelation (παραμόρφωση των εικονοστοιχείων)

Πίνακας 4 *– Οι διαφορές μεταξύ SVG και Canvas (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-svg.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ήχος στην HTML 5

Αυτή η υποενότητα έχει σκοπό να εξηγήσει πώς να ενσωματώσετε τον ήχο σε ένα έγγραφο HTML.
Καθώς τα προγράμματα περιήγησης ιστού δεν διέθεταν στο παρελθόν ένα ενιαίο πρότυπο για την ενσωμάτωση αρχείων πολυμέσων ως ήχου, η ενσωμάτωση ήχου δεν ήταν εύκολη υπόθεση. Ωστόσο, τώρα υπάρχουν πολλοί τρόποι για να ενσωματώσετε τον ήχο σε μια ιστοσελίδα, από την απλή χρήση ενός απλού συνδέσμου έως τη χρήση του στοιχείου <audio> της HTML5. Αυτό το στοιχείο παρέχει έναν τυπικό τρόπο εισαγωγής ήχου σε ιστοσελίδες. Δεδομένου ότι το στοιχείο ήχου είναι κάπως καινούργιο, εκτελείται στα περισσότερα σύγχρονα προγράμματα περιήγησης ιστού.

Υπάρχουν πολλοί τρόποι εισαγωγής ήχου σε ένα έγγραφο HTML5. Ένα από αυτά είναι η χρήση της προεπιλεγμένης ομάδας στοιχείων ελέγχου του προγράμματος περιήγησης, με μια πηγή που ορίζεται από το χαρακτηριστικό src, όπως μπορεί να επαληθευτεί σε αυτόν τον κώδικα, στον οποίο μπορείτε να ακούσετε μερικά πουλιά να κελαηδούν.

Ένας άλλος τρόπος μπορεί να επιτευχθεί χρησιμοποιώντας το στοιχείο <object>, το οποίο χρησιμοποιείται για την ενσωμάτωση διαφορετικών τύπων αρχείων πολυμέσων. Αυτό το παράδειγμα ενσωματώνει ένα αρχείο ήχου σε μια ιστοσελίδα ακολουθώντας την προαναφερθείσα μέθοδο. Ωστόσο, το στοιχείο <object> δεν υποστηρίζεται ευρέως και εξαρτάται από τον τύπο του αντικειμένου που προστίθεται. Άλλες μέθοδοι όπως το στοιχείο <audio> της HTML5 ή οι εξωτερικές συσκευές αναπαραγωγής ήχου HTML5 μπορεί να αποτελούν καλύτερη επιλογή σε πολλές περιπτώσεις.

Τέλος, το στοιχείο <embed> μπορεί επίσης να αποτελέσει τρόπο εισαγωγής μέσων σε ένα έγγραφο HTML, ακολουθώντας αυτό το παράδειγμα. Παρόλο που το στοιχείο <embed> υποστηρίζεται σε πολύ μεγάλο βαθμό στα σημερινά προγράμματα περιήγησης και ορίζεται ως το καθιερωμένο στην HTML5, ο ήχος που θα εισαχθεί ενδέχεται να μην αναπαράγεται καθώς μπορεί η μορφή του αρχείου να μην

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

υποστηρίζεται από το πρόγραμμα περιήγησης ή να μην υπάρχει πρόσβαση στις προσθήκες.

Βίντεο στην HTML5

Ας μάθουμε τώρα πώς γίνεται η ενσωμάτωση βίντεο σε ένα έγγραφο HTML.

Όπως και στην περίπτωση του ήχου, τα αρχεία με βίντεο ήταν επίσης δύσκολο να εισαχθούν σε μια ιστοσελίδα και για τον ίδιο λόγο (τα προγράμματα περιήγησης ιστού δεν είχαν ένα ενιαίο πρότυπο για τον ορισμό των ενσωματωμένων αρχείων πολυμέσων όπως το βίντεο). Στις επόμενες παραγράφους, θα επεξηγηθούν διάφοροι τρόποι εισαγωγής αυτών των αρχείων.

Το στοιχείο <video>, το οποίο εισήχθη πρόσφατα λειτουργεί στα περισσότερα από τα σύγχρονα προγράμματα περιήγησης. Αυτό το παράδειγμα εξηγεί πώς μπορείτε να εισαγάγετε ένα βίντεο με απλό τρόπο σε ένα έγγραφο HTML, χρησιμοποιώντας το προεπιλεγμένο σύνολο εντολών του προγράμματος περιήγησης, με μία πηγή που ορίζεται από το χαρακτηριστικό src .

Το στοιχείο <object> χρησιμοποιείται επίσης για την ενσωμάτωση διαφορετικών τύπων αρχείων πολυμέσων. Ακολουθώντας αυτό το παράδειγμα, είναι εύκολο να καταλάβουμε πως να ενσωματώσουμε ένα βίντεο Flash σε μια ιστοσελίδα (μόνο τα προγράμματα περιήγησης/εφαρμογές που υποστηρίζουν το Flash θα μπορούν να το αναπαράγουν). Πρέπει να σημειωθεί ότι το <object> στοιχείο δεν υποστηρίζεται εκτενώς και εξαρτάται πολύ από τον τύπο του ενσωματωμένου αντικειμένου. Άλλες μέθοδοι θα μπορούσαν να αποτελέσουν καλύτερη επιλογή σε πολλές περιπτώσεις, καθώς, συσκευές όπως iPad και iPhone δεν μπορούν να αναπαράγουν βίντεο Flash.

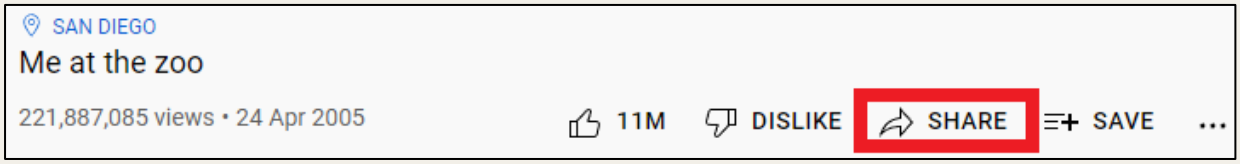
Πώς ενσωματώνουμε **βίντεο από το YouTube**; Αυτός είναι ο πιο απλός και συνηθισμένος τρόπος ενσωμάτωσης αρχείων βίντεο σε ιστοσελίδες στις μέρες μας. Ο σχεδιαστής ιστοσελίδων πρέπει απλώς να ανεβάσει το βίντεο στο YouTube και να

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

εισαγάγει έναν κώδικα HTML για να εμφανίσει το βίντεο στην ιστοσελίδα του. Ακολουθεί ένας μικρός οδηγός βήμα προς βήμα:

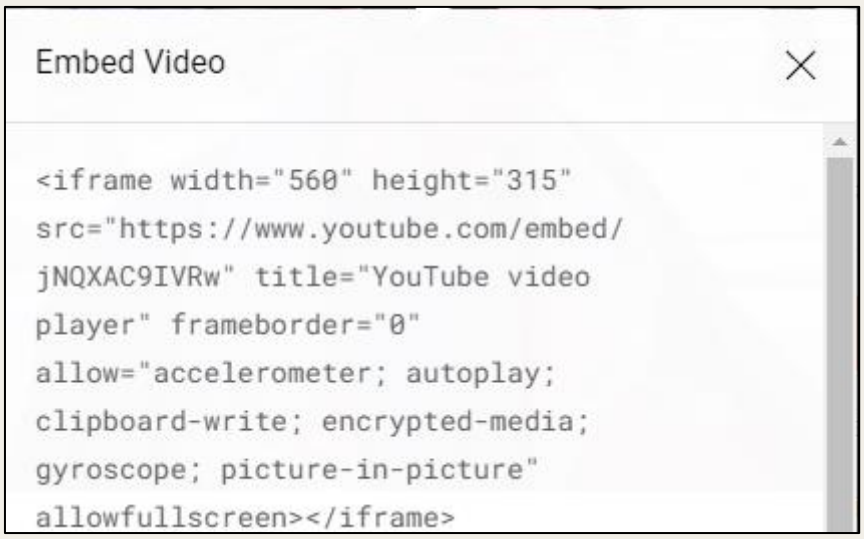
Βήμα 1 – Ανεβάστε ένα βίντεο στο YouTube.

Βήμα 2 – Μετά τη μεταφόρτωση του βίντεο στο YouTube, ο σχεδιαστής ιστού θα πρέπει να αναζητήσει το κουμπί «Κοινοποίηση» (Share), το οποίο βρίσκεται κάτω από το βίντεο στο πρόγραμμα αναπαραγωγής βίντεο της πλατφόρμας, όπως ακριβώς φαίνεται στο παράδειγμα:



Εικόνα 97 *– Κουμπί «Κοινοποίησης» στο YouTube (Πηγή: Συντάκτης)*

Όταν κάνετε κλικ στο κουμπί «Κοινοποίηση», θα ανοίξει ένα πλαίσιο για να κοινοποιήσετε το βίντεο που εμφανίζει μερικές ακόμη επιλογές. Τώρα, θα πρέπει να κάνετε κλικ στο κουμπί «**Ενσωμάτωση βίντεο**» (Embed video), το οποίο θα δημιουργήσει τον κώδικα HTML για την απευθείας ενσωμάτωση του βίντεο στην ιστοσελίδα. Για να συμβεί αυτό, ο σχεδιαστής Ιστού θα πρέπει να αντιγράψει και να επικολλήσει αυτόν τον κώδικα στο έγγραφο HTML.



Εικόνα 98 *– Επιλογή «Ενσωμάτωσης βίντεο» στο YouTube (Πηγή: Συντάκτης)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αυτός ο κώδικας μπορεί να προσαρμοστεί περαιτέρω επιλέγοντας την επιλογή προσαρμογής που δίνεται ακριβώς κάτω από το πλαίσιο εισαγωγής κώδικα ενσωμάτωσης.

Η εισαγωγή ενός βίντεο YouTube σε μια ιστοσελίδα εξηγείται σε αυτό το παράδειγμα.

Αποθήκευση δεδομένων στον ιστό στην HTML5 (Web storage)

Αναρωτηθήκατε ποτέ πώς μπορείτε να χρησιμοποιήσετε τη δυνατότητα αποθήκευσης δεδομένων στον ιστό στην HTML5 για την αποθήκευσή τους στο πρόγραμμα περιήγησης του χρήστη; Οι ακόλουθες παράγραφοι θα σας βοηθήσουν να κατανοήσετε πώς γίνεται αυτό.

Πρώτον, είναι σημαντικό να κατανοήσουμε τα θέματα που προκύπτουν με την «αποθήκευση δεδομένων στον ιστό» (*web storage*).

Με την αποθήκευση δεδομένων στον ιστό , οι διαδικτυακές εφαρμογές μπορούν να αποθηκεύουν δεδομένα τοπικά στο πρόγραμμα περιήγησης του χρήστη. Πριν από την HTML5, τα δεδομένα της εφαρμογής έπρεπε να αποθηκεύονται σε cookies, ενσωματωμένα σε κάθε αίτημα διακομιστή. Η αποθήκευση στον ιστό είναι πιο ασφαλής, ενώ μπορούν να αποθηκευτούν σημαντικές ποσότητες δεδομένων, χωρίς να επηρεάζεται η απόδοση του ιστότοπου. Οι πληροφορίες που διατηρούνται στο χώρο όπου αποθηκεύονται δεδομένα στον ιστό δεν αποστέλλονται στον διακομιστή ιστού, σε αντίθεση με τα cookies όπου τα δεδομένα παραδίδονται στον διακομιστή με κάθε αίτημα. Επιπλέον, τα cookies επιτρέπουν την αποθήκευση μόνο ενός μικρού όγκου δεδομένων (σχεδόν 4 KB), ενώ ο χώρος αποθήκευσης δεδομένων στον ιστό επιτρέπει έως και 5 MB.

Υπάρχουν δύο τύποι αποθήκευσης δεδομένων στον ιστό:

- **Τοπική αποθήκευση περιήγησης *(Local storage)*** — χρησιμοποιεί το αντικείμενο localStorage για την αποθήκευση δεδομένων για ολόκληρο τον ιστότοπο σε μια *μόνιμη βάση*. Επομένως, τα αποθηκευμένα τοπικά δεδομένα θα

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

είναι διαθέσιμα την επόμενη ημέρα, την επόμενη εβδομάδα ή το επόμενο έτος, εκτός εάν καταργηθούν.

- **Αποθήκευση περιόδου λειτουργίας *(Session Storage)*** — χρησιμοποιεί το αντικείμενο sessionStorage για την αποθήκευση δεδομένων σε μια *προσωρινή βάση*, για ένα μόνο παράθυρο ή καρτέλα του προγράμματος περιήγησης. Τα δεδομένα εξαφανίζονται όταν τελειώνει η περίοδος λειτουργίας, για παράδειγμα όταν ο χρήστης κλείνει αυτό το παράθυρο ή την καρτέλα του προγράμματος περιήγησης.

Όσον αφορά την **τοπική αποθήκευση περιήγησης (Local storage)**, κάθε στοιχείο δεδομένων συλλέγεται σε ένα ζεύγος κλειδιού/τιμής. Το κλειδί προσδιορίζει το όνομα της πληροφορίας (π.χ. «first_name») και η τιμή είναι η τιμή που σχετίζεται με το ίδιο κλειδί (δηλ. «Peter»). Ο παρακάτω κώδικας JS εκφράζει τα εξής:

- TolocalStorage.setItem (key, value) αποθηκεύει την τιμή που σχετίζεται με ένα κλειδί.
- TolocalStorage.getItem (key) αποθηκεύει την τιμή που σχετίζεται με το κλειδί.

Μπορείτε επίσης να αφαιρέσετε ένα συγκεκριμένο στοιχείο από την αποθήκευση, προσθέτοντας το όνομα του κλειδιού στη μέθοδο removeItem(), π.χ. localStorage.removeItem("first_name").

Ωστόσο, εάν ο σχεδιαστής ιστοσελίδων θέλει να αφαιρέσει ολόκληρο τον χώρο αποθήκευσης, θα πρέπει να χρησιμοποιήσει τη μέθοδο clear(), δηλαδή localStorage.clear(). Η μέθοδος clear() απλώς διαγράφει όλα τα ζεύγη κλειδιών/τιμών από το localStorage ταυτόχρονα, **επομένως πρέπει να χρησιμοποιείται με προσοχή**. Τα δεδομένα που είναι αποθηκευμένα στον ιστό δεν θα είναι προσβάσιμα μεταξύ διαφορετικών προγραμμάτων περιήγησης.

Τέλος, το αντικείμενο sessionStorage λειτουργεί με παρόμοιο τρόπο με το localStorage, με τη διαφορά ότι αποθηκεύει τα δεδομένα μόνο για μία περίοδο λειτουργίας. Το παράδειγμα σε αυτό τον σύνδεσμο εξηγεί λεπτομερώς πώς λειτουργεί αυτό.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εφαρμογές προσωρινής μνήμης στην HTML5 (Application Cache)

Σε αυτή την υποενότητα, οι εκπαιδευόμενοι θα έχουν τη δυνατότητα να μάθουν για τον τρόπο δημιουργίας εφαρμογών εκτός σύνδεσης χρησιμοποιώντας τη **λειτουργία προσωρινής αποθήκευσης στην HTML5**.

Γνωρίζουμε ήδη ότι οι περισσότερες εφαρμογές που παρέχονται μέσω του ιστού δεν λειτουργούν εάν ο σχεδιαστής ιστοσελίδων είναι εκτός σύνδεσης. Ωστόσο, η HTML5 εισήγαγε έναν μηχανισμό προσωρινής αποθήκευσης εφαρμογών που επιτρέπει στο πρόγραμμα περιήγησης να αποθηκεύει αυτόματα το έγγραφο HTML και όλους τους άλλους πόρους που απαιτούνται για να το εμφανίσει σωστά στον τοπικό υπολογιστή, με αυτόν τον τρόπο το πρόγραμμα περιήγησης μπορεί να έχει πρόσβαση στην ιστοσελίδα και τους πόρους της χωρίς σύνδεση στο διαδίκτυο. Αυτό υποστηρίζεται σε όλα τα μεγάλα σύγχρονα προγράμματα περιήγησης ιστού (Firefox, Chrome, Opera, Safari και Internet Explorer 10 και άνω.

Προκύπτουν πολλά πλεονεκτήματα από τη χρήση αυτής της δυνατότητας:

- **Περιήγηση εκτός σύνδεσης —** Οι επισκέπτες μπορούν να χρησιμοποιήσουν την εφαρμογή ακόμη και όταν δεν είναι συνδεδεμένοι στο διαδίκτυο ή υπάρχουν απροσδόκητες διακοπές στη σύνδεση δικτύου.
- **Βελτίωση απόδοσης —** Οι αποθηκευμένοι πόροι φορτώνονται απευθείας από το μηχάνημα του χρήστη αντί από τον απομακρυσμένο διακομιστή, ώστε οι ιστοσελίδες να φορτώνουν πιο γρήγορα και να έχουν καλύτερη απόδοση.
- **Μείωση αιτημάτων HTTP και φόρτου του διακομιστή —** Το πρόγραμμα περιήγησης πρέπει μόνο να πραγματοποιήσει τη λήψη των ενημερωμένων/τροποποιημένων πόρων από τον απομακρυσμένο διακομιστή, μειώνοντας τα αιτήματα HTTP και εξοικονομώντας πολύτιμο εύρος ζώνης, καθώς μειώνουν επίσης τον φόρτο στον διακομιστή ιστού.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Υπάρχουν μερικά βήματα που πρέπει να ακολουθήσετε για να αποθηκεύσετε προσωρινά τα αρχεία για χρήση εκτός σύνδεσης και συγκεκριμένα:

Υπάρχουν μερικά βήματα που πρέπει να ακολουθήσετε για να αποθηκεύσετε τα αρχεία στην προσωρινή μνήμη για χρήση εκτός σύνδεσης:

ΒΗΜΑ 1 – Δημιουργήστε ένα αρχείο Cache Manifest. Αυτό είναι ένα ειδικό αρχείο κειμένου που ενημερώνει τα προγράμματα περιήγησης ποια αρχεία πρέπει να αποθηκεύουν (και ποια όχι) και ποια αρχεία πρέπει να αντικαταστήσουν. Ξεκινά πάντα με τις λέξεις CACHE MANIFEST (πάντα με κεφαλαία).

Η Εικόνα 99 παρακάτω είναι ένα παράδειγμα ενός αρχείου manifest:

Example ⬇ Download

```
CACHE MANIFEST
# v1.0 : 10-08-2014

CACHE:
# pages
index.html

# styles & scripts
css/theme.css
js/jquery.min.js
js/default.js

# images
/favicon.ico
images/logo.png

NETWORK:
login.php

FALLBACK:
/ /offline.html
```

Εικόνα 99 *– Παράδειγμα αρχείου manifest (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-application-cache.php)*

Ας εξηγήσουμε τώρα τον κώδικα που χρησιμοποιήθηκε στην εικόνα της τελευταίας διαφάνειας.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Πρώτον, είναι σημαντικό να κατανοήσουμε ότι τα αρχεία manifest μπορούν να έχουν τρία διαφορετικά τμήματα**: CACHE , NETWORK** και **FALLBACK**.
- Τα αρχεία που παρατίθενται κάτω από την κεφαλίδα του CACHE: (ή αμέσως μετά τη γραμμή CACHE MANIFEST) αποθηκεύονται μόνο στην προσωρινή μνήμη μετά την πρώτη τους λήψη.
- Τα αρχεία κάτω από το NETWORK: είναι πόροι στη λευκή λίστα που δεν αποθηκεύονται ποτέ στην προσωρινή μνήμη και είναι διαθέσιμοι μόνο στο διαδίκτυο. Σημαίνει ότι οι χρήστες δεν μπορούν ποτέ να έχουν πρόσβαση στη σελίδα login.php όταν είναι εκτός σύνδεσης.
- Το FALLBACK: καθορίζει εναλλακτικές σελίδες που θα πρέπει να χρησιμοποιεί το πρόγραμμα περιήγησης σε περίπτωση που δεν μπορεί να γίνει η σύνδεση με τον διακομιστή. Κάθε καταχώρηση σε αυτήν την ενότητα παραθέτει δύο URI. Το πρώτο είναι ο κύριος πόρος και το δεύτερο είναι ο εναλλακτικός. Για παράδειγμα, στην περίπτωση της *Εικόνας 98*, η σελίδα offline.html θα εμφανιστεί εάν ο χρήστης είναι εκτός σύνδεσης. Επίσης, και οι δύο URI πρέπει να προέρχονται από την ίδια προέλευση με το αρχείο manifest.
- Θα πρέπει να σημειωθεί ότι οι γραμμές που ξεκινούν με το σύμβολο '#' είναι γραμμές με σχόλια.

Επομένως, εάν υπάρχει μια εφαρμογή προσωρινής μνήμης, το πρόγραμμα περιήγησης φορτώνει το έγγραφο και τους σχετικούς πόρους απευθείας από την προσωρινή μνήμη, χωρίς να έχει πρόσβαση στο διαδίκτυο. Στη συνέχεια, το πρόγραμμα περιήγησης ελέγχει για να διαπιστώσει εάν το αρχείο manifest έχει ενημερωθεί στον διακομιστή. Εάν έχει ενημερωθεί, το πρόγραμμα περιήγησης κατεβάζει τη νέα έκδοση του αρχείου και τους πόρους που αναφέρονται σε αυτό.

Είναι σημαντικό να σημειωθεί ότι το ίδιο το αρχείο manifest δεν πρέπει να προσδιορίζεται στο αρχείο Cache Manifest. Εάν γίνει αυτό, θα είναι αρκετά δύσκολο να ειδοποιηθεί το πρόγραμμα περιήγησης ότι είναι διαθέσιμο ένα νέο αρχείο manifest.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

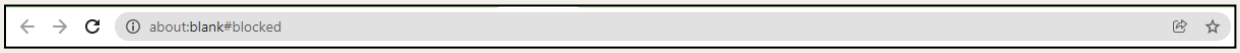
ΒΗΜΑ 2 – Χρησιμοποιήστε το αρχείο cache manifest. Μετά τη δημιουργία του, ο σχεδιαστής ιστού θα πρέπει να ανεβάσει το αρχείο cache manifest στον διακομιστή ιστού. Βεβαιωθείτε ότι ο διακομιστής ιστού είναι ρυθμισμένος έτσι ώστε να υποστηρίζει τα αρχεία manifest με το MIME type text/cache-manifest.

Για να λειτουργήσει το αρχείο cache manifest, ο σχεδιαστής ιστοσελίδων θα πρέπει να το ενεργοποιήσει στις ιστοσελίδες, προσθέτοντας το χαρακτηριστικό manifest στο βασικό στοιχείο, το < html >, όπως φαίνεται στην *Εικόνα 100* παρακάτω:

```
1  <!DOCTYPE html>
2  <html lang="en" manifest="example.appcache">
3  <head>
4      <title>Using the Application Cache</title>
5  </head>
6  <body>
7      <!--The document content will be inserted here-->
8  </body>
9  </html>
```

Εικόνα 100 *– Θέτοντας σε λειτουργία το αρχείο cache manifest (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-application-cache.php)*

Εάν ο χρήστης είναι συνδεδεμένος με το διαδίκτυο, το αποτέλεσμα που θα προκύψει από αυτόν τον κώδικα θα είναι το εξής:



Εικόνα 101 *– about_blank#blocked (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-application-cache.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Web Workers στην HTML5

Αυτή η υποενότητα θα είναι χρήσιμη για την εμβάθυνση σε θέματα JS, καθώς θα δείξει στους εκπαιδευόμενους πώς να χρησιμοποιούν τα web workers της HTML5 για την εκτέλεση κώδικα JS. Έτσι, οι εκπαιδευόμενοι μπορούν να ξαναδούν τις οδηγίες εάν αντιμετωπίσουν οποιεσδήποτε δυσκολίες.

Αν κάποιος προσπαθήσει να εκτελέσει εντατικούς, χρονοβόρους και απαιτητικούς υπολογισμούς που απαιτούν εργασίες με χρήση κώδικα JavaScript, πιθανότατα θα παγώσει τις ιστοσελίδες και θα εμποδίσει τους χρήστες από το να κάνουν οτιδήποτε μέχρι να ολοκληρωθεί η εργασία. Γιατί; Λοιπόν, επειδή ο κώδικας JS τρέχει πάντα στο προσκήνιο. Ωστόσο, η HTML5 διαθέτει μια νέα τεχνολογία («web worker») που δημιουργήθηκε για να εκτελεί εργασίες στο παρασκήνιο εκτός από άλλα σενάρια διεπαφής χρήστη, χωρίς να επηρεάζει την απόδοση της σελίδας. Σε αντίθεση με τις κανονικές λειτουργίες JS, το web worker δεν διακόπτει τον χρήστη και η ιστοσελίδα συνεχίζει να λειτουργεί επειδή εκτελεί τις εργασίες στο παρασκήνιο.

Τα Web workers είναι ιδιαίτερα χρήσιμα για την εκτέλεση μιας χρονοβόρας εργασίας. Έτσι, στο πρώτο παράδειγμα, εκτελείται μια απλή εργασία JS που μετράει από το μηδέν έως το 100 000 *(το όνομα του αρχείου θα πρέπει να είναι worker.js)*, όπως φαίνεται στην *Εικόνα 102:*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
1  var i = 0;
2  function countNumbers() {
3      if(i < 100000) {
4          i = i + 1;
5          postMessage(i);
6      }
7
8      // Wait for sometime before running this script again
9      setTimeout("countNumbers()", 500);
10 }
11 countNumbers();
```

Εικόνα 102 *– Δημιουργία εργασίας στην JS που μετράει από το 0 έως το 100.000 (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-web-workers.php)*

Σημείωση: Για καλύτερη κατανόηση, συνιστάται να κάνετε λήψη του κώδικα από την Εικόνα 101 και να ακολουθήσετε όλα τα βήματα αυτού του κεφαλαίου.

Έτσι, τώρα που δημιουργήθηκε το αρχείο web worker, ήρθε η ώρα να ξεκινήσουμε το web worker από ένα έγγραφο HTML που εκτελεί τον κώδικα μέσα στο αρχείο με το όνομα «worker.js» στο παρασκήνιο και εμφανίζει σταδιακά το αποτέλεσμα στην ιστοσελίδα. Πρέπει να σημειωθεί ότι ο αριθμός στα δεξιά θα αυξάνεται πάντα μέχρι να φτάσει το 100.000.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="utf-8">
5  <title>Using HTML5 Web Workers</title>
6  <script>
7      if(window.Worker) {
8          // Create a new web worker
9          var worker = new Worker("/examples/js/worker.js");
10
11         // Fire onMessage event handler
12         worker.onmessage = function(event) {
13             document.getElementById("result").innerHTML = event.data;
14         };
15     } else {
16         alert("Sorry, your browser do not support web worker.");
17     }
18 </script>
19 </head>
20 <body>
21     <div id="result">
22         <!--Received messages will be inserted here-->
23     </div>
24 </body>
25 </html>
```

26

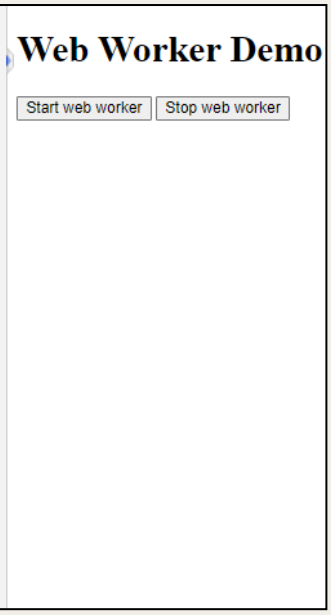
Εικόνα 103 *– Εκκίνηση του web worker (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-web-workers.php)*

Τώρα για να εξηγήσουμε τι συμβαίνει στο παράδειγμα της προηγούμενης διαφάνειας, η εντολή **var worker = new Worker("worker.js");** δημιουργεί ένα νέο αντικείμενο web worker, το οποίο χρησιμοποιείται για την επικοινωνία με τον web worker. Όταν ο web worker δημοσιεύει ένα μήνυμα, ενεργοποιεί το πρόγραμμα χειρισμού συμβάντων **onmessage** (γραμμή 14) που επιτρέπει στον κώδικα να λαμβάνει μηνύματα από τον web worker. Το στοιχείο **event.data** περιλαμβάνει το μήνυμα που αποστέλλεται από τον web worker. Για την ακρίβεια, ο κώδικας που εκτελεί ένας web worker αποθηκεύεται πάντα σε ένα ξεχωριστό αρχείο JavaScript για να εμποδίσει τον προγραμματιστή ιστού από το να γράψει τον κώδικα του web worker που επιχειρεί να χρησιμοποιήσει παγκόσμιες μεταβλητές ή να ανοίξει απευθείας τα στοιχεία στην ιστοσελίδα.

Είναι επίσης δυνατό **να τερματιστεί ένας τρέχων web worker** στη μέση της λειτουργίας του, ακολουθώντας το παρακάτω παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Start/Stop Web Worker in HTML5</title>
<script>
    // Set up global variable
    var worker;

    function startWorker() {
        // Initialize web worker
        worker = new Worker("/examples/js/worker.js");

        // Run update function, when we get a message from worker
        worker.onmessage = update;

        // Tell worker to get started
        worker.postMessage("start");
    }

    function update(event) {
        // Update the page with current message from worker
        document.getElementById("result").innerHTML = event.data;
    }

    function stopWorker() {
        // Stop the worker
        worker.terminate();
    }
</script>
</head>
```



Εικόνα 104 *– Διακοπή του τρέχοντος web worker (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-web-workers.php)*

Το παραπάνω παράδειγμα δείχνει πώς να ξεκινήσετε και να τερματίσετε τον web worker από μια ιστοσελίδα χρησιμοποιώντας απλά τα κουμπιά της HTML.

Αποστολή συμβάντων από τον διακομιστή στην HTML5 (Server-Sent Events)

Αυτή η υποενότητα θα είναι χρήσιμη για την κατανόηση του τρόπου χρήσης της δυνατότητας αποστολής συμβάντων (SSE) από τον διακομιστή στην HTML5 για τη δημιουργία μιας μονοκατευθυντικής και μόνιμης σύνδεσης μεταξύ μιας ιστοσελίδας και ενός διακομιστή.

Η SSE στην HTML5 είναι ένας καινοτόμος τρόπος επικοινωνίας των ιστοσελίδων με έναν διακομιστή ιστού. Ωστόσο, υπάρχουν ορισμένες περιπτώσεις όπου οι ιστοσελίδες χρειάζονται μια σύνδεση μεγαλύτερης διάρκειας με τον διακομιστή ιστού, για παράδειγμα, σε τιμές μετοχών σε ιστότοπους με χρηματοοικονομικά θέματα όπου οι τιμές ενημερώνονται αυτόματα ή ο χρόνος παιχνιδιών σε διάφορους αθλητικούς ιστότοπους. Αυτά γίνονται εφικτά με την SSE στην HTML5, καθώς δίνει τη δυνατότητα σε μια ιστοσελίδα να διατηρεί ανοιχτή σύνδεση με έναν διακομιστή ιστού, με τρόπο που

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ο διακομιστής Ιστού μπορεί να στείλει μια νέα απάντηση μηχανικά ανά πάσα στιγμή. Σε αυτό το σημείο, δεν χρειάζεται κάθε να επανασυνδέεστε κάθε φορά και να εκτελείτε το ίδιο σενάριο διακομιστή από την αρχή.

Για να κατανοήσετε καλύτερα τις προαναφερθέντες έννοιες, ανοίξτε ένα αρχείο PHP[1] με το όνομα «server_time.php» και πληκτρολογήστε το ακόλουθο σενάριο σε αυτό. Αυτό το αρχείο θα αναφέρει απλώς την τρέχουσα ώρα του ενσωματωμένου ρολογιού του διακομιστή Ιστού σε τακτά χρονικά διαστήματα:

```php
<?php
header("Content-Type: text/event-stream");
header("Cache-Control: no-cache");

// Get the current time on server
$currentTime = date("h:i:s", time());

// Send it in a message
echo "data: " . $currentTime . "\n\n";
flush();
?>
```

Εικόνα 105 *– παράδειγμα του server_time.php (Πηγή https://www.tutorialrepublic.com/html-tutorial/html5-server-sent-events.php)*

Οι δύο πρώτες γραμμές του σεναρίου του PHP ορίζουν δύο πολύ σημαντικές κεφαλίδες. Η πρώτη γραμμή, ορίζει τον τύπο MIME σε ροή κειμένου/συμβάντος (text/event-stream), που απαιτείται από το πρότυπο SSE. Η δεύτερη γραμμή ενημερώνει τον διακομιστή Ιστού να απενεργοποιήσει την προσωρινή μνήμη, διαφορετικά το σενάριο που θα εξαχθεί μπορεί να αποθηκευτεί προσωρινά.

Κάθε μήνυμα που αποστέλλεται μέσω SSE στην HTML5 πρέπει να ξεκινά με το στοιχείο data: ακολουθούμενο από το πραγματικό κείμενο του μηνύματος και τη νέα ακολουθία χαρακτήρων σε νέα γραμμή (\n\n). Και τέλος, η συνάρτηση flush() στο PHP έχει χρησιμοποιηθεί για να διασφαλιστεί ότι τα δεδομένα αποστέλλονται αμέσως, αντί να αποθηκεύονται στην προσωρινή μνήμη μέχρι να ολοκληρωθεί ο κώδικας PHP.

[1] Ένα αρχείο PHP είναι ένα αρχείο απλού κειμένου που περιέχει κώδικα γραμμένο στη γλώσσα προγραμματισμού PHP. Δεδομένου ότι η PHP είναι μια γλώσσα προγραμματισμού από την πλευρά του διακομιστή (back-end), ο κώδικας που είναι γραμμένος σε αυτήν εκτελείται στον διακομιστή. Στην πραγματικότητα, ένα αρχείο PHP μπορεί να περιέχει απλό κείμενο, ετικέτες HTML ή κώδικα σύμφωνα με τη σύνταξη της PHP. Η PHP χρησιμοποιείται συνήθως για την ανάπτυξη διαδικτυακών εφαρμογών που υποβάλλονται σε επεξεργασία από μια μηχανή PHP στον διακομιστή Ιστού.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τώρα όσον αφορά **στον τρόπο επεξεργασίας μηνυμάτων σε μια ιστοσελίδα**, το αντικείμενο EventΠηγή χρησιμοποιείται για τη λήψη μηνυμάτων από τα συμβάντα που αποστέλλονται από τον διακομιστή. Στο παράδειγμα που ακολουθεί, οι εκπαιδευόμενοι θα δουν πώς ένα έγγραφο HTML λαμβάνει απλώς την τρέχουσα ώρα που αναφέρεται από τον διακομιστή ιστού και την εμφανίζει στους επισκέπτες της ιστοσελίδας. Για καλύτερη κατανόηση, θα δημιουργηθεί ένα έγγραφο HTML με το όνομα «demo_sse.htlm» και στη συνέχεια θα τοποθετηθεί στον ίδιο κατάλογο του έργου όπου βρίσκεται το «server_time.php».

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Using Server-Sent Events</title>
<script>
    window.onload = function() {
        var source = new EventSource("server_time.php");
        source.onmessage = function(event) {
            document.getElementById("result").innerHTML += "New time received
from web server: " + event.data + "<br>";
        };
    };
</script>
</head>
<body>
    <div id="result">
        <!--Server response will be inserted here-->
    </div>
</body>
</html>
```

***Εικόνα 106** – Τρόπος επεξεργασίας μηνυμάτων σε μια ιστοσελίδα (Πηγή https://www.tutorialrepublic.com/html-tutorial/html5-server-sent-events.php)*

Γεωτοποθεσία και γεωεντοπισμός μέσω HTML5 (Geolocation)

Μέσα από αυτή την υποενότητα, οι εκπαιδευόμενοι θα λάβουν μερικές πληροφορίες σχετικά με τον τρόπο χρήσης της δυνατότητας Γεωεντοπισμού μέσω HTML5 για τον εντοπισμό της τοποθεσίας του χρήστη. Αυτή η δυνατότητα επιτρέπει στον προγραμματιστή να ανακαλύψει τις γεωγραφικές συντεταγμένες (γεωγραφικό πλάτος και μήκος) της τρέχουσας τοποθεσίας του επισκέπτη του ιστότοπου. Είναι ιδιαίτερα χρήσιμο για την παροχή της καλύτερης εμπειρίας περιήγησης στον επισκέπτη, καθώς,

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

για παράδειγμα, αυτό το εργαλείο μπορεί να εμφανίζει αποτελέσματα αναζήτησης που βρίσκονται κοντά στην τοποθεσία του χρήστη.

Η λήψη των πληροφοριών της τοποθεσίας του επισκέπτη του ιστότοπου χρησιμοποιώντας το API γεωγραφικής τοποθεσίας HTML5 δεν είναι δύσκολη. Εκμεταλλεύεται τις τρεις μεθόδους εντός του αντικειμένου navigator.geolocation — getCurrentPosition (), watchPosition () και clearWatch().

Αφού ο χρήστης επιτρέψει στο πρόγραμμα περιήγησης να ενημερώσει τον διακομιστή ιστού σχετικά με την τοποθεσία του (τα προγράμματα περιήγησης δεν θα κοινοποιήσουν την τοποθεσία επισκέπτη με μια ιστοσελίδα εκτός εάν ο χρήστης συμφωνήσει), η διαδικασία γεωεντοπισμού θα πρέπει να πραγματοποιηθεί όπως *φαίνεται στην παρακάτω εικόνα.*

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Get Visitor's Location Using HTML5 Geolocation</title>
<script>
    function showPosition() {
        if(navigator.geolocation) {
            navigator.geolocation.getCurrentPosition(function(position) {
                var positionInfo = "Your current position is (" + "Latitude: " +
position.coords.latitude + ", " + "Longitude: " + position.coords.longitude + ")";
                document.getElementById("result").innerHTML = positionInfo;
            });
        } else {
            alert("Sorry, your browser does not support HTML5 geolocation.");
        }
    }
</script>
</head>
<body>
    <div id="result">
        <!--Position information will be inserted here-->
    </div>
    <button type="button" onclick="showPosition();">Show Position</button>
</body>
</html>
```

Your current position is (Latitude: 41.0079509, Longitude: -8.6270736)
Show Position

Εικόνα 107 *– Η διαδικασία της λειτουργίας Γεωτοποθεσίας (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-geolocation.php)*

Σε περίπτωση που ένας χρήστης δεν θέλει να μοιραστεί τα δεδομένα της τοποθεσίας του με τον ιστότοπο, ο προγραμματιστής μπορεί να παρέχει δύο δευτερεύουσες λειτουργίες όταν θέτει τη λειτουργία getCurrentLocation(). Η πρώτη λειτουργία χρησιμοποιείται σε περίπτωση που η προσπάθεια γεωεντοπισμού είναι επιτυχής, ενώ η δεύτερη χρησιμοποιείται εάν η προσπάθεια γεωεντοπισμού αποτύχει.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Handling the Geolocation Errors and Rejections</title>
<script>
    // Set up global variable
    var result;

    function showPosition() {
        // Store the element where the page displays the result
        result = document.getElementById("result");

        // If geolocation is available, try to get the visitor's position
        if(navigator.geolocation) {
            navigator.geolocation.getCurrentPosition(successCallback, errorCallback);
            result.innerHTML = "Getting the position information...";
        } else {
            alert("Sorry, your browser does not support HTML5 geolocation.");
        }
    };

    // Define callback function for successful attempt
    function successCallback(position) {
        result.innerHTML = "Your current position is (" + "Latitude: " + position.coords.latitude + ", " + "Longitude: " + position.coords.longitude + ")";
    }

    // Define callback function for failed attempt
    function errorCallback(error) {
        if(error.code == 1) {
            result.innerHTML = "You've decided not to share your position, but it's OK. We won't ask you again.";
        } else if(error.code == 2) {
            result.innerHTML = "The network is down or the positioning service can't be reached.";
        } else if(error.code == 3) {
            result.innerHTML = "The attempt timed out before it could get the location data.";
        } else {
            result.innerHTML = "Geolocation failed due to unknown error.";
        }
    }
</script>
</head>
<body>
    <div id="result">
        <!--Position information will be inserted here-->
    </div>
    <button type="button" onclick="showPosition();">Show Position</button>
</body>
</html>
```

Εικόνα 108 *– Εφαρμογή δύο δευτερευουσών λειτουργιών στη λειτουργία getCurrentLocation() (Πηγή: https://www.tutorialrepublic.com/html-tutorial/html5-geolocation.php)*

Υπάρχουν πολλές ενδιαφέρουσες λειτουργίες που μπορούν να εξρευνηθούν με τα δεδομένα γεωεντοπισμού, όπως η εμφάνιση της τοποθεσίας του χρήστη στους Χάρτες Google. Με βάση τα δεδομένα γεωγραφικού πλάτους και μήκους που ανακτώνται μέσω της δυνατότητας γεωεντοπισμού στην HTML5, αυτό το παράδειγμα δείχνει την τρέχουσα τοποθεσία του χρήστη. Αυτό δείχνει απλώς μια στατική εικόνα που δείχνει την τοποθεσία του χρήστη, αν και ένας διαδραστικός χάρτης Google με δυνατότητα μεταφοράς, μεγέθυνσης/σμίκρυνσης και άλλων λειτουργιών, όπως δείχνει αυτό το παράδειγμα.

Όλα τα προαναφερθέντα παραδείγματα έχουν βασιστεί στη λειτουργία getCurrentPosition(). Ωστόσο, η συνάρτηση γεωεντοπισμού έχει μια άλλη τεχνική, την watchPosition () που επιτρέπει την παρακολούθηση της κίνησης του επισκέπτη εμφανίζοντας την ενημερωμένη θέση καθώς αλλάζει η τοποθεσία. Η watchPosition () έχει τις ίδιες παραμέτρους εισόδου με την getCurrentPosition(). Ωστόσο, η

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

watchPosition() μπορεί να ενεργοποιήσει την επιτυχή λειτουργία πολλές φορές, όταν λαμβάνει την τοποθεσία για πρώτη φορά και ξανά, κάθε φορά που εντοπίζει μια νέα θέση, όπως φαίνεται σε αυτό το παράδειγμα.

Σύρσιμο και απόθεση στην HTML5 (Drag and Drop)

Το σύρσιμο και η απόθεση ενός στοιχείου σε μια άλλη τοποθεσία σε έναν ιστότοπο αποτελεί μια κοινή διαδικασία στην καθημερινή ρουτίνα του Διαδικτύου. Η δυνατότητα για σύρσιμο και απόθεση στην HTML5 επιτρέπει αυτή τη διαδικασία, σύροντας και αποθέτοντας οποιοδήποτε στοιχείο. Αυτό το απλό παράδειγμα μεταφοράς και απόθεσης δείχνει στους εκπαιδευόμενους τον τρόπο με τον οποίο μπορούν να προσπαθήσουν να εξοικειωθούν περισσότερο με αυτήν την έννοια. Παρόλο που ο κώδικας φαίνεται να είναι δυσνόητος, είναι αρκετά απλός και λογικός:

- Πρώτον, για να έχει ένα στοιχείο τη δυνατότητα μεταφοράς, το χαρακτηριστικό draggable πρέπει να είναι true:

```
<img draggable= "true">
```

- Στη συνέχεια, θα πρέπει να καθοριστεί τι θα συμβεί μετά τη μεταφορά του στοιχείου. Στο παράδειγμα που δόθηκε παραπάνω, το χαρακτηριστικό ondragstart θέτει μια λειτουργία (drag (event)) που καθορίζει ποια δεδομένα θα συρθούν. Η διαδικασία dataTransfer.setData() ορίζει τον τύπο δεδομένων και την τιμή των δεδομένων που έχουν συρθεί:

```
function drag(ev)                                    {
ev.dataTransfer.setData("text",ev.target.id);
}
```

Σε αυτό το παράδειγμα, ο τύπος δεδομένων είναι το «text», που είναι η τιμή του αναγνωριστικού του στοιχείου με δυνατότητα μεταφοράς («drag1»).

- Το συμβάν ondragover ορίζει πού μπορούν να τοποθετηθούν τα δεδομένα που έχουν συρθεί. Ωστόσο, δεν είναι δυνατή η απόθεση δεδομένων/στοιχείων σε άλλα στοιχεία, από προεπιλογή. Για να επιτραπεί μια απόθεση, ο σχεδιαστής

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ιστοτόπου θα πρέπει να αποτρέψει τον προεπιλεγμένο χειρισμό του στοιχείου, με τη λειτουργια event.preventDefault() για το συμβάν ondragover:

event.preventDefault ()

- Με την απόθεση των δεδομένων που σύρθηκαν, συμβαίνει ένα συμβάν απόθεσης. Στο προηγούμενο παράδειγμα, το χαρακτηριστικό ondrop χρησιμοποίησε μια λειτουργία drop(event):

```
function drop(ev) {
 ev.preventDefault();
 var data=ev.dataTransfer.getData("text");
 ev.target.appendChild(document.getElementById(data));
}
```

- ✓ Η λειτουργία preventDefault() αποτρέπει τον προεπιλεγμένο χειρισμό των δεδομένων από το πρόγραμμα περιήγησης (η προεπιλογή είναι ανοιχτή ως σύνδεσμος κατά την απόθεση).
- ✓ Ο προγραμματιστής λαμβάνει τα δεδομένα που έχουν συρθεί με την τεχνική dataTransfer.getData(). Αυτή η τεχνική θα εμφανίσει τυχόν δεδομένα που ορίστηκαν στον ίδιο τύπο στην τεχνική setData().
- ✓ Τα δεδομένα που σύρθηκαν είναι το αναγνωριστικό του στοιχείου που σύρθηκε («drag1»)
- ✓ Ο προγραμματιστής θα πρέπει επίσης να προσαρτήσει το στοιχείο που έχει συρθεί στο στοιχείο απόθεσης.

2.4 Αναφορές της HTML5

Για μια λεπτομερή, ολοκληρωμένη λίστα στοιχείων σχετικά με τις **Ετικέτες/Στοιχεία της HTML5, τα παγκόσμια χαρακτηριστικά της HTML5, τα Χαρακτηριστικά συμβάντων της HTML5, τους Κώδικες Γλωσσών HTML5, τις Οντότητες χαρακτήρων της HTML5**, τους **καταστατικούς κώδικες HTTP, τον Επιλογέα χρωμάτων στην HTML5** και άλλες χρήσιμες αναφορές, μπορείτε να ανατρέξετε σε αυτόν τον σύνδεσμο (Ενότητα Αναφορών της HTML5).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

3. CSS – Cascading Style Sheets

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πληροφορίες αναφορικά με την ενότητα

Ενότητα:

3. CSS

Προϋποθέσεις:

Βασικές ψηφιακές δεξιότητες, εγκατεστημένο λογισμικό, βασικές γνώσεις εργασίας με αρχεία και βασικές γνώσεις στην HTML (διαβάστε την εισαγωγή μας στην HTML.)

Φόρτος εργασίας:

10 ώρες

Περιγραφή:

Η Cascading Style Sheets γνωστή με τη συντομογραφία CSS (Διαδοχικά Φύλλα Ύφους ή Επικαλυπτόμενα Φύλλα Στυλ) χρησιμοποιείται για να παρουσιάσει το περιεχόμενο ενός εγγράφου που είναι γραμμένο στις γλώσσες σήμανσης HTML ή XML (συμπεριλαμβανομένων των εκδόσεων της XML όπως SVG, MathML ή XHTML). Η CSS περιγράφει τον τρόπο με τον οποίο τα δομικά στοιχεία μιας ιστοσελίδας θα πρέπει να αποδίδονται στην οθόνη, στο χαρτί, σε μια ομιλία ή σε άλλα μέσα.

Μαθησιακά αποτελέσματα:

Οι εκπαιδευόμενοι θα μάθουν πώς να ορίζουν την CSS και πώς να χρησιμοποιούν τη βασική της σύνταξη για να κατασκευάζουν ιστοσελίδες. Θα χρησιμοποιήσουν επίσης της CSS για την μορφοποίηση των στυλ κειμένου, γραμματοσειράς και ιδιοτήτων καθώς και του φόντου μιας σελίδας. Τέλος, θα μορφοποιήσουν λίστες της CSS με επιλογείς στοιχείων όπως κλάσεις και αναγνωριστικά, θα εφαρμόσουν ιδιότητες της CSS όπως όρια, ύψος και πλάτος, θα χρησιμοποιήσουν ψευδοστοιχεία της CSS και θα μάθουν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

πώς να τοποθετούν τα στοιχεία μέσω της ιδιότητας τοποθέτησης στοιχείων της CSS (Position), καθώς και πώς να επικυρώνουν την CSS και την HTML.

Απαιτούμενα υλικά:

- Υπολογιστής ή φορητός υπολογιστής
- Σύνδεση στο Διαδίκτυο
- Επεξεργαστής κειμένου (online ή offline): Sublime Text/Brackets/W3Schools online editor

Σενάριο μαθήματος:

Ο συνολικός χρόνος για την ενότητα αυτή είναι 10 ώρες και εναπόκειται στον εκπαιδευτικό να αποφασίσει πόσο χρόνο θα αφιερώσει για την διδασκαλία του κάθε υποθέματος. Προκειμένου να αξιοποιήσετε στο έπακρο όλο τον διαθέσιμο χρόνο, προτείνουμε τη χρήση του εκπαιδευτικού υλικού που συντάχθηκε στο πλαίσιο του έργου (παρουσιάσεις PPT), το οποίο σχεδιάστηκε με γνώμονα την αποτελεσματική χρήση του χρόνου. Αυτές οι παρουσιάσεις αποτελούνται από τα ακόλουθα στοιχεία:

- Ανάπτυξη των υποθεμάτων και των βασικών ιδεών που πρέπει να διατηρηθούν.
- Προτεινόμενες Δραστηριότητες/Ασκήσεις.

Επομένως, εάν ο εκπαιδευτής ακολουθήσει τη λογική σειρά των PPTs, σίγουρα θα μπορέσει να ολοκληρώσει την ενότητα εντός του καθορισμένου χρονικού ορίου. Αυτές οι παρουσιάσεις μπορούν επίσης να διατεθούν στους μαθητές για ατομική μελέτη.

Υποθέματα:

3.1. Εισαγωγή στην CSS
3.2. Το Responsive Web Design της CSS
3.3. Πλέγμα CSS
3.4. Προηγμένη CSS

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Επιπλέον πόροι:

- Μάθηματα στην CSS: w3schools
- Διαδικτυακά Μαθήματα στην HTML και την CSS: CodeAcademy
- Sublime Text - πρόγραμμα επεξεργασίας πηγαίου κώδικα σε πολλαπλές πλατφόρμες κοινής χρήσης που υποστηρίζει πολλές γλώσσες προγραμματισμού και σήμανσης

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

3.1 Σύντομη επισκόπηση στην CSS

Τι είναι η CSS;

Η Cascading Style Sheets (Διαδοχικά Φύλλα Ύφους ή Επικαλυπτόμενα Φύλλα Στυλ), γνωστή με το ακρώνυμο CSS, είναι μια απλή γλώσσα σχεδιασμού ηλεκτρονικών υπολογιστών που βοηθά στην απλοποίηση της διαδικασίας που καθιστά τις ιστοσελίδες ευπαρουσίαστες.

Η CSS διαχειρίζεται την εμφάνιση και την γενική εντύπωση που έχει μια ιστοσελίδα. Με τη χρήση της CSS, μπορείτε να ελέγξετε το χρώμα του κειμένου, το στυλ των γραμματοσειρών, το διάστημα μεταξύ των παραγράφων, το μέγεθος και τη διάταξη που έχουν οι στήλες, τις εικόνες φόντου ή τα χρώματα γενικά, καθώς και διάφορα άλλα εφέ.

Η CSS είναι συνήθως εύληπτη και κατανοητή στους χρήστες, ενώ τους δίνει παράλληλα τη δυνατότητα να έχουν ουσιαστικό έλεγχο επί της εμφάνισης ενός εγγράφου HTML. Συνήθως, η CSS συνδυάζεται με τις γλώσσες σήμανσης HTML ή XHTML.

Προαπαιτούμενες γνώσεις

Οι εκπαιδευόμενοι θα πρέπει:

- Να έχουν βασικές γνώσεις στην επεξεργασία κειμένου με τη βοήθεια οποιουδήποτε επεξεργαστή κειμένου.
- Να γνωρίζουν πως να δημιουργούν καταλόγους και αρχεία.
- Να γνωρίζουν πώς να περιηγούνται σε διαφορετικούς τύπους ευρετηρίων.
- Να γνωρίζουν πώς να περιηγούνται στο διαδίκτυο χρησιμοποιώντας γνωστά προγράμματα περιήγησης όπως το Internet Explorer ή το Firefox.
- Να είναι εξοικειωμένοι με τον σχεδιασμό απλών ιστοσελίδων μέσω της χρήσης HTML ή XHTML.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πλεονεκτήματα της CSS

- **Η CSS εξοικονομεί χρόνο** - Μπορείτε να γράψετε σε γλώσσα CSS μία φορά και στη συνέχεια να επαναχρησιμοποιήσετε το ίδιο φύλλο σε πολλές σελίδες HTML. Μπορείτε να ορίσετε ένα στυλ για κάθε στοιχείο HTML και να το εφαρμόσετε σε όσες ιστοσελίδες θέλετε.
- **Οι σελίδες φορτώνονται γρηγορότερα** - Εάν χρησιμοποιείτε CSS, δεν χρειάζεται να γράφετε κάθε φορά τις ιδιότητες ετικέτας HTML. Απλά ορίστε έναν κανόνα CSS σε μια ετικέτα και εφαρμόστε τον σε όλες τις εμφανίσεις αυτής της ετικέτας. Επομένως, όσο λιγότερη είναι η κωδικοποίηση τόσο γρηγορότερη είναι η λήψη.
- **Εύκολη συντήρηση** - Για να κάνετε μια αλλαγή για ολόκληρη τη σελίδα, απλά αλλάξτε το στυλ και όλα τα στοιχεία σε όλες τις ιστοσελίδες και θα ενημερωθούν αυτόματα.
- **Ανώτερα στυλ από την HTML** - η CSS παρέχει ένα ευρύτερο φάσμα ιδιοτήτων από την HTML, που σας επιτρέπει να βελτιστοποιήσετε την εμφάνιση και μιας ιστοσελίδας HTML.
- **Συμβατότητα πολλαπλών συσκευών** - Τα φύλλα ύφους επιτρέπουν τη βελτιστοποίηση του περιεχομένου για περισσότερους από έναν τύπους συσκευών. Μέσω της χρήσης του ίδιου εγγράφου HTML, δίνεται η δυνατότητα παρουσίασης διαφορετικών εκδοχών ενός ιστότοπου για φορητές συσκευές όπως είναι οι PDA (Προσωπικοί Ψηφιακοί Οδηγοί) και τα κινητά τηλέφωνα ή για σκοπούς εκτύπωσης.
- **Παγκόσμια πρότυπα ιστού** – Τώρα τα χαρακτηριστικά της HTML καταργούνται και συνιστάται η χρήση της CSS. Έτσι λοιπόν, θα ήταν καλύτερα να αρχίσετε να χρησιμοποιείτε την CSS σε όλες τις σελίδες HTML, για να τις καταστήσετε συμβατές με μελλοντικά προγράμματα περιήγησης.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ποιος Δημιουργεί και Διατηρεί την CSS;

Η CSS δημιουργείται και συντηρείται μέσω μιας ομάδας ατόμων εντός του W3C, που ονομάζεται Ομάδα Εργασίας της CSS. Η ομάδα εργασίας της CSS δημιουργεί έγγραφα που ονομάζονται τεχνικά χαρακτηριστικά. Όταν ένα τεχνικό χαρακτηριστικό/τεχνική προδιαγραφή έχει συζητηθεί και επικυρωθεί επίσημα από τα μέλη της W3C, γίνεται σύσταση.
Τα επικυρωμένα τεχνικά χαρακτηριστικά ονομάζονται συστάσεις επειδή το W3C δεν έχει κανέναν ουσιαστικό έλεγχο επί της εφαρμογής της γλώσσας. Ανεξάρτητες εταιρείες και οργανισμοί είναι υπεύθυνοι για τη δημιουργία του λογισμικού.

ΣΗΜΕΙΩΣΗ: Η Κοινοπραξία Παγκόσμιου Ιστού (World Wide Web Consortium ή W3C) είναι μια ομάδα που κάνει συστάσεις σχετικά με το πώς λειτουργεί το Διαδίκτυο και πώς θα πρέπει να εξελιχθεί.

Εκδόσεις της CSS

Η CSS κυκλοφόρησε αρχικά το 1996 και παρείχε τις δυνατότητες για τη διαμόρφωση χαρακτηριστικών κειμένου όπως η στοίχιση, συμπεριλαμβανομένων των επιλογών γραμματοσειράς και το χρώματος έμφασης του κειμένου, το φόντο και άλλα χαρακτηριστικά. Το CSS2 κυκλοφόρησε το 1998 με επιπρόσθετες επιλογές στυλ για άλλους τύπους μέσων δικτύωσης, έτσι ώστε να μπορεί να χρησιμοποιηθεί για το σχεδιασμό της διάταξης μιας ιστοσελίδας. Το CSS3 κυκλοφόρησε το 1999 και σ' αυτό προστέθηκαν ιδιότητες τύπου παρουσίασης που επιτρέπουν τη δημιουργία μιας παρουσίασης από έγγραφα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σύνταξη των στυλ CSS

Η CSS αποτελείται από ένα σύνολο στυλιστικών κανόνων τα οποία πρέπει να ερμηνεύσει ένα πρόγραμμα περιήγησης για να τα εφαρμόσει στα αντίστοιχα στοιχεία μιας ιστοσελίδας. Ένας κανόνας αποτελείται από τρία μέρη:

- **Επιλογέας**: Ο επιλογέας είναι μια ετικέτα HTML στην οποία θέλουμε να εφαρμοστεί ένα στυλ. Αυτό θα μπορούσε να είναι οποιαδήποτε ετικέτα όπως <h1> ή <table> κτλ.
- **Ιδιότητα**: Οι ιδιότητες προσθέτουν διάφορα χαρακτηριστικά στις ετικέτες της HTML. Με απλά λόγια, όλα τα χαρακτηριστικά της HTML μετατρέπονται σε ιδιότητες CSS, δηλαδή χρώμα, περίγραμμα κ.λπ.
- **Τιμή**: Οι τιμές εκχωρούνται στις ιδιότητες. Για παράδειγμα, η ιδιότητα χρώματος μπορεί να έχει την τιμή του κόκκινου χρώματος είτε # F1F1F1 κλπ.

Επιλογείς CSS

Οι επιλογείς CSS χρησιμοποιούνται για την «εύρεση» (ή επιλογή) των στοιχείων HTML που θέλετε να διαμορφώσετε.
Μπορούμε να διαιρέσουμε τους επιλογείς CSS σε πέντε κατηγορίες:

- **Απλοί επιλογείς** (επιλογή στοιχείων με βάση το όνομα, το αναγνωριστικό, την κλάση)
- **Επιλογείς συνδυασμού** (επιλογή στοιχείων με βάση μια συγκεκριμένη σχέση μεταξύ τους)
- **Επιλογείς ψευδο-κλάσης** (επιλογή στοιχείων με βάση μια συγκεκριμένη κατάσταση)
- **Επιλογείς ψευδο-στοιχείων** (επιλογή και στυλιστική ανάπτυξη ενός μέρους ενός στοιχείου)
- **Επιλογείς ιδιοτήτων** (επιλογή στοιχείων με βάση μια ιδιότητα ή τιμή ιδιότητας)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο επιλογέας στοιχείου CSS

Ο επιλογέας στοιχείων επιλέγει ουσιαστικά στοιχεία HTML με βάση το όνομα του στοιχείου.

```
p {
  text-align: center;
  color: red;
}
```

Εικόνα 1 - *Ο επιλογέας στοιχείου CSS (Πηγή: https://www.w3schools.com/css/css_selectors.asp)*

Ο επιλογέας αναγνωριστικού CSS (ID)

Ο επιλογέας αναγνωριστικού χρησιμοποιεί ένα χαρακτηριστικό αναγνώρισης ενός στοιχείου HTML για να επιλέξει ένα συγκεκριμένο στοιχείο.
Το αναγνωριστικό ενός στοιχείου είναι μοναδικό σε μια σελίδα, οπότε ο επιλογέας αναγνωριστικού χρησιμοποιείται για την επιλογή ενός μοναδικού στοιχείου!
Για να επιλέξετε ένα στοιχείο με ένα συγκεκριμένο αναγνωριστικό, γράψτε έναν χαρακτήρα hash (#), ακολουθούμενο από το αναγνωριστικό του στοιχείου.

```
#para1 {
  text-align: center;
  color: red;
}
```

Εικόνα 2 *- Ο επιλογέας αναγνωριστικού CSS (ID) (Πηγή: https://www.w3schools.com/css/css_selectors.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο επιλογέας κλάσης CSS

Ο επιλογέας κλάσης επιλέγει στοιχεία HTML με μια συγκεκριμένη ιδιότητα κλάσης. Για να επιλέξετε στοιχεία με μια συγκεκριμένη κλάση, γράψτε έναν περιοδικό χαρακτήρα (.), ακολουθούμενο από το όνομα της κλάσης.

```
.center {
  text-align: center;
  color: red;
}
```

Εικόνα 3 *- Ο επιλογέας κλάσης CSS (Πηγή: https://www.w3schools.com/css/css_selectors.asp)*

Ο καθολικός επιλογέας CSS

Ο καθολικός επιλογέας (*) επιλέγει όλα τα στοιχεία HTML στη σελίδα.

```
* {
  text-align: center;
  color: blue;
}
```

Εικόνα 4 *- Ο καθολικός επιλογέας CSS (Πηγή: https://www.w3schools.com/css/css_selectors.asp)*

Ομαδοποίηση Επιλογέων

Μπορείτε να εφαρμόσετε στυλ σε πολλούς επιλογείς αν θέλετε. Απλά διαχωρίστε τους επιλογείς με ένα κόμμα, όπως δίνεται στο ακόλουθο παράδειγμα:

```
h1, h2, p {
  text-align: center;
  color: red;
}
```

Εικόνα 5 *- Ομαδοποίηση Επιλογέων (Πηγή: https://www.w3schools.com/css/css_selectors.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σχόλια CSS

- Τα σχόλια χρησιμοποιούνται για να επεξηγούν τον κώδικα και μπορεί να σας βοηθήσουν όταν επεξεργαζεστε τον πηγαίο κώδικα σε μεταγενέστερη ημερομηνία.
- Τα σχόλια αγνοούνται από τα προγράμματα περιήγησης.

Ένα σχόλιο CSS ξεκινά με /* και τελειώνει με */

```
/* This is a single-line comment */
p {
  color: red;
}
```

Εικόνα 6 *- Σχόλια CSS (Πηγή: https://www.w3schools.com/css/css_comments.asp)*

Χρώματα CSS

Τα χρώματα στην CSS μπορούν να καθοριστούν με τους ακόλουθους τύπους σεμειογραφίας:

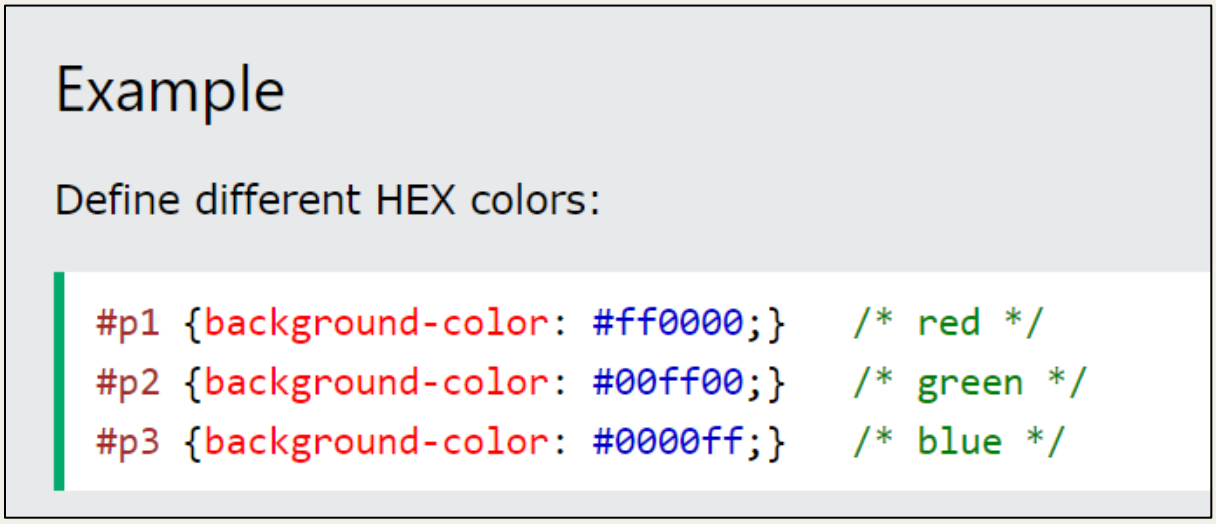
- Δεκαεξαδικά χρώματα
- Δεκαεξαδικά χρώματα με διαφάνεια
- Χρώματα RGB
- Χρώματα RGBA
- Χρώματα HSL
- Χρώματα HSLA
- Καθιερωμένα ονόματα χρωμάτων φυλλομετρητή/Αποκλειστικά ανά φυλλομετρητή ονόματα χρωμάτων
- # - # - # - # - # gnome-commander.master.el.po (el)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Δεκαεξαδικά χρώματα

Ένα δεκαεξαδικό χρώμα ορίζεται με: #RRGGBB, όπου οι δεκαεξαδικοί ακέραιοι αριθμοί RR (κόκκινο), GG (πράσινο) και BB (μπλε) καθορίζουν τα συστατικά στοιχεία του χρώματος. Όλες οι τιμές πρέπει να είναι μεταξύ 00 και FF.

Για παράδειγμα, η τιμή # 0000ff αποδίδεται ως μπλε, επειδή το μπλε συστατικό στοιχείο έχει οριστεί στην υψηλότερη τιμή του (ff) και τα άλλα έχουν οριστεί στο 00.

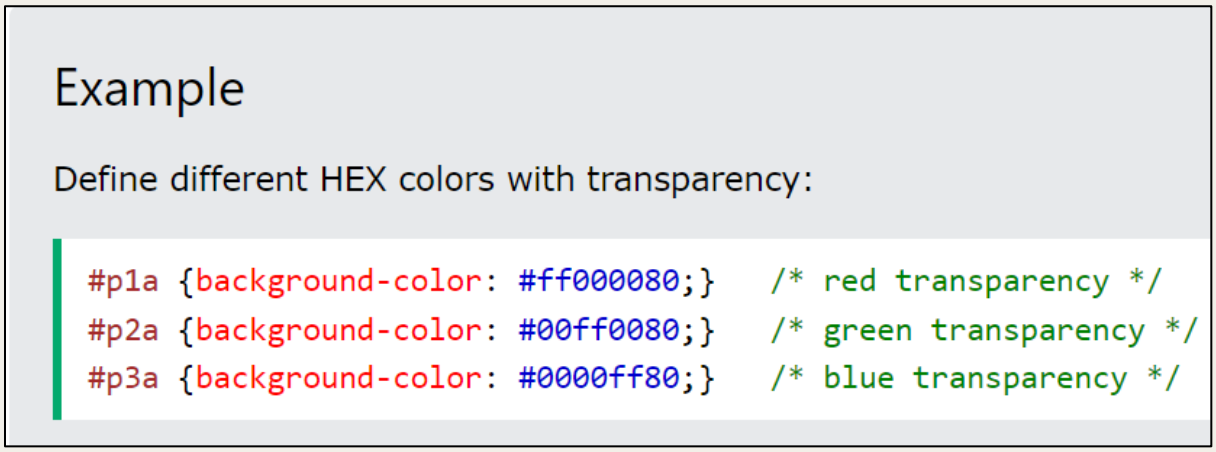


Εικόνα 7 *- Δεκαεξαδικά Χρώματα (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Δεκαεξαδικά χρώματα με διαφάνεια

Ένα δεκαεξαδικό χρώμα ορίζεται με: #RRGGBB. Για να προσθέσετε διαφάνεια, προσθέστε δύο επιπλέον ψηφία μεταξύ 00 και FF.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 8 *- Δεκαεξαδικά χρώματα με διαφάνεια (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Χρώματα RGB

Μια τιμή χρώματος RGB καθορίζεται με τη συνάρτηση rgb(), η οποία έχει την ακόλουθη σύνταξη:

rgb(κόκκινο, πράσινο, μπλε)

Κάθε παράμετρος (κόκκινη, πράσινη και μπλε) ορίζει την ένταση του χρώματος και μπορεί να είναι ένας ακέραιος αριθμός μεταξύ 0 και 255 ή μια ποσοστιαία τιμή (από 0% έως 100%).

Για παράδειγμα, η τιμή rgb(0,0,255) αποδίδεται ως μπλε, επειδή η μπλε παράμετρος ρυθμίζεται στην υψηλότερη τιμή της (255) και οι άλλες ρυθμίζονται στο 0.

Επίσης, οι ακόλουθες τιμές ορίζουν ισόποσα χρώματα: rgb(0,0,255) και rgb(0%,0%,100%).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Define different RGB colors:

```
#p1 {background-color: rgb(255, 0, 0);}   /* red */
#p2 {background-color: rgb(0, 255, 0);}   /* green */
#p3 {background-color: rgb(0, 0, 255);}   /* blue */
```

Εικόνα 9 *- Χρώματα RGB (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Χρώματα RGBA

Οι τιμές χρώματος RGBA είναι μια επέκταση των τιμών χρώματος RGB με ένα κανάλι άλφα - το οποίο καθορίζει την αδιαφάνεια του αντικειμένου.

Μια τιμή χρώματος RGB καθορίζεται με τη συνάρτηση rgb(), η οποία έχει την ακόλουθη σύνταξη:

rgba(κόκκινο, πράσινο, μπλε, άλφα)

Η παράμετρος άλφα είναι ένας αριθμός μεταξύ 0,0 (πλήρως διαφανής) και 1,0 (πλήρως αδιαφανής).

Example

Define different RGB colors with opacity:

```
#p1 {background-color: rgba(255, 0, 0, 0.3);}   /* red with opacity */
#p2 {background-color: rgba(0, 255, 0, 0.3);}   /* green with opacity */
#p3 {background-color: rgba(0, 0, 255, 0.3);}   /* blue with opacity */
```

Εικόνα 10 *- RGB Χρώματα με αδιαφάνεια (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Χρώματα HSL

Το HSL αντιπροσωπεύει την απόχρωση, τον κορεσμό και την φωτεινότητα - και αντιπροσωπεύει μια κυλινδρική συντεταγμένη αναπαράσταση χρωμάτων.

Μια τιμή χρώματος RGB καθορίζεται με τη συνάρτηση rgb(), η οποία έχει την ακόλουθη σύνταξη:

hsl(απόχρωση, κορεσμός, φωτεινότητα)

Η απόχρωση είναι ένας βαθμός στον τροχό των χρωμάτων (από 0 έως 360) - το 0 (ή 360) είναι κόκκινο, το 120 είναι πράσινο, το 240 είναι μπλε. Ο κορεσμός είναι μια ποσοστιαία τιμή. Το 0% σημαίνει μια απόχρωση του γκρι και 100% είναι το πλήρες χρώμα. Η φωτεινότητα είναι επίσης μια ποσοστιαία τιμή. Το 0% είναι μαύρο, το 100% είναι λευκό.

Example

Define different HSL colors:

```
#p1 {background-color: hsl(120, 100%, 50%);}   /* green */
#p2 {background-color: hsl(120, 100%, 75%);}   /* light green */
#p3 {background-color: hsl(120, 100%, 25%);}   /* dark green */
#p4 {background-color: hsl(120, 60%, 70%);}    /* pastel green */
```

Εικόνα 11- *Χρώματα HSL (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Χρώματα HSLA

Οι τιμές χρώματος RGBA είναι μια επέκταση των τιμών χρώματος RGB με ένα κανάλι άλφα - το οποίο καθορίζει την αδιαφάνεια του αντικειμένου.

Μια τιμή χρώματος RGB καθορίζεται με τη συνάρτηση rgb(), η οποία έχει την ακόλουθη σύνταξη:

hsla(απόχρωση, κορεσμός, φωτεινότητα, άλφα)

Η παράμετρος άλφα είναι ένας αριθμός μεταξύ 0,0 (εντελώς διαφανής) και 1,0 (εντελώς αδιαφανής).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Define different HSL colors with opacity:

```
#p1 {background-color: hsla(120, 100%, 50%, 0.3);}   /* green with opacity */
#p2 {background-color: hsla(120, 100%, 75%, 0.3);}   /* light green with opacity */
#p3 {background-color: hsla(120, 100%, 25%, 0.3);}   /* dark green with opacity */
#p4 {background-color: hsla(120, 60%, 70%, 0.3);}    /* pastel green with opacity */
```

Εικόνα 12 *- HSL Χρώματα με αδιαφάνεια (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Καθιερωμένα ονόματα χρωμάτων φυλλομετρητή/Αποκλειστικά ονόματα χρωμάτων ανά φυλλομετρητή

Υπάρχουν 140 καθιερωμένα ονόματα χρωμάτων στις προδιαγραφές χρώματος HTML και CSS.

Για παράδειγμα : μπλε, κόκκινο, κοραλί, καφέ, κλπ:

Example

Define different color names:

```
#p1 {background-color: blue;}
#p2 {background-color: red;}
#p3 {background-color: coral;}
#p4 {background-color: brown;}
```

Εικόνα 13 *- Αποκλειστικά ονόματα χρωμάτων ανά φυλλομετρητή (Πηγή: https://www.w3schools.com/css/css_colors.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Το currentcolor Keyword

Το currentcolor Keyword αναφέρεται στην τιμή της ιδιότητας χρώματος ενός στοιχείου.

Example

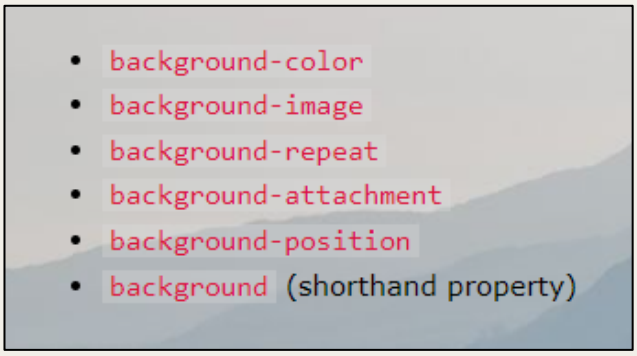
The border color of the following element will be blue, because the text color of the element is blue:

```
#myDIV {
  color: blue; /* Blue text color */
  border: 10px solid currentcolor; /* Blue border color */
}
```

Εικόνα 14 *- Το currentcolor keyword (Πηγή: https://www.w3schools.com/colors/colors_currentcolor.asp)*

Φόντο CSS (background)

- Οι ιδιότητες φόντου CSS χρησιμοποιούνται για την προσθήκη εφέ φόντου σε ένα στοιχείο.
- Ιδιότητες φόντου

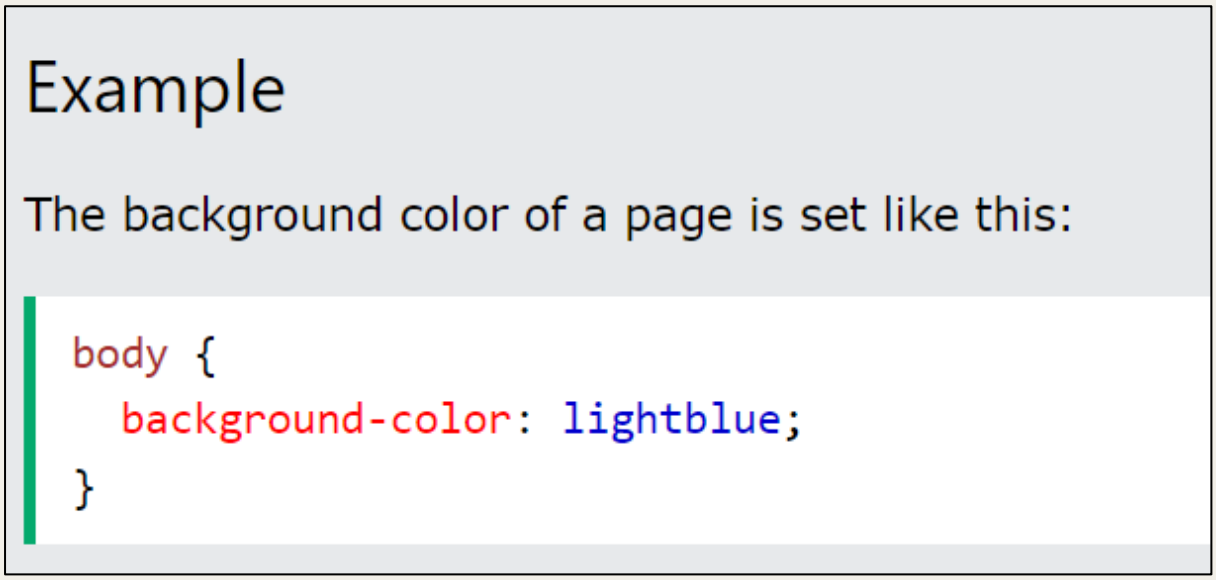


Εικόνα 15 *- Φόντο CSS (background)*

Χρώμα φόντου CSS (background-color)

Η ιδιότητα χρώμα φόντου καθορίζει το χρώμα φόντου ενός στοιχείου.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



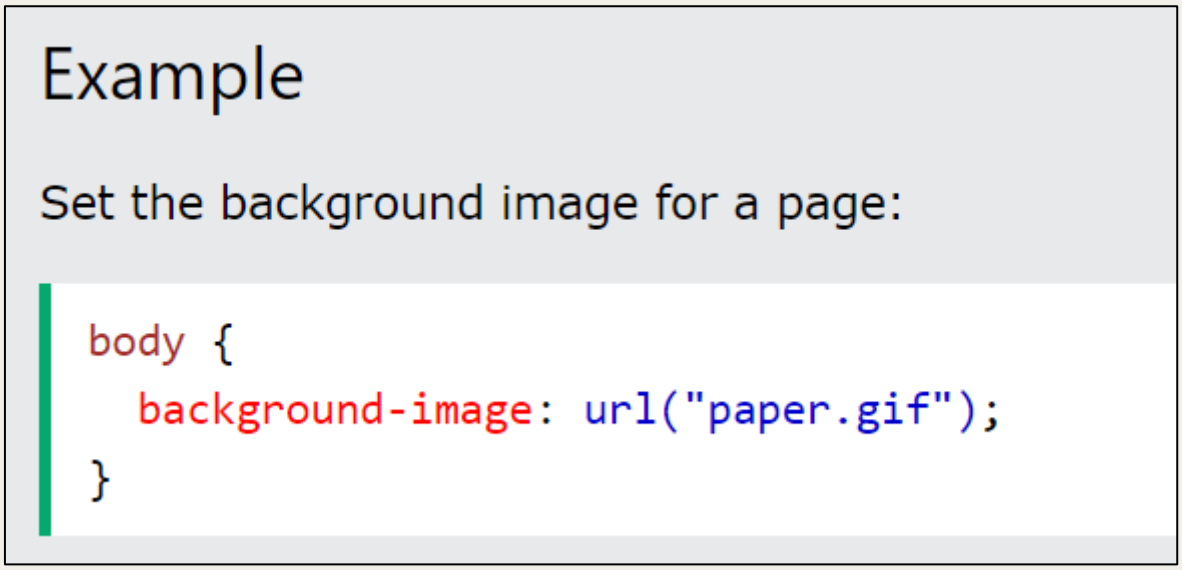
Εικόνα 16 *- Χρώμα φόντου CSS (background-color) (Πηγή: https://www.w3schools.com/css/css_background.asp)*

Εικόνα φόντου CSS

Η ιδιότητα μιας εικόνας φόντου καθορίζει μια εικόνα για χρήση ως φόντο ενός στοιχείου.

Από προεπιλογή, η εικόνα επαναλαμβάνεται έτσι ώστε να καλύπτει ολόκληρο το στοιχείο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 17 - *Εικόνα φόντου CSS (Πηγή: https://www.w3schools.com/css/css_background.asp)*

Επανάληψη φόντου CSS (background-repeat)

Από προεπιλογή, η ιδιότητα φόντου-εικόνας επαναλαμβάνει μια εικόνα τόσο οριζόντια όσο και κάθετα.
Ορισμένες εικόνες θα πρέπει να επαναλαμβάνονται μόνο οριζόντια ή κάθετα, διαφορετικά θα φαίνονται παράξενες.

Εισαγωγή φόντου CSS (background attachment)

Η ιδιότητα εισαγωγής φόντου καθορίζει εάν μια εικόνα φόντου πρέπει να μετακινηθεί ή να διορθωθεί.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Specify that the background image should be fixed:

```
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
  background-attachment: fixed;
}
```

Εικόνα 18 *- Εισαγωγή φόντου CSS (background attachment) (Πηγή: https://www.w3schools.com/css/css_background_attachment.asp)*

Example

Specify that the background image should scroll with the rest of the page:

```
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
  background-attachment: scroll;
}
```

Εικόνα 19 *- Εισαγωγή φόντου CSS (background attachment) (Πηγή: https://www.w3schools.com/css/css_background_attachment.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Φόντο CSS - Σύντομη ιδιότητα/Στενογραφία (shorthand)

Για να συντομεύσετε τον κώδικα, είναι επίσης δυνατό να καθορίσετε όλες τις ιδιότητες φόντου σε μια μόνο ιδιότητα. Αυτό ονομάζεται στενογραφία.

Αντί να γράψετε:

```
body {
  background-color: #ffffff;
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
}
```

Εικόνα 20 *- Φόντο CSS - Σύντομη ιδιότητα/Στενογραφία (Πηγή: https://www.w3schools.com/css/css_background_shorthand.asp)*

Μπορείτε να χρησιμοποιήσετε την ιδιότητα φόντου με στενογραφία:

Example

Use the shorthand property to set the background properties in one declaration:

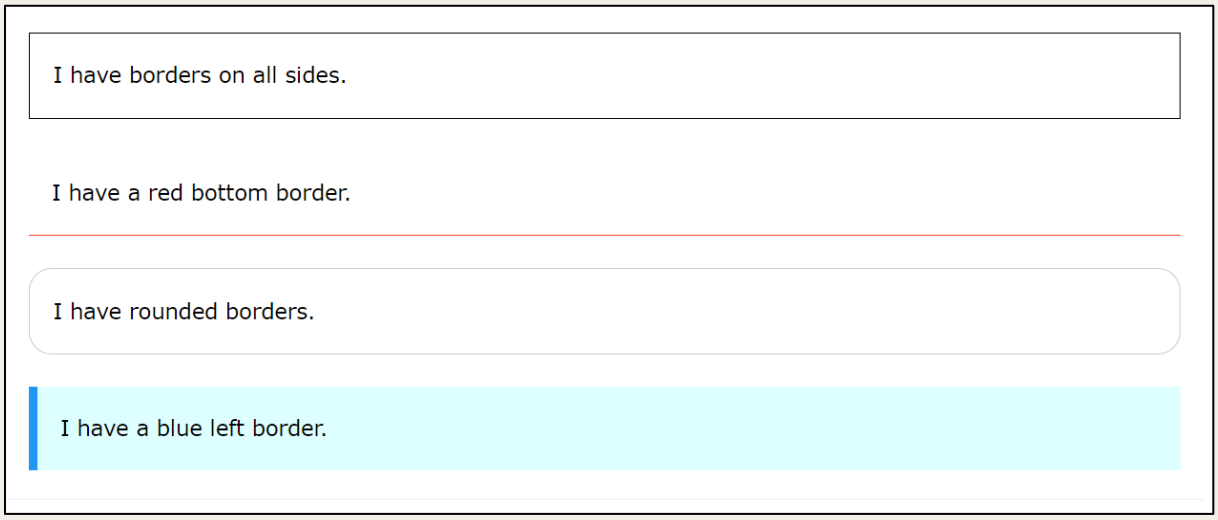
```
body {
  background: #ffffff url("img_tree.png") no-repeat right top;
}
```

Εικόνα 21 *- Φόντο CSS - Σύντομη ιδιότητα/Στενογραφία (Πηγή: https://www.w3schools.com/css/css_background_shorthand.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Όρια CSS (borders)

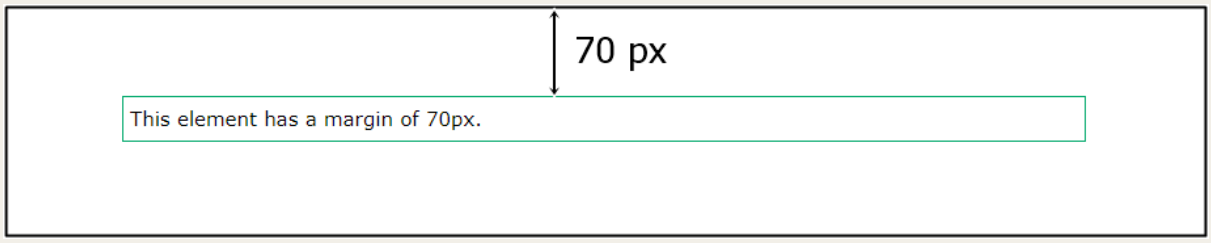
Οι ιδιότητες περιγράμματος/συνόρων CSS σας επιτρέπουν να καθορίσετε το στυλ, το πλάτος και το χρώμα του περιγράμματος ενός στοιχείου.



Εικόνα 22 - *Όρια CSS (borders) (Πηγή: https://www.w3schools.com/css/css_border.asp)*

Περιθώρια CSS (margins)

Τα περιθώρια χρησιμοποιούνται για τη δημιουργία χώρου γύρω από στοιχεία, εκτός των καθορισμένων ορίων.



Εικόνα 23 - *Περιθώρια CSS (margins) (Πηγή: https://www.w3schools.com/css/css_margin.asp)*

Με την CSS, έχεις τον πλήρη έλεγχο των περιθωρίων. Υπάρχουν ιδιότητες για τον καθορισμό του περιθωρίου για κάθε πλευρά ενός στοιχείου (πάνω, δεξιά, κάτω και αριστερά).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

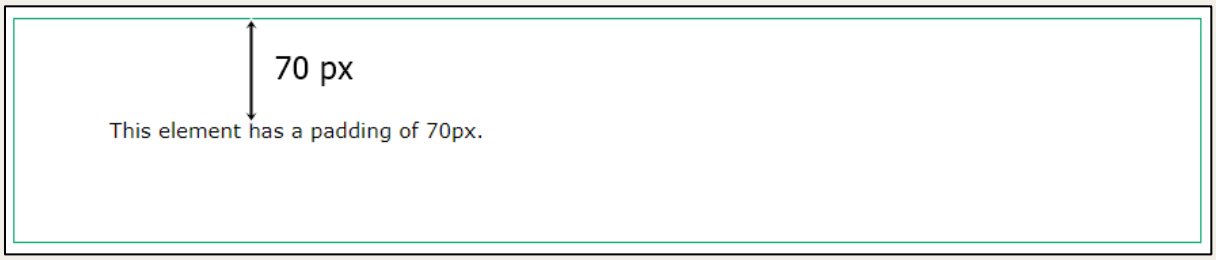
Set different margins for all four sides of a <p> element:

```
p {
  margin-top: 100px;
  margin-bottom: 100px;
  margin-right: 150px;
  margin-left: 80px;
}
```

Εικόνα 24 *- Περιθώρια CSS (margins) (Πηγή: https://www.w3schools.com/css/css_margin.asp)*

Επένδυση CSS (padding)

Η ιδιότητα επένδυσης (padding) χρησιμοποιείται για τη δημιουργία χώρου γύρω από το περιεχόμενο ενός στοιχείου, μέσα σε οποιοδήποτε καθορισμένο περίγραμμα.



Εικόνα 25 *- Επένδυση CSS (padding) (Πηγή: https://www.w3schools.com/css/css_padding.asp)*

Με την CSS, έχετε τον πλήρη έλεγχο μιας επένδυσης. Υπάρχουν ιδιότητες για τον καθορισμό του περιθωρίου για κάθε πλευρά ενός στοιχείου (πάνω, δεξιά, κάτω και αριστερά).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Set different padding for all four sides of a element:

```
div {
  padding-top: 50px;
  padding-right: 30px;
  padding-bottom: 50px;
  padding-left: 80px;
}
```

Εικόνα 26 *- Επένδυση CSS (padding) (Πηγή: https://www.w3schools.com/css/css_padding.asp)*

Ύψος και πλάτος CSS

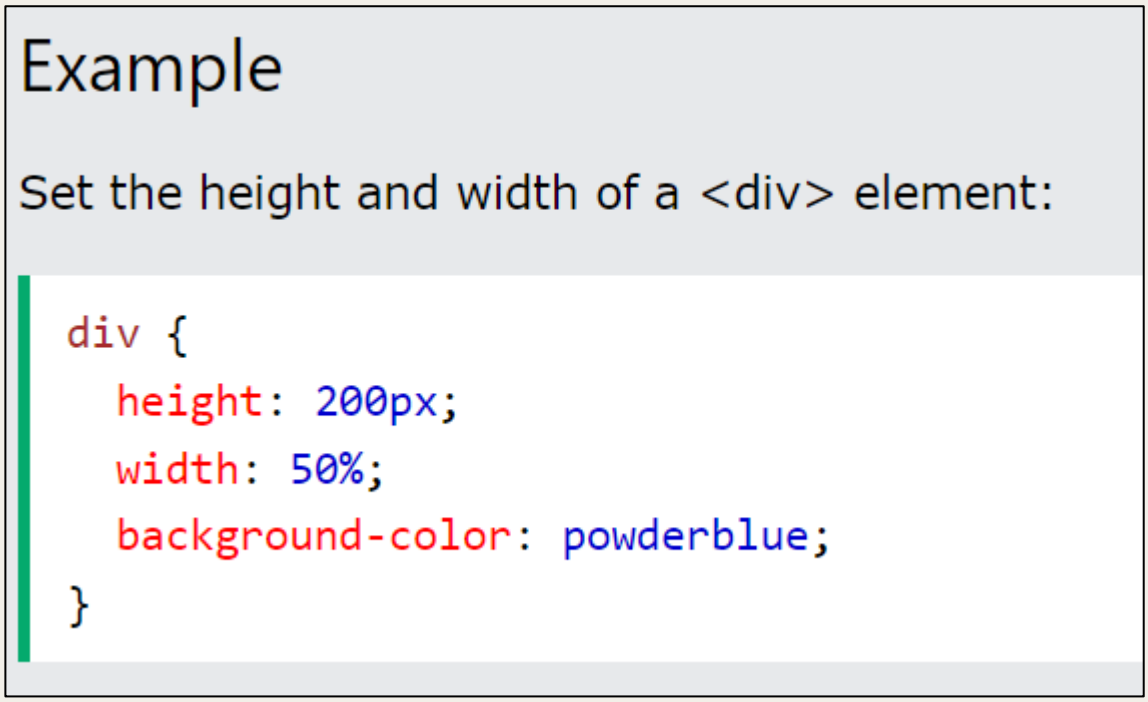
Οι ιδιότητες ύψους και πλάτους χρησιμοποιούνται για τον καθορισμό του ύψους και του πλάτους ενός στοιχείου.

Οι ιδιότητες ύψους και πλάτους δεν περιλαμβάνουν επένδυση, περιγράμματα ή περιθώρια. Ορίζει το ύψος/πλάτος της περιοχής μέσα στην επένδυση (padding), το περίγραμμα και το περιθώριο του στοιχείου.

Οι ιδιότητες ύψους και πλάτους μπορεί να έχουν τις ακόλουθες τιμές:

- auto - Αυτό είναι προεπιλογή. Το πρόγραμμα περιήγησης υπολογίζει το ύψος και το πλάτος
- μήκος - Ορίζει το ύψος/πλάτος σε px, cm κ.λπ.
- % - Ορίζει το ύψος/πλάτος σε ποσοστιαία τιμή του μπλοκ που περιέχει
- αρχικό - Ορίζει το ύψος/πλάτος στην προεπιλεγμένη τιμή
- CSS «Κληρονομικότητα» - Το ύψος/πλάτος θα κληρονομείται από τη γονική του τιμή

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



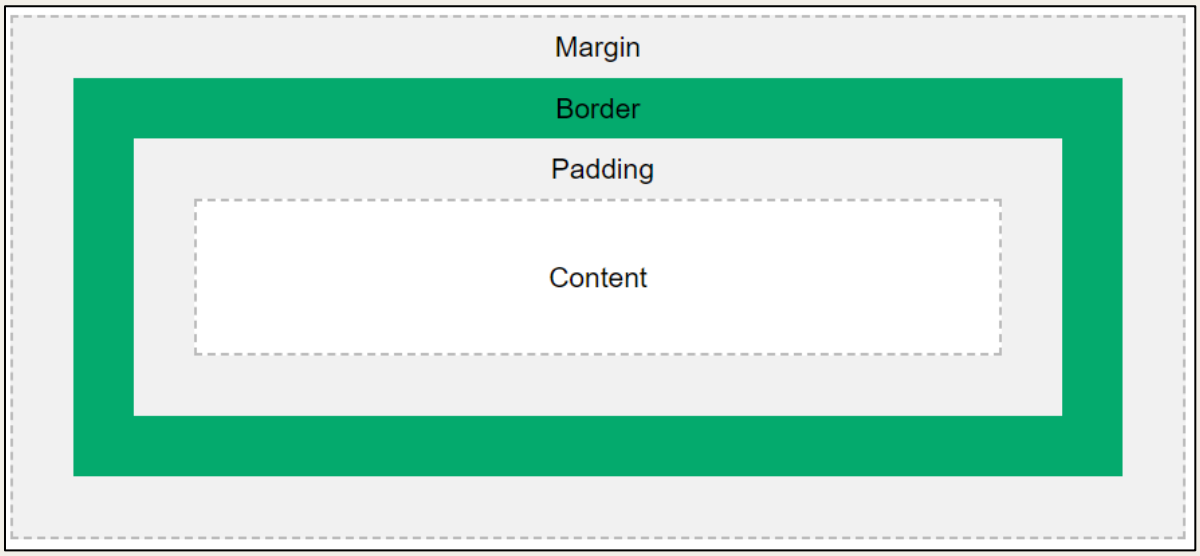
Εικόνα 27 *- Ύψος και πλάτος CSS (Πηγή: https://www.w3schools.com/css/css_dimension.asp)*

Το μοντέλο κουτιού CSS (Box Model)

Στην CSS, ο όρος «μοντέλο κουτιού» (Box Model) χρησιμοποιείται όταν μιλάμε για το σχεδιασμό και τη διάταξη.

Το μοντέλο κουτιού CSS είναι ουσιαστικά ένα πλαίσιο που καλύπτει κάθε στοιχείο HTML. Αποτελείται από: περιθώρια, περιγράμματα, επένδυση, και το περιεχόμενο. Η παρακάτω εικόνα απεικονίζει το μοντέλο πλαισίου:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 28 *-Το μοντέλο κουτιού CSS (Box Model)(Πηγή: https://www.w3schools.com/css/css_boxmodel.asp)*

Επεξήγηση των διάφορων μερών:

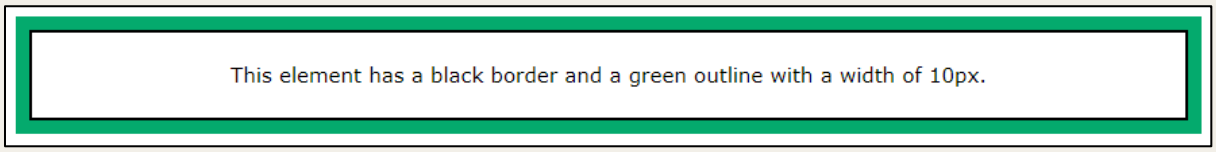
- Περιεχόμενο - Το περιεχόμενο του πλαισίου, όπου εμφανίζονται το κείμενο και οι εικόνες
- Επένδυση - Διασαφηνίζει μια περιοχή γύρω από το περιεχόμενο. Η επένδυση είναι διαφανής
- Περίγραμμα - Ένα περίγραμμα που περιτριγυρίζει την επένδυση και το περιεχόμενο
- Περιθώριο - Διασαφηνίζει μια περιοχή έξω από τα σύνορα. Το περιθώριο είναι διαφανές

Το μοντέλο κουτιού CSS μας επιτρέπει να προσθέσουμε ένα περίγραμμα γύρω από τα στοιχεία και να ορίσουμε το διάστημα μεταξύ των στοιχείων.

Περίγραμμα CSS (outline)

Ένα περίγραμμα είναι μια γραμμή που σχεδιάζεται γύρω από τα στοιχεία, ΕΚΤΌΣ των ορίων, για να κάνει το στοιχείο «να ξεχωρίζει".

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 29 *- Περίγραμμα CSS (Πηγή: https://www.w3schools.com/css/css_outline.asp)*

Κείμενο CSS

Η CSS διαθέτει πολλές ιδιότητες για τη μορφοποίηση κειμένου.

Εικόνα 30 *- Κείμενο CSS (Πηγή: https://www.w3schools.com/css/css_text.asp)*

- Χρώμα Κειμένου
- Στοίχιση Κειμένου
- Διακόσμηση κειμένου
- Μετασχηματισμός κειμένου
- Διάστιχο Κειμένου:
- Σκίαση κειμένου

Γραμματοσειρές CSS (Fonts)

- Η επιλογή της κατάλληλης γραμματοσειράς μπορεί να ενισχύσει την ταυτότητα της επωνυμίας σας.
- Η επιλογή μιας γραμματοσειράς που είναι ευανάγνωστη είναι σημαντική. Η γραμματοσειρά προσθέτει αξία στο κείμενό σας. Είναι επίσης σημαντικό να επιλέξετε το σωστό χρώμα και το σωστό μέγεθος κειμένου για τη γραμματοσειρά.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Generic Font Family	Examples of Font Names
Serif	Times New Roman Georgia Garamond
Sans-serif	Arial Verdana Helvetica
Monospace	Courier New Lucida Console Monaco
Cursive	Brush Script MT Lucida Handwriting
Fantasy	Copperplate Papyrus

Εικόνα 31 *- Γραμματοσειρές CSS (Πηγή: https://www.w3schools.com/css/css_font.asp)*

Εικονίδια CSS (icons)

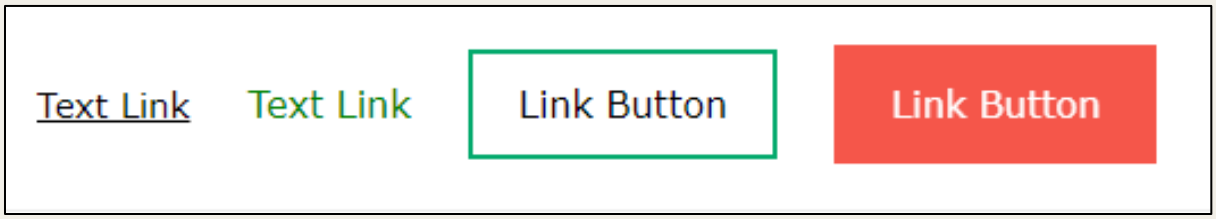
- Ο απλούστερος τρόπος για να προσθέσετε ένα εικονίδιο σε μια σελίδα HTML, είναι με μια βιβλιοθήκη εικονιδίων, όπως το Font Awesome.
- Προσθέστε το όνομα μιας συγκεκριμένης κλάσης εικονιδίων σε οποιοδήποτε ενσωματωμένο στοιχείο HTML (όπως <i> ή).
- Όλα τα εικονίδια στις παρακάτω βιβλιοθήκες εικονιδίων, είναι κλιμακούμενα διανυσματικά γραφικά που μπορούν να προσαρμοστούν με την CSS

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εικόνα 32 *- Εικονίδια CSS (Πηγή: https://www.w3schools.com/css/css_icons.asp)*

Σύνδεσμοι CSS (links)

Οι σύνδεσμοι μπορούν να διαμορφωθούν με οποιαδήποτε ιδιότητα CSS (π.χ. χρώμα, γραμματοσειρά-οικογένεια, φόντο κ.λπ.)



Εικόνα 33 *- Σύνδεσμοι CSS (Πηγή: https://www.w3schools.com/css/css_link.asp)*

Επιπλέον, οι σύνδεσμοι μπορούν να διαμορφωθούν με διαφορετικούς τρόπους ανάλογα με την κατάσταση στην οποία βρίσκονται.

Οι τέσσερις σύνδεσμοι είναι:

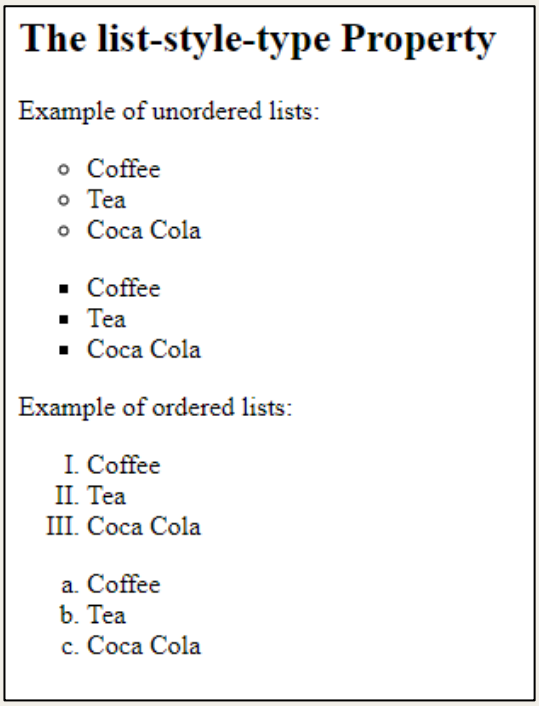
- a:link - ένας κανονικός, μη επισκέψιμος σύνδεσμος
- a:visited - ένας σύνδεσμος που έχει επισκεφθεί ο χρήστης
- a:hover - ένας σύνδεσμος όταν ο χρήστης μετακινεί το ποντίκι από πάνω του
- a:active - ένας σύνδεσμος τη στιγμή που πατιέται

Λίστες CSS (lists)

Οι ιδιότητες λιστών CSS σας επιτρέπουν:

- Να ορίσετε διαφορετικούς δείκτες αντικειμένων λίστας για ταξινομημένες λίστες
- Να ορίσετε διαφορετικούς δείκτες στοιχείων λίστας για μη ταξινομημένες λίστες
- Να ορίσετε μια εικόνα ως δείκτη στοιχείου λίστας
- Να προσθέσετε χρώματα φόντου σε λίστες και φτιάξετε λίστες αντικειμένων.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



The list-style-type Property

Example of unordered lists:

- Coffee
- Tea
- Coca Cola

- Coffee
- Tea
- Coca Cola

Example of ordered lists:

I. Coffee
II. Tea
III. Coca Cola

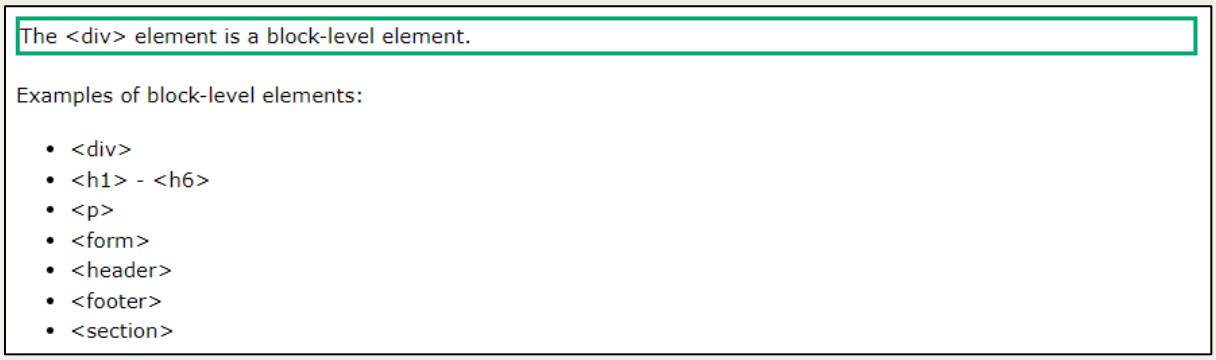
a. Coffee
b. Tea
c. Coca Cola

Εικόνα 34 *- Λίστες CSS (Πηγή: https://www.w3schools.com/css/css_list.asp)*

Προβολή CSS (display)

- Η ιδιότητα Display (Προβολή) καθορίζει εάν/πώς εμφανίζεται ένα στοιχείο.
- Κάθε στοιχείο HTML έχει μια προεπιλεγμένη τιμή εμφάνισης ανάλογα με τον τύπο του στοιχείου που είναι. Η προεπιλεγμένη τιμή εμφάνισης για τα περισσότερα στοιχεία είναι επιπέδου μπλοκ ή επιπέδου inline.

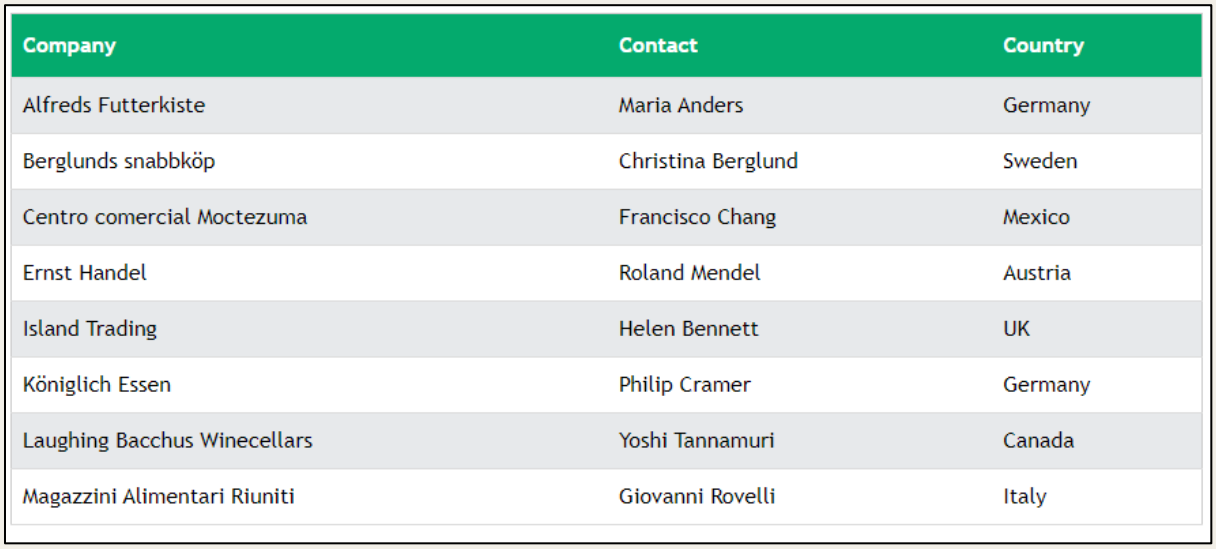
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 35 *- Προβολή CSS (Πηγή: https://www.w3schools.com/css/css_display_visibility.asp)*

Πίνακες CSS (tables)

Η εμφάνιση ενός πίνακα HTML μπορεί να βελτιωθεί σημαντικά με την CSS:



Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Berglunds snabbköp	Christina Berglund	Sweden
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Königlich Essen	Philip Cramer	Germany
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy

Εικόνα 36 *- Πίνακες CSS (Πηγή: https://www.w3schools.com/css/css_table.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μέγιστο πλάτος CSS (max-width)

- Το βασικότερο πρόβλημα που προκύπτει συνήθως σχετικά με την ετικέτα πιο πάνω είναι ότι το παράθυρο του προγράμματος περιήγησης είναι μικρότερο από το πλάτος του στοιχείου. Στην περίπτωση αυτή, το πρόγραμμα περιήγησης προσθέτει μια οριζόντια γραμμή κύλισης στη σελίδα.
- Η χρήση, αντίθετα, ενός μέγιστου πλάτους, στην περίπτωση αυτή, θα βελτιώσει τον χειρισμό μικρών παραθύρων από το πρόγραμμα περιήγησης. Αυτό είναι σημαντικό όταν αναπτύσσετε έναν ιστότοπο που θα χρησιμοποιηθεί σε μικρές συσκευές.

Εικόνα 37- *Μέγιστο πλάτος CSS (Πηγή: https://www.w3schools.com/css/css_max-width.asp)*

Τοποθέτηση στοιχείων CSS (position)

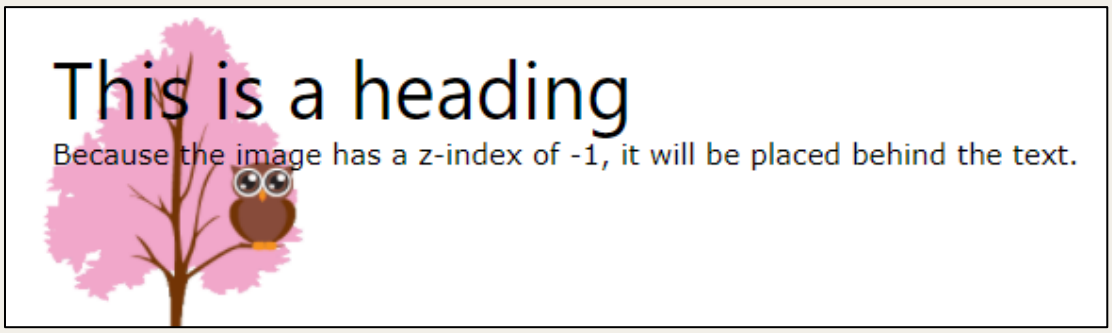
- Η ιδιότητα τοποθέτησης στοιχείων CSS μας επιτρέπει να ορίσουμε ακριβείς συντεταγμένες τοποθέτησης για οποιοδήποτε στοιχείο σε μια σελίδα.
- Υπάρχουν πέντε διαφορετικές τιμές τοποθέτησης:
- static
- relative
- fixed
- absolute

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- sticky

CSS Z-index

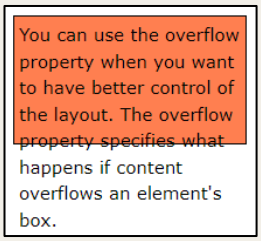
- Η ιδιότητα z-index καθορίζει τη σειρά στοίβαξης ενός στοιχείου (ποιο στοιχείο πρέπει να τοποθετηθεί μπροστά ή πίσω από τα άλλα).
- Ένα στοιχείο μπορεί να έχει θετική ή αρνητική σειρά στοίβαξης:



Εικόνα 38 *- CSS Z-index (Πηγή: https://www.w3schools.com/css/css_z-index.asp)*

Ιδιότητα Υπερχείλισης CSS (overflow)

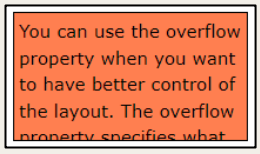
- Η ιδιότητα overflow (υπερχείλιση) ορίζει αν θα γίνεται απόσπαση του περιεχομένου ή προσθήκη γραμμών κύλισης όταν το περιεχόμενο ενός στοιχείου είναι πολύ μεγάλο για να χωρέσει σε μια οριοθετημένη περιοχή.
- Η ιδιότητα υπερχείλισης έχει τις ακόλουθες τιμές:
- visible



Εικόνα 39 *- Ιδιότητα Υπερχείλισης CSS (overflow) (Πηγή: https://www.w3schools.com/css/css_overflow.asp)*

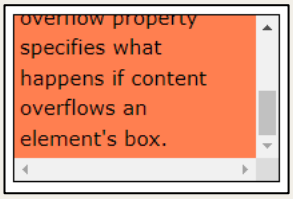
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- hidden



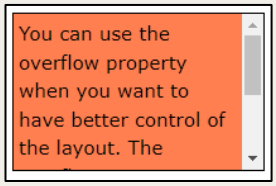
Εικόνα 40 *- Ιδιότητα Υπερχείλισης CSS (overflow) (Πηγή: https://www.w3schools.com/css/css_overflow.asp)*

- scroll



Εικόνα 41 *- Ιδιότητα Υπερχείλισης CSS (overflow) (Πηγή: https://www.w3schools.com/css/css_overflow.asp)*

- auto



Εικόνα 42 *- Ιδιότητα Υπερχείλισης CSS (overflow) (Πηγή: https://www.w3schools.com/css/css_overflow.asp)*

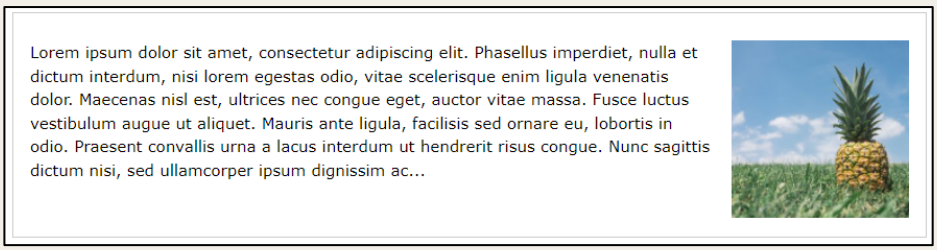
CSS Float

Η ιδιότητα float χρησιμοποιείται για την τοποθέτηση και τη μορφοποίηση του περιεχομένου

Παράδειγμα - **float: δεξιά**

Το ακόλουθο παράδειγμα ορίζει ότι μια εικόνα πρέπει να επιπλέει προς τα **δεξιά** σε ένα κείμενο:

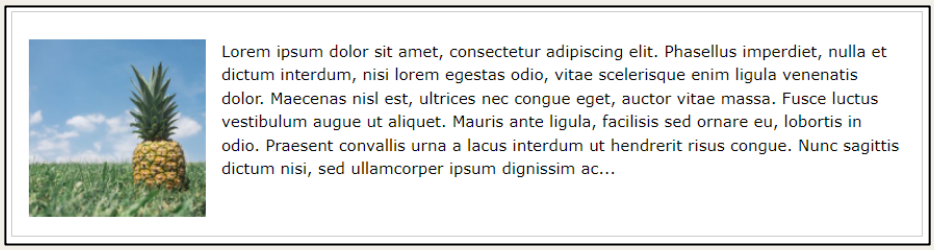
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 43 *- Float-δεξιά (Πηγή: https://www.w3schools.com/css/css_float.asp)*

Παράδειγμα - **float: αριστερά**

Το ακόλουθο παράδειγμα ορίζει ότι μια εικόνα πρέπει να επιπλέει προς τα **αριστερά** σε ένα κείμενο:

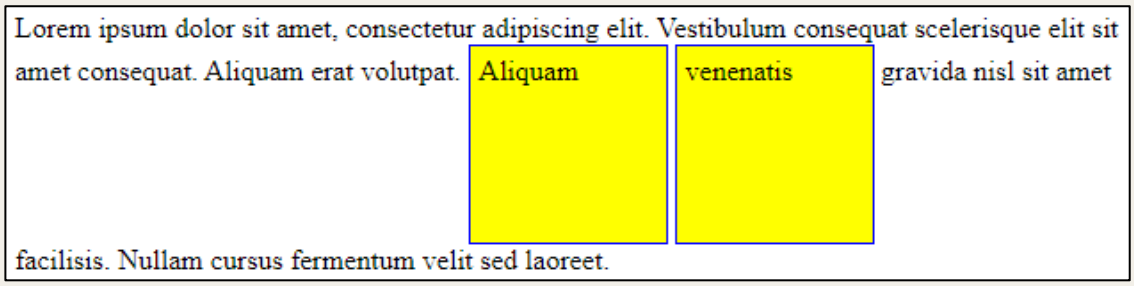


Εικόνα 44 *- Float-αριστερά (Πηγή: https://www.w3schools.com/css/css_float.asp)*

Ενσωματωμένο μπλοκ CSS (inline-block)

- Η εμφάνιση: το ενσωματωμένο μπλοκ επιτρέπει τον ορισμό του πλάτους και του ύψους σε ένα στοιχείο.
- Με την εμφάνιση: inline-block τα περιθώρια (margins) και οι επενδύσεις (paddings) που βρίσκονται πάνω και κάτω διατηρούνται.
- Η εμφάνιση: inline-block δεν προσθέτει μια αλλαγή γραμμής μετά το στοιχείο, έτσι ώστε το στοιχείο να μπορεί να περιστοιχιστεί δίπλα σε άλλα στοιχεία.

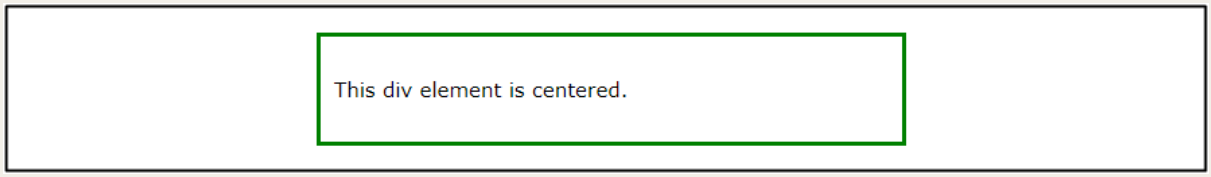
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 45 *- Ενσωματωμένο μπλοκ CSS (inline-block) (Πηγή: https://www.w3schools.com/css/css_inline-block.asp)*

Ευθυγράμμιση CSS (align)

- Για να κεντράρετε οριζόντια ένα στοιχείο μπλοκ (όπως), χρησιμοποιήστε το περιθώριο : auto
- Η ρύθμιση του πλάτους του στοιχείου θα αποτρέψει την επέκτασή του στις άκρες του περιεχομένου του.
- Στη συνέχεια, το στοιχείο θα λάβει το καθορισμένο πλάτος και ο υπόλοιπος χώρος θα διαχωριστεί εξίσου μεταξύ των περιθωρίων:



Εικόνα 46 *- Ευθυγράμμιση CSS (Πηγή: https://www.w3schools.com/css/css_align.asp)*

Συνδυαστές CSS (combinators)

- Ένας επιλογέας CSS μπορεί να περιέχει περισσότερους από έναν απλούς επιλογείς. Μεταξύ των απλών επιλογέων, μπορούμε να συμπεριλάβουμε έναν συνδυαστή.
- Υπάρχουν τέσσερις διαφορετικοί συνδυαστές στην CSS :
- επιλογέας «απόγονος»/ descendant selector (διάστημα) -> π.χ. div p { φόντο-χρώμα: κίτρινο }

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- παιδικός επιλογέας (>) -> π.χ. div > p { φόντο-χρώμα: κίτρινο }
- παρακείμενος επιλογέας αδελφών/ adjacent sibling selector (+) -> π.χ. DIV + p {χρώμα φόντου: κίτρινο }
- επιλογέας γενικών συνδυασμών αδελφών/ general sibling selector (~) -> π.χ. div ~ p { φόντο-χρώμα: κίτρινο }

Ψευδό-κλάση CSS (pseudo-class)

- Μια ψευδό-κλάση χρησιμοποιείται για να καθορίσει μια ειδική κατάσταση ενός στοιχείου.
- Για παράδειγμα, μπορεί να χρησιμοποιηθεί για:
    - Το στυλ ενός στοιχείου όταν ένας χρήστης μετακινεί το ποντίκι από πάνω του
    - Το στυλ επισκέψιμων και μη επισκέψιμων συνδέσμων με διαφορετικό τρόπο
    - Τη στιλιστική ανάπτυξη ενός στοιχείου όταν εστιάζεται

```
/* unvisited link */
a:link {
  color: #FF0000;
}

/* visited link */
a:visited {
  color: #00FF00;
}

/* mouse over link */
a:hover {
  color: #FF00FF;
}

/* selected link */
a:active {
  color: #0000FF;
}
```

Εικόνα 47 *- Ψευδό-κλάση CSS (Πηγή: https://www.w3schools.com/css/css_pseudo_classes.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CSS Ψευδο-στοιχείο (pseudo-element)

- Ένα ψευδο-στοιχείο CSS χρησιμοποιείται για το στυλ συγκεκριμένων τμημάτων ενός στοιχείου.
- Για παράδειγμα, μπορεί να χρησιμοποιηθεί για:
    - Το στυλ του πρώτου γράμματος ή γραμμής ενός στοιχείου
    - Την εισαγωγή περιεχομένου πριν ή μετά το περιεχόμενο ενός στοιχείου

```
p::first-line {
  color: #ff0000;
  font-variant: small-caps;
}
```

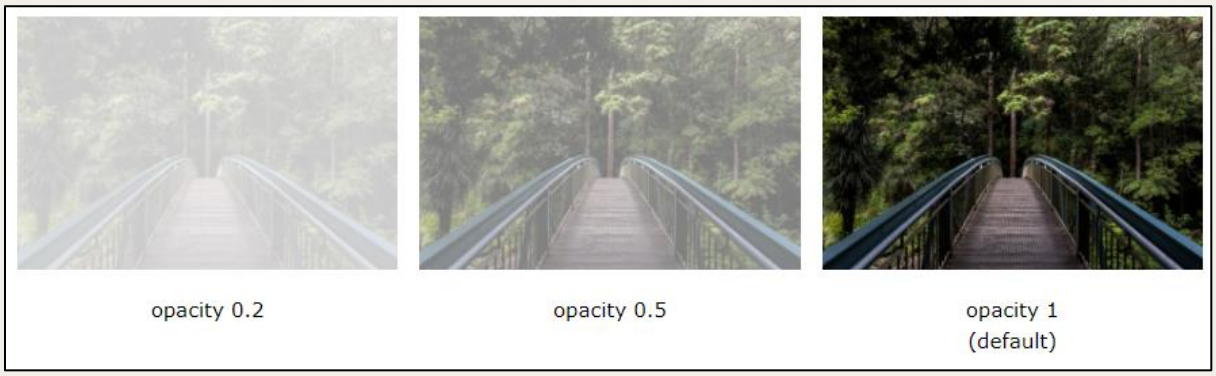
Εικόνα 48 *- CSS Ψευδο-στοιχείο (Πηγή: https://www.w3schools.com/css/css_pseudo_elements.asp)*

Το ψευδο-στοιχείο ::first-line (πρώτη γραμμή) χρησιμοποιείται για να προσθέσει ένα ειδικό στυλ στην πρώτη γραμμή ενός κειμένου.

Αδιαφάνεια CSS (opacity)

- Η ιδιότητα αδιαφάνειας καθορίζει τον βαθμό αδιαφάνειας/διαφάνειας ενός στοιχείου.
- Η ιδιότητα αδιαφάνειας μπορεί να λάβει τιμές από το 0,0 - 1,0. Η χαμηλότερη τιμή είναι και η πιο διαφανής.

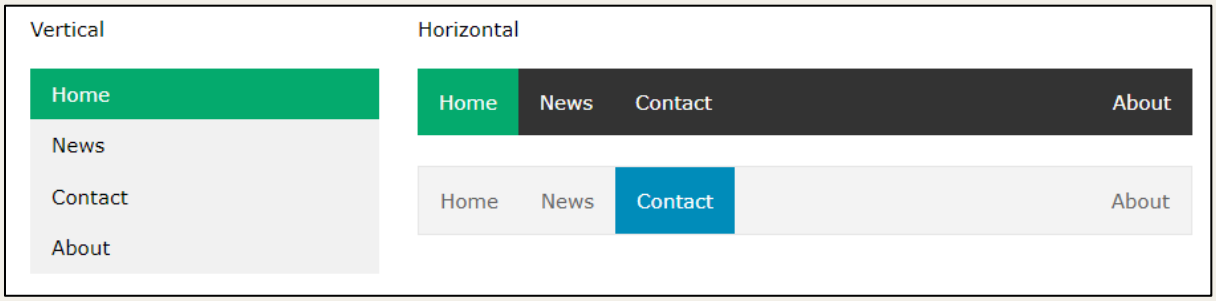
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 49 *- Αδιαφάνεια CSS (Πηγή: https://www.w3schools.com/css/css_image_transparency.asp)*

Γραμμή πλοήγησης CSS (navigation bar)

Με την CSS μπορείτε να μετατρέψετε βαρετά μενού επιλογών HTML σε ευπαρουσίαστες γραμμές πλοήγησης.

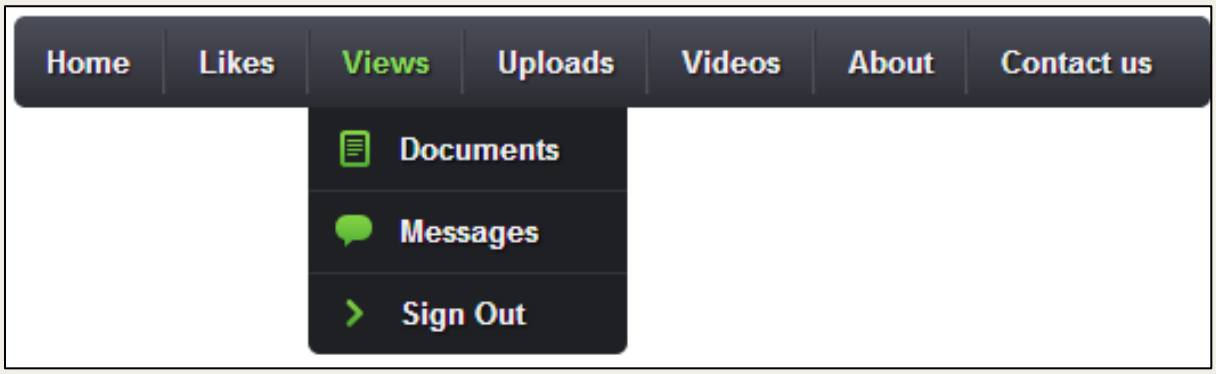


Εικόνα 50 *- Γραμμή πλοήγησης CSS (Πηγή: https://www.w3schools.com/css/css_navbar.asp)*

Αναπτυσσόμενο μενού CSS (dropdowns)

Με την CSS μπορείτε να μετατρέψετε βαρετά μενού επιλογών HTML σε ευπαρουσίαστες γραμμές πλοήγησης.

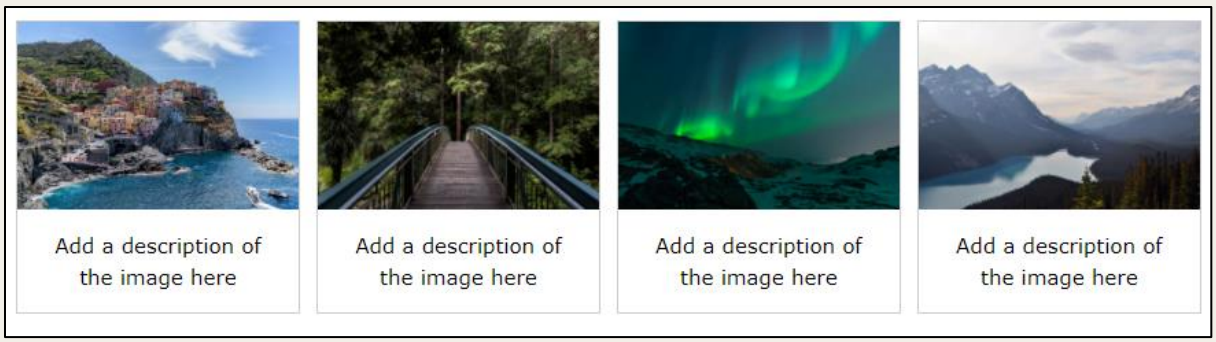
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 51 *- Αναπτυσσόμενο μενού CSS (Πηγή: https://www.w3schools.com/css/css_dropdowns.asp)*

Συλλογή εικόνων CSS (image gallery)

Η CSS μπορεί να χρησιμοποιηθεί για τη δημιουργία μιας συλλογής εικόνων.

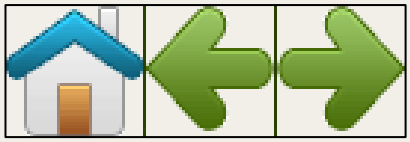


Εικόνα 52 *- Συλλογή εικόνων CSS (Πηγή: https://www.w3schools.com/css/css_image_gallery.asp)*

Εικόνες Sprites CSS

- Μια εικόνα sprites είναι μια συλλογή από εικόνες που τοποθετούνται σε μια ενιαία εικόνα.
- Μια ιστοσελίδα με πολλές εικόνες μπορεί να πάρει πολύ χρόνο να φορτωθεί και δημιουργεί πολλαπλά αιτήματα διακομιστή.
- Η χρήση των εικόνων sprites θα μειώσει τον αριθμό των αιτημάτων διακομιστή και θα αποθηκεύσει το εύρος ζώνης.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 53 *- Εικόνες Sprites CSS (Πηγή: https://www.w3schools.com/css/css_image_sprites.asp)*

Επιλογείς Attr CSS

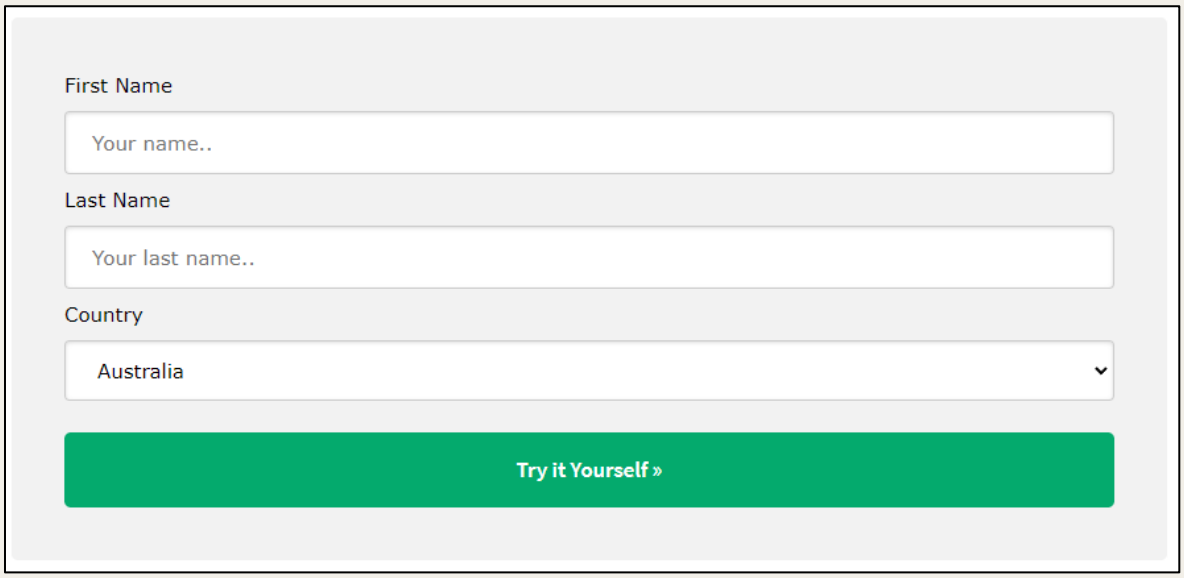
- Ο επιλογέας [attribute] χρησιμοποιείται για την επιλογή στοιχείων με μια προκαθορισμένη ιδιότητα.
- Το ακόλουθο παράδειγμα επιλέγει όλα τα <a> στοιχεία με μια ιδιότητα - στόχου:

```
a[target] {
  background-color: yellow;
}
```

Εικόνα 54 *- Επιλογείς Attr CSS (Πηγή: https://www.w3schools.com/css/css_attribute_selectors.asp)*

Φόρμες CSS (forms)

Η εμφάνιση μιας φόρμας HTML μπορεί να βελτιωθεί σημαντικά με την CSS:



Εικόνα 55 *- Φόρμες CSS (Πηγή: https://www.w3schools.com/css/css_form.asp)*

Μετρητές CSS (Counters)

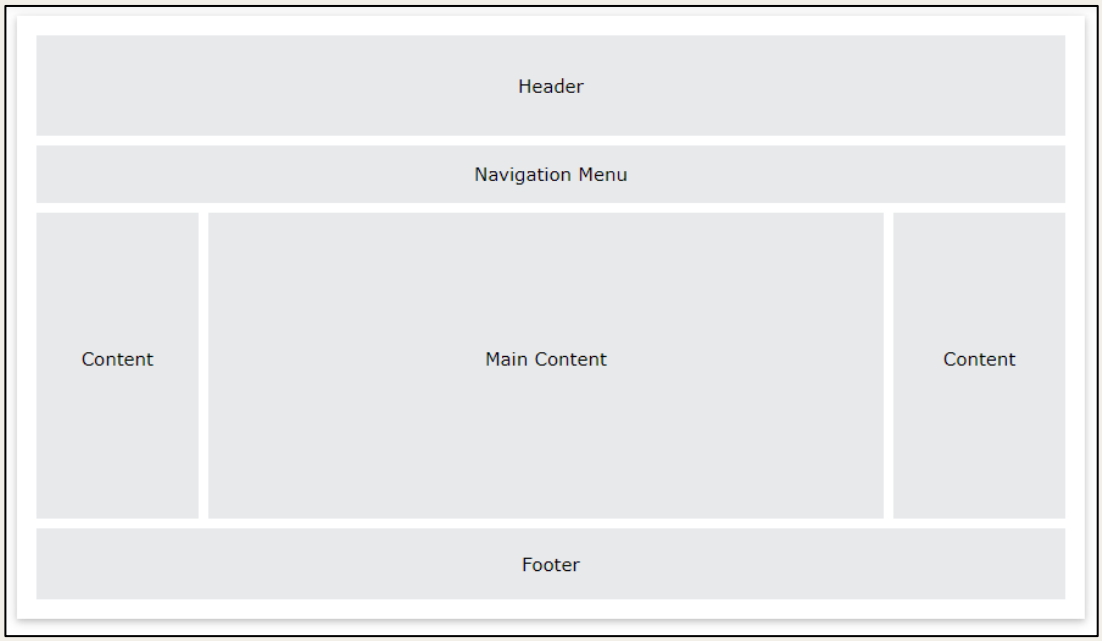
Οι μετρητές CSS είναι «μεταβλητές» που διατηρούνται από την CSS, οι τιμές των οποίων μπορούν να αυξηθούν από τους κανόνες CSS (για να παρακολουθείτε πόσες φορές χρησιμοποιούνται). Οι μετρητές CSS σάς επιτρέπουν να προσαρμόζετε την εμφάνιση του περιεχομένου με βάση την τοποθέτησή του στο έγγραφο.

Εικόνα 56 *- Μετρητές CSS (Πηγή: https://www.w3schools.com/css/css_counters.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Διάταξη Ιστοσελίδας CSS (website layout)

Ένας ιστότοπος χωρίζεται συχνά σε κεφαλίδες, μενού, περιεχόμενο και υποσέλιδο:



Εικόνα 57- *Διάταξη Ιστοσελίδας CSS (Πηγή: https://www.w3schools.com/css/css_website_layout.asp)*

Μονάδες μέτρησης CSS (units)

- Η CSS έχει πολλές διαφορετικές μονάδες για τη διαμόρφωση ενός μήκους.
- Πολλές ιδιότητες CSS λαμβάνουν τιμές «μήκους», όπως πλάτος, περιθώριο, επένδυση (padding), μέγεθος γραμματοσειράς κ.λπ.,
- Το μήκος είναι ένας αριθμός που ακολουθείται από μια μονάδα μήκους, όπως 10px, 2em, κ.λπ.,

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Unit	Description
cm	centimeters
mm	millimeters
in	inches (1in = 96px = 2.54cm)
px *	pixels (1px = 1/96th of 1in)
pt	points (1pt = 1/72 of 1in)
pc	picas (1pc = 12 pt)

Εικόνα 58 *- Μονάδες μέτρησης CSS (Πηγή: https://www.w3schools.com/css/css_units.asp)*

Ειδικότητα CSS (specificity)

- Εάν υπάρχουν δύο ή περισσότεροι κανόνες CSS που επισημαίνουν το ίδιο στοιχείο, ο επιλογέας με την υψηλότερη τιμή ειδικότητας θα «κερδίσει» και η δήλωση στυλ του θα εφαρμοστεί στο στοιχείο HTML.
- Θεωρήστε την ειδικότητα ως σκορ/βαθμό που καθορίζει ποια δήλωση στυλ θα εφαρμοστεί τελικά σε ένα στοιχείο.

```
<html>
<head>
  <style>
    p {color: red;}
  </style>
</head>
<body>

<p>Hello World!</p>

</body>
</html>
```

Εικόνα 59 *- Ειδικότητα CSS (Πηγή: https://www.w3schools.com/css/css_specificity.asp)*

Στο πιο πάνω παράδειγμα, χρησιμοποιήσαμε το στοιχείο «p» ως επιλογέα και ορίσαμε ένα κόκκινο χρώμα για το στοιχείο. Έτσι λοιπόν, το κείμενο θα είναι κόκκινο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CSS !important

- The !important είναι ένας κανόνας της CSS που χρησιμοποιείται για να προσθέσει μεγαλύτερη σημασία σε μια ιδιότητα/τιμή από το κανονικό.
- Στην πραγματικότητα, αν χρησιμοποιήσεις τον κανόνα !important, θα παρακάμψει ΌΛΟΥΣ τους προηγούμενους κανόνες στυλ για τη συγκεκριμένη ιδιότητα στο στοιχείο!

```
#myid {
  background-color: blue;
}

.myclass {
  background-color: gray;
}

p {
  background-color: red !important;
}
```

Εικόνα 60 *- !important (Πηγή: https://www.w3schools.com/css/css_important.asp)*

Μαθηματικές συναρτήσεις CSS (math function)

- Οι μαθηματικές συναρτήσεις CSS επιτρέπουν τη χρήση μαθηματικών εκφράσεων ως τιμές ιδιοτήτων. Μερικές μαθηματικές συναρτήσεις είναι: calc(), max() και min() συναρτήσεις.
- Παράδειγμα: η χρήση του calc() για τον υπολογισμό του πλάτους ενός στοιχείου :

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
#div1 {
  position: absolute;
  left: 50px;
  width: calc(100% - 100px);
  border: 1px solid black;
  background-color: yellow;
  padding: 5px;
}
```

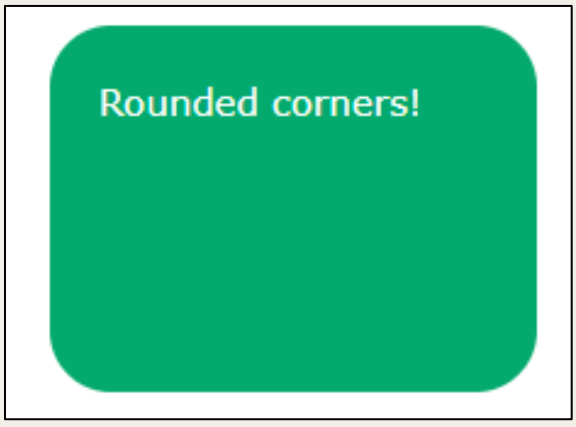
Εικόνα 61 *- Μαθηματικές συναρτήσεις CSS (Πηγή: https://www.w3schools.com/css/css_math_functions.asp)*

3.2 Προηγμένοι κανόνες σύνταξης CSS

Στρογγυλεμένες γωνιές CSS (rounded corners)

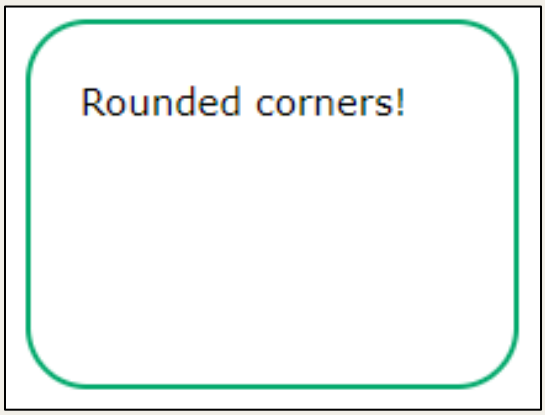
Η ιδιότητα ορίων ακτίνας (rounded corners) καθορίζει την ακτίνα των γωνιών ενός στοιχείου.

Συμβουλή: Αυτή η ιδιότητα σας επιτρέπει να προσθέσετε στρογγυλεμένες γωνίες σε στοιχεία!



Εικόνα 62 *– Στρογγυλεμένες γωνιές CSS (rounded corners) (Πηγή: https://www.w3schools.com/css/css3_borders.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



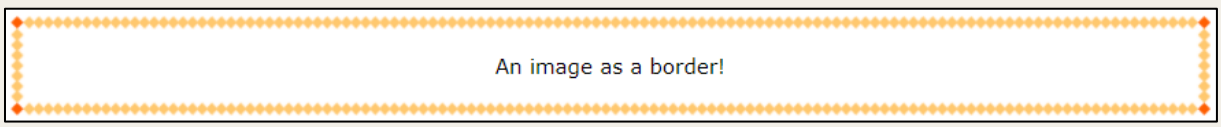
Εικόνα 63 *– Στρογγυλεμένες γωνιές CSS (rounded corners) (Πηγή: https://www.w3schools.com/css/css3_borders.asp)*

Εικόνες περιγράμματος CSS (border images)

Η ιδιότητα εικόνας περιγράμματος CSS σας επιτρέπει να καθορίσετε μια εικόνα που θα χρησιμοποιηθεί αντί για το κανονικό περίγραμμα γύρω από ένα στοιχείο.

Η ιδιότητα αυτή αποτελείται από τρία μέρη:

- Η εικόνα που θα χρησιμοποιηθεί ως περίγραμμα
- Ο καθορισμός των ορίων όπου θα τεμαχιστεί η εικόνα
- Ο καθορισμός των μεσαίων τμημάτων αν πρόκειται να επαναληφθούν ή να επεκταθούν



Εικόνα 64 *– Εικόνες περιγράμματος CSS (border images) (Πηγή: https://www.w3schools.com/css/css3_border_images.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Φόντο CSS (background)

Η CSS σας επιτρέπει να προσθέσετε πολλαπλές εικόνες φόντου για ένα στοιχείο, μέσω της ιδιότητας φόντου-εικόνας

Το ακόλουθο παράδειγμα έχει δύο εικόνες φόντου, η πρώτη εικόνα είναι ένα λουλούδι (ευθυγραμμισμένη με την κάτω δεξιά γωνιά) και η δεύτερη εικόνα είναι ένα φόντο χαρτιού (ευθυγραμμισμένη με την πάνω αριστερή γωνία)

```
#example1 {
  background-image: url(img_flwr.gif), url(paper.gif);
  background-position: right bottom, left top;
  background-repeat: no-repeat, repeat;
}
```

Εικόνα 65 *– Φόντο CSS (background) (Πηγή: https://www.w3schools.com/css/css3_backgrounds.asp)*

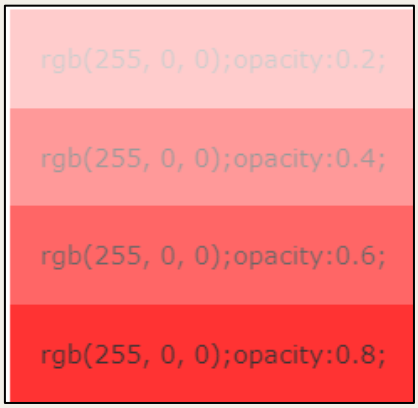
Εικόνα 66 *– Φόντο CSS (background) (Πηγή: https://www.w3schools.com/css/css3_backgrounds.asp)*

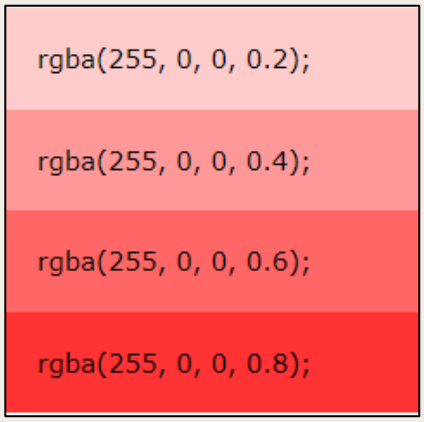
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Χρώματα CSS

- Η CSS υποστηρίζει +140 ονόματα χρωμάτων, δεκαεξαδικές τιμές, τιμές RGB, τιμές RGBA, τιμές HSL, τιμές HSLA και αδιαφάνεια.

Εικόνα 67 and 68 *– Χρώματα CSS (Πηγή: https://www.w3schools.com/css/css3_colors.asp)*





Εικόνα 69 and 70 *– Χρώματα CSS (Πηγή: https://www.w3schools.com/css/css3_colors.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Λέξεις-κλειδιά για διάφορες αποχρώσεις χρωμάτων CSS (Color Keywords)

Στο σημείο αυτό θα αναλύσουμε τις λέξεις κλειδιά των ιδιοτήτων transparent, currentcolor και inherit:

- Η ιδιότητα transparent keyword χρησιμοποιείται για να μετατρέψει ένα χρώμα διαφανές. Αυτή η ιδιότητα χρησιμοποιείται συχνά για να μετατρέψει ένα διαφανές χρώμα φόντου σε ένα στοιχείο.
- Η ιδιότητα currentcolor keyword είναι σαν μια μεταβλητή που διατηρεί την τρέχουσα τιμή της χρωματικής ιδιότητας ενός στοιχείου.

Αυτή μπορεί να φανεί χρήσιμη αν θέλετε ένα συγκεκριμένο χρώμα να είναι συμβατό με ένα στοιχείο ή μια σελίδα.

- Η ιδιότητα inherit keyword διευκρινίζει ότι μια ιδιότητα θα πρέπει να κληρονομήσει την τιμή της από το γονικό της στοιχείο.

  Η ιδιότητα inherit keyword μπορεί να χρησιμοποιηθεί για οποιαδήποτε ιδιότητα CSS και σε οποιοδήποτε στοιχείο HTML.

Διαβαθμίσεις CSS (gradient)

Οι διαβαθμίσεις CSS σάς επιτρέπουν να εμφανίζετε ομαλές μεταβάσεις μεταξύ δύο ή περισσότερων καθορισμένων χρωμάτων.
Η CSS ορίζει τρεις τύπους διαβαθμίσεων:

- Γραμμικές διαβαθμίσεις (μετακινώντας προς τα κάτω/πάνω/αριστερά/δεξιά/διαγώνια)
- Ακτινικές διαβαθμίσεις (οι οποίες ορίζονται από το κέντρο τους)
- Κωνικές διαβαθμίσεις (οι οποίες περιστρέφονται γύρω από ένα κεντρικό σημείο)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

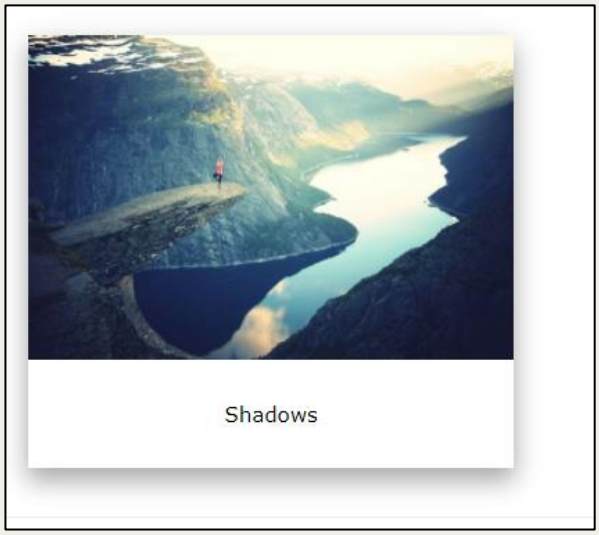
Εικόνα 71 *– Διαβαθμίσεις CSS (gradient) (Πηγή: https://www.w3schools.com/css/css3_gradients.asp)*

Σκίαση CSS (shadows)

Με την CSS μπορείτε να προσθέσετε σκιάσεις στο κείμενο και στα στοιχεία.

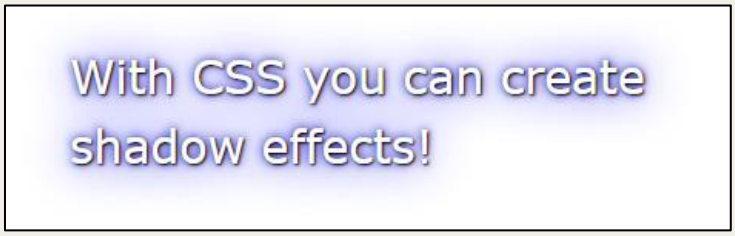
Στο κεφάλαιο αυτό θα μάθετε για τις ακόλουθες ιδιότητες :

- Σκίαση κειμένου (text-shadow)
- Σκίαση κουτιού (box-shadow)



Εικόνα 72 *– Σκίαση CSS (shadows) (Πηγή: https://www.w3schools.com/css/css3_shadows.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 73 *– Σκίαση CSS (shadows) (Πηγή: https://www.w3schools.com/css/css3_shadows.asp)*

Εφέ κειμένου CSS (text effects)

Στο κεφάλαιο αυτό θα μάθετε για το Text-overflow, word-wrap, word-break, writing-mode :

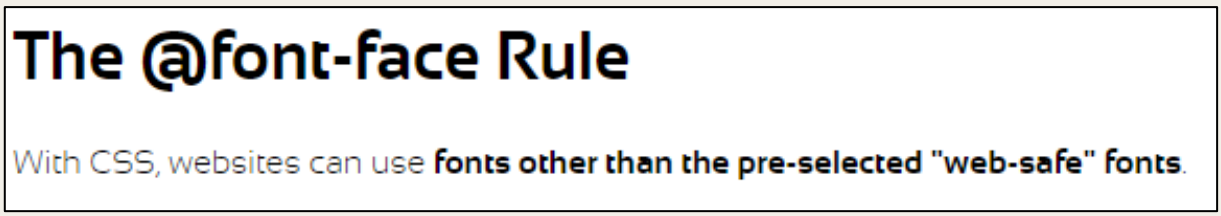
- Η ιδιότητα υπερχείλισης κειμένου (text overflow) καθορίζει τον τρόπο με τον οποίο το περιεχόμενο υπερχείλισης κειμένου που δεν εμφανίζεται θα πρέπει να επισημαίνεται στον χρήστη.
- Η ιδιότητα CSS word-wrap επιτρέπει το συλλαβικό σπάσιμο των μεγάλων λέξεων και τη μεταφορά μέρους τους στην επόμενη γραμμή.
- -Η ιδιότητα περιτύλιξης λέξεων CSS ορίζει τους κανόνες αλλαγής γραμμής για μια λέξη.
- Η ιδιότητα writing mode CSS καθορίζει εάν οι γραμμές του κειμένου είναι οριζοντίως ή καθέτως.

Γραμματοσειρές Ιστού CSS (Web Fonts)

- Οι γραμματοσειρές ιστού επιτρέπουν στους σχεδιαστές ιστοσελίδων να χρησιμοποιούν γραμματοσειρές που δεν είναι εγκατεστημένες στον υπολογιστή ενός χρήστη.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

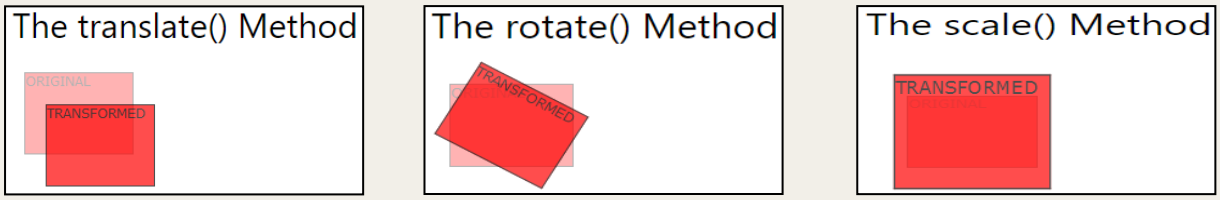
- Όταν βρείτε/αγοράσετε τη γραμματοσειρά που θέλετε να χρησιμοποιήσετε, απλά συμπεριλάβετε το αρχείο γραμματοσειράς στον διακομιστή Web και θα μεταφορτωθεί αυτόματα στον χρήστη όταν χρειαστεί.
- Οι δικές σας γραμματοσειρές ορίζονται στον κανόνα CSS @ font-face



Εικόνα 74 *– Γραμματοσειρές Ιστού CSS (Web Fonts) (Πηγή: https://www.w3schools.com/css/css3_fonts.asp)*

Μετασχηματισμοί CSS 2D (Transforms)

Οι μετασχηματισμοί CSS 2D σας επιτρέπουν να μετακινείτε, να περιστρέφετε, να κλιμακώνετε και να κυρτώνετε στοιχεία.



Εικόνα 75*– Μετασχηματισμοί CSS 2D (**Πηγή**: https://www.w3schools.com/css/css3_2dtransforms.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μετασχηματισμοί CSS 3D

Με την ιδιότητα μετασχηματισμού CSS 3D μπορείτε να χρησιμοποιήσετε τις ακόλουθες μεθόδους μετασχηματισμού 3D:

- RotateX
- RotateY
- RotateZ

Εικόνα 76– *Μετασχηματισμοί CSS 3D (Πηγή: https://www.w3schools.com/css/css3_3dtransforms.asp)*

Μεταβάσεις CSS 3D (transitions)

- Οι μεταβάσεις CSS σας επιτρέπουν να αλλάξετε τις τιμές μιας ιδιότητας ομαλά για μια δεδομένη διάρκεια.
- Για να δημιουργήσετε ένα εφέ μετάβασης, πρέπει να καθορίσετε δύο πράγματα:
    - την ιδιότητα CSS στην οποία θέλετε να προσθέσετε ένα εφέ
    - τη διάρκεια του εφέ
- Το παρακάτω παράδειγμα δείχνει ένα κόκκινο στοιχείο 100px * 100px. Το στοιχείο έχει επίσης καθορίσει ένα εφέ μετάβασης για την ιδιότητα πλάτους, με διάρκεια 2 δευτερόλεπτα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
div {
  width: 100px;
  height: 100px;
  background: red;
  transition: width 2s;
}
```

Εικόνα 77– *Μεταβάσεις CSS 3D (transitions) (Πηγή: https://www.w3schools.com/css/css3_transitions.asp)*

Κινούμενα Σχέδια CSS (Animations)

- Ένα κινούμενο σχέδιο επιτρέπει σε ένα στοιχείο να μεταλλάσσει σταδιακά από το ένα στυλ στο άλλο.
- Μπορείτε να αλλάξετε όσες ιδιότητες CSS θέλετε, όσες φορές θέλετε.
- Για να χρησιμοποιήσετε ένα κινούμενο σχέδιο CSS, πρέπει πρώτα να καθορίσετε μερικά βασικά πλαίσια για το κινούμενο σχέδιο.
- Τα πλαίσια-κλειδιά διατηρούν τα στυλ που θα έχει το στοιχείο σε συγκεκριμένες χρονικές στιγμές.
- Αυτές είναι οι κύριες ιδιότητες του κινούμενου σχεδίου:

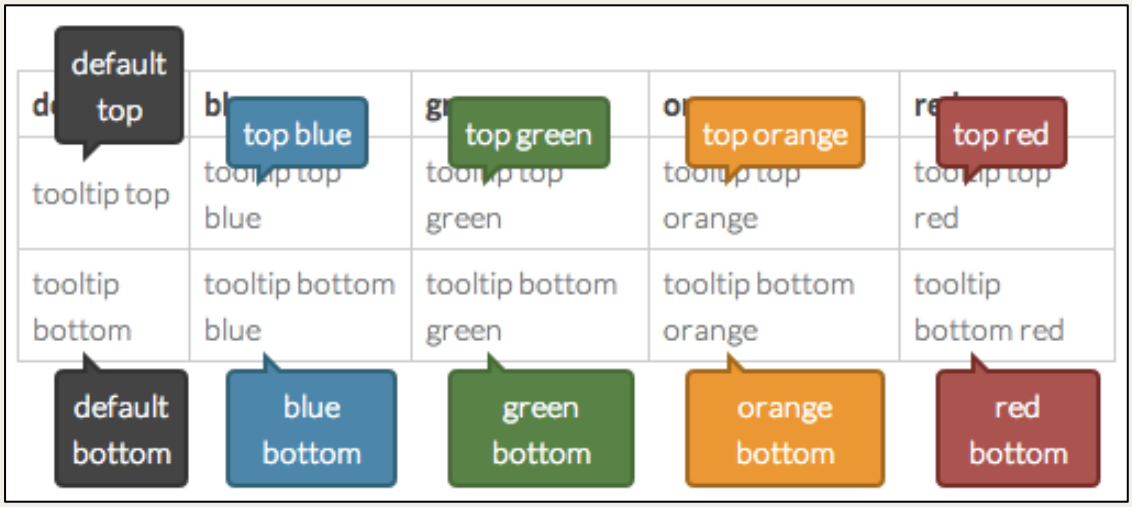
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- @keyframes
- animation-name
- animation-duration
- animation-delay
- animation-iteration-count
- animation-direction
- animation-timing-function
- animation-fill-mode
- animation

Εικόνα 78– *Κινούμενα Σχέδια CSS (Animations) (Πηγή:https://www.w3schools.com/css/css3_animations.asp)*

Συμβουλές εργαλείων CSS (Tooltips)

- Μια επεξήγηση εργαλείου χρησιμοποιείται συχνά για να διασαφηνίσει πρόσθετες πληροφορίες σχετικά με κάτι όταν ο χρήστης μετακινεί τον δείκτη του ποντικιού πάνω από ένα στοιχείο:



Εικόνα 79– *Συμβουλές εργαλείων CSS (Tooltips) (Πηγή: https://www.w3schools.com/css/css3_animations.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εικόνες στυλ CSS (Style images)

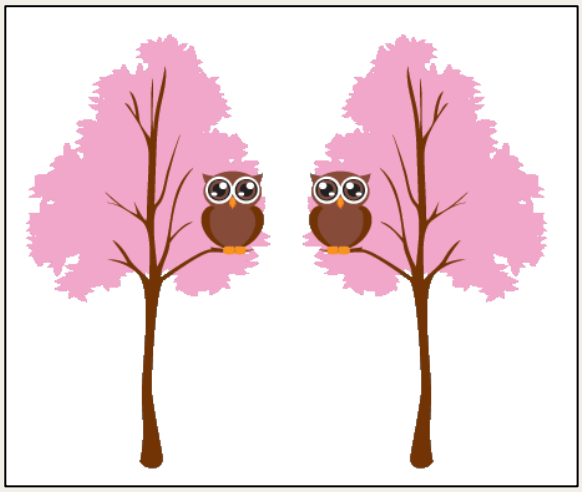
- Η χρήση της CSS για την στιλιστική ανάπτυξη των εικόνων σάς επιτρέπει να καθορίσετε ομοιόμορφα τον τρόπο εμφάνισης των εικόνων στον ιστότοπό σας με ένα περιορισμένο σύνολο κανόνων :
- προσθέστε ένα περίγραμμα και αλλάξτε το σχήμα και το μέγεθος της εικόνας, …

Εικόνα 80– *Εικόνες στυλ CSS (Style images) (Πηγή: https://www.w3schools.com/css/css3_image_reflection.asp)*

Εικόνες με εφέ αντανάκλασης CSS (image reflection)

- Η ιδιότητα αντανάκλασης πλαισίου χρησιμοποιείται για τη δημιουργία εφέ αντανάκλασης μιας εικόνας.
- Η τιμή της ιδιότητας αντανάκλασης πλαισίου μπορεί να είναι : κάτω, πάνω, αριστερά ή δεξιά

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



***Εικόνα 81**– Εικόνες με εφέ αντανάκλασης CSS (Πηγή: https://www.w3schools.com/css/css3_image_reflection.asp)*

CSS Object-fit

Η ιδιότητα Object-fit CSS χρησιμοποιείται για τον καθορισμό του τρόπου με τον οποίο ένα <img> ή <video> θα πρέπει να αλλάξει το μέγεθός του, ούτως ώστε να χωράει στον περιεχόμενο ενός στοιχείου.

Αυτή η ιδιότητα δίνει εντολή στο περιεχόμενο να γεμίσει τον περιέκτη με διάφορους τρόπους, όπως πχ. «διατήρησε αυτή την αναλογία διαστάσεων» ή «τέντωσε και χρησιμοποίησε όσο το δυνατόν περισσότερο χώρο».

Κοιτάξτε την παρακάτω εικόνα από το Παρίσι. Αυτή η εικόνα έχει πλάτος 400 εικονοστοιχεία (pixels) και ύψος 300 εικονοστοιχεία:

***Εικόνα 82**– Object fit (**Πηγή**: https://www.w3schools.com/css/css3_object-fit.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Θέση αντικειμένου CSS (Object-position)

- Ας πούμε ότι το μέρος της εικόνας που εμφανίζεται, δεν είναι τοποθετημένο όπως το θέλουμε. Για να τοποθετήσετε την εικόνα με τον επιθυμητό τρόπο, θα χρησιμοποιήσουμε την ιδιότητα θέσης αντικειμένου CSS
- Εδώ θα χρησιμοποιήσουμε την ιδιότητα θέσης αντικειμένου CSS για να τοποθετήσουμε την εικόνα έτσι ώστε το μεγάλο παλαιό κτίριο να βρίσκεται στο κέντρο:

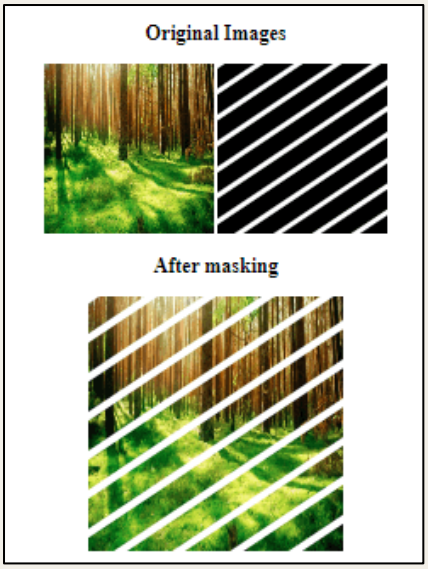
Εικόνα 83– *Θέση αντικειμένου CSS (**Πηγή**: https://www.w3schools.com/css/css3_object-position.asp)*

Ιδιότητα Μάσκας-Στρώσης CSS (Masking)

- Με την ιδιότητα CSS Masking σας δίνεται η δυνατότητα να δημιουργήσετε μια μάσκα-στρώσης την οποία θα τοποθετήσετε πάνω από ένα στοιχείο για να αποκρύψετε μερικώς ή πλήρως τμήματά του.
- Η ιδιότητα μάσκας-στρώσης CSS καθορίζει ποια μέρη της στρώσης μιας εικόνας είναι αδιαφανή, ημιδιαφανή ή διαφανή.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

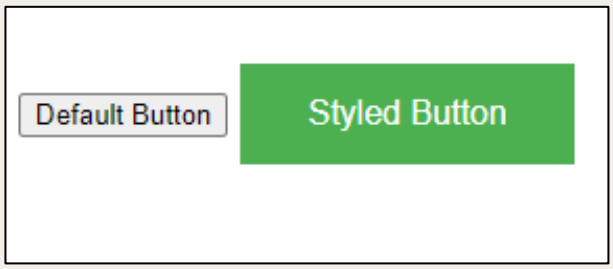
- Μια εικόνα με μάσκα-στρώσης μπορεί να είναι μια εικόνα PNG, μια εικόνα SVG, μια διαβάθμιση CSS (gradient) ή ένα <mask>στοιχείο SVG.



***Εικόνα 84**– Ιδιότητα Μάσκας-Στρώσης CSS (**Πηγή**: https://www.w3schools.com/css/css3_masking.asp)*

Κουμπιά CSS (Buttons)

- Υπάρχουν πολλές ιδιότητες για την στιλιστική ανάπτυξη ενός κουμπιού στην CSS
- Πιο κάτω παρατίθεται ένα παράδειγμα:



***Εικόνα 85**– Κουμπιά CSS (Buttons) (**Πηγή**: https://www.w3schools.com/css/css3_buttons.asp)*

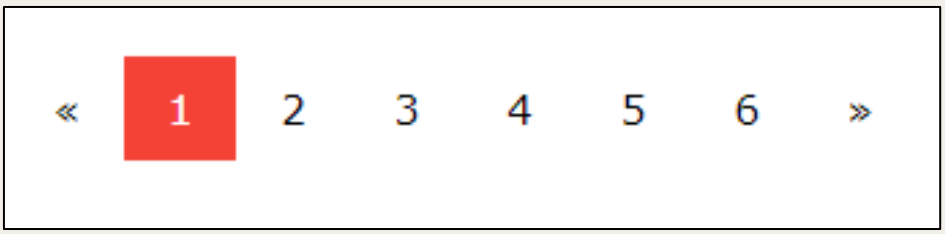
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```css
.button {
  background-color: #4CAF50;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  font-size: 16px;
  margin: 4px 2px;
}
```

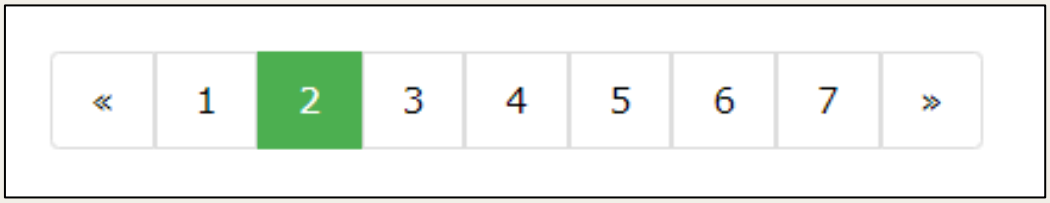
Εικόνα 86– *Κουμπιά CSS (Buttons) (**Πηγή**: https://www.w3schools.com/css/css3_buttons.asp)*

Σελιδοποίηση CSS (Pagination)

Αν έχετε μια ιστοσελίδα με πολλές σελίδες, μπορεί να θέλετε να προσθέσετε κάποιο είδος σελιδοποίησης σε κάθε σελίδα και πιο κάτω παρατίθενται μερικά παραδείγματα :



Εικόνα 87– *Σελιδοποίηση CSS (Pagination) (**Πηγή**: https://www.w3schools.com/css/css3_pagination.asp)*

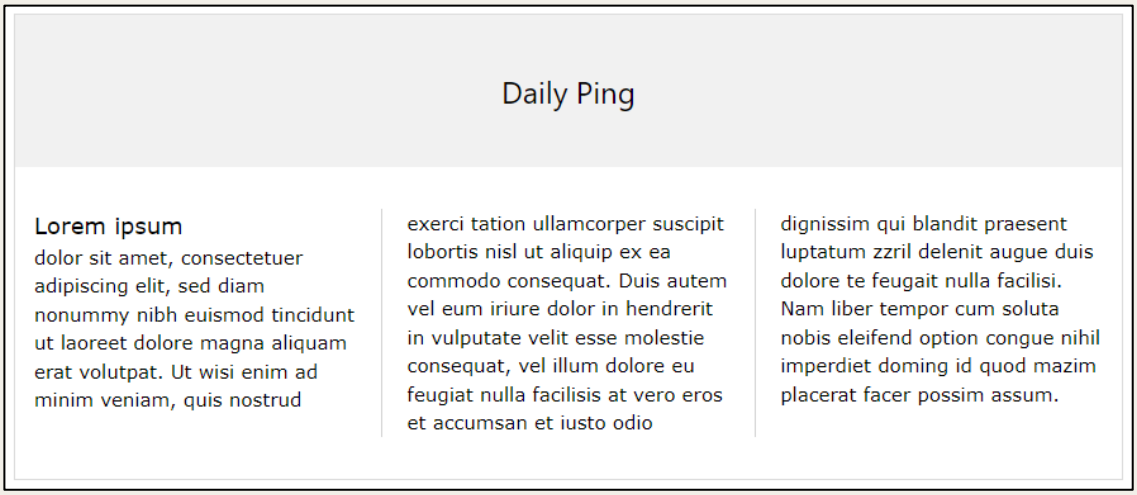


Εικόνα 88– *Σελιδοποίηση CSS (Pagination) (**Πηγή**: https://www.w3schools.com/css/css3_pagination.asp)*

Πολλαπλές στήλες CSS (Multiple Columns)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η διάταξη πολλαπλών στηλών CSS επιτρέπει τον εύκολο ορισμό πολλαπλών στηλών κειμένου - ακριβώς όπως στις εφημερίδες:



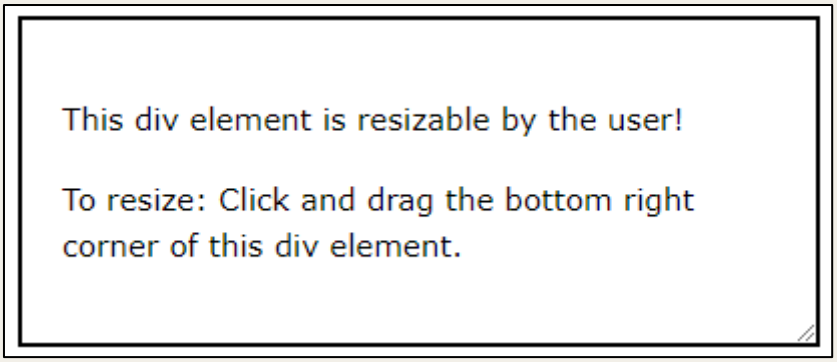
Εικόνα 89– *Πολλαπλές στήλες CSS (Multiple Columns)*

(***Πηγή***: *https://www.w3schools.com/css/css3_multiple_columns.asp*)

Διεπαφή Χρήστη CSS (User Interface)

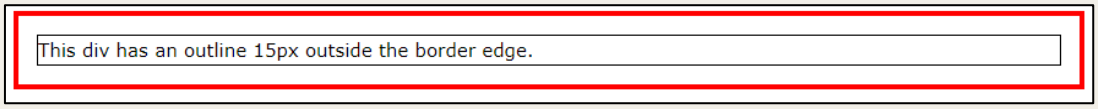
Στο κεφάλαιο αυτό θα μάθετε για τις ακόλουθες ιδιότητες διεπαφής χρήστη CSS:

- Η ιδιότητα αυξομείωσης καθορίζει εάν (και πώς) ένα στοιχείο πρέπει ή μπορεί να αλλάξει το μέγεθός του από το χρήστη.



Εικόνα 90– *Διεπαφή Χρήστη* (***Πηγή***: *https://www.w3schools.com/css/css3_user_interface.asp*)

- Η ιδιότητα outline-offset προσθέτει χώρο μεταξύ ενός περιγράμματος ή της άκρης ή του περιγράμματος ενός στοιχείου.



Εικόνα 91– *Διεπαφή Χρήστη (**Πηγή**: https://www.w3schools.com/css/css3_user_interface.asp)*

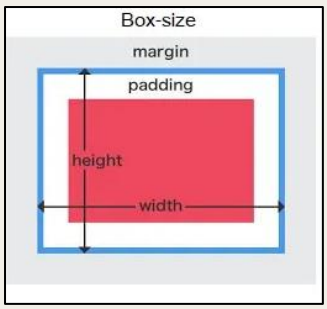
Μεταβλητές CSS (Variables)

- Η συνάρτηση var() χρησιμοποιείται για την εισαγωγή της τιμής μιας μεταβλητής CSS.
- Οι μεταβλητές CSS έχουν πρόσβαση στο Dom, πράγμα που σημαίνει ότι μπορείτε να δημιουργήσετε μεταβλητές με τοπική ή παγκόσμια εμβέλεια, να αλλάξετε τις μεταβλητές με JavnceaScript ή ακόμη και να αλλάξετε τις μεταβλητές με βάση τα ερωτήματα πολυμέσων.
- Μια καλή χρήση των μεταβλητών CSS μπορεί να γίνει για την επιλογή των χρωμάτων κατά τη φάση του σχεδιασμού. Αντί να αντιγράψετε και να επικολλήσετε τα ίδια χρώματα ξανά και ξανά, μπορείτε να τα τοποθετήσετε σε μεταβλητές.

Ιδιότητα ορισμού μεγέθους κιβωτίου CSS (Box sizing)

- Η ιδιότητα μεγέθους κουτιού στο CSS καθορίζει τον τρόπο με τον οποίο ο χρήστης θα πρέπει να υπολογίζει το συνολικό πλάτος και ύψος ενός στοιχείου, δηλαδή της επένδυσης (padding) ενός περιγράμματος, που πρέπει να περιλαμβάνονται ή όχι.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



***Εικόνα 92**– Ιδιότητα ορισμού μεγέθους κιβωτίου (**Πηγή**: https://www.w3schools.com/css/css3_box-sizing.asp)*

Ερωτήματα μέσων CSS (Media Queries)

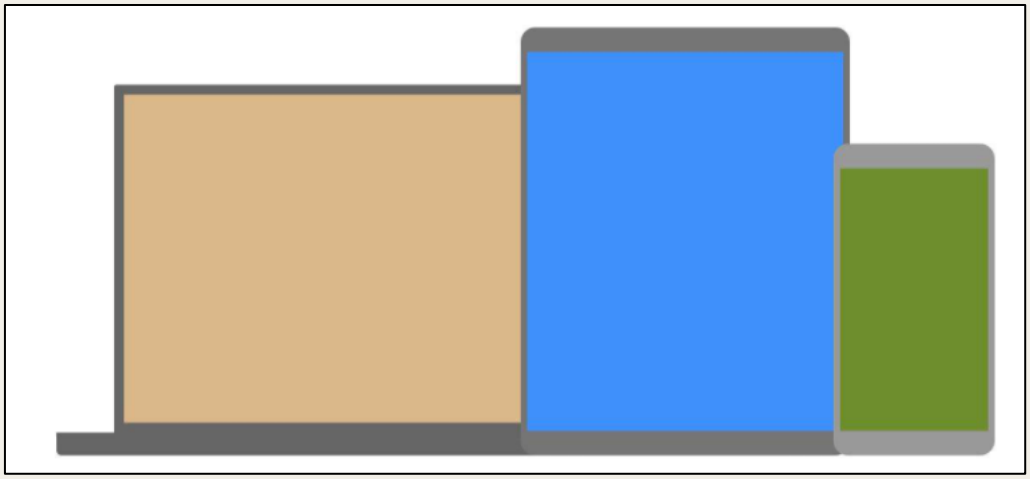
Τα ρωτήματα πολυμέσων στο CSS3 επέκτειναν την ιδέα των τύπων πολυμέσων CSS2: Αντί να αναζητήσουν έναν τύπο συσκευής, εξετάζουν τις δυνατότητες μιας συσκευής. Τα ερωτήματα πολυμέσων μπορούν να χρησιμοποιηθούν για να ελέγξουν πολλούς παραμέτρους, όπως:

- Το πλάτος και ύψος της θύρας προβολής
- Το πλάτος και το ύψος της διάταξης
- Τον προσανατολισμό (εάν δηλαδή το tablet/τηλέφωνο βρίσκεται σε λειτουργία οριζόντιας ή κατακόρυφης κλίσης (πορτρέτο) και σύνθεσης μιας εικόνας)
- Η ανάλυση/ ευκρίνεια μιας εικόνας (resolution)

Η χρήση ερωτημάτων πολυμέσων είναι μια δημοφιλής τεχνική για την εφαρμογή ενός ειδικά σχεδιασμένου φύλλου ύφους σε επιτραπέζιους υπολογιστές, φορητούς υπολογιστές, tablet και κινητά τηλέφωνα (όπως τηλέφωνα iPhone και Android).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παραδείγματα MQ CSS (MQ Examples)

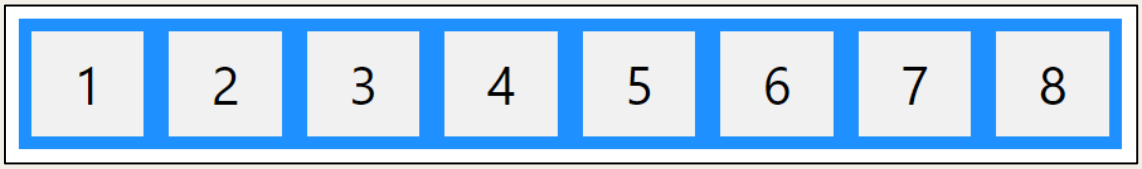


Εικόνα 93– *Παράδειγμα MQ*

(***Πηγή***: *https://www.w3schools.com/css/css3_mediaqueries_ex.asp*)

Ιδιότητα Flexbox της CSS

Η ιδιότητα Flexbox της CSS διευκολύνει τον σχεδιασμό μιας ευέλικτης δομής με άμεση ανταπόκριση χωρίς τη χρήση αριθμών κινητής υποδιαστολής (float) ή τοποθέτησης στοιχείων (position).



Εικόνα 94– *Flexbox* (***Πηγή***: *https://www.w3schools.com/css/css3_flexbox.asp*)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

3.2. Το CSS Responsive Web Design (RWD)

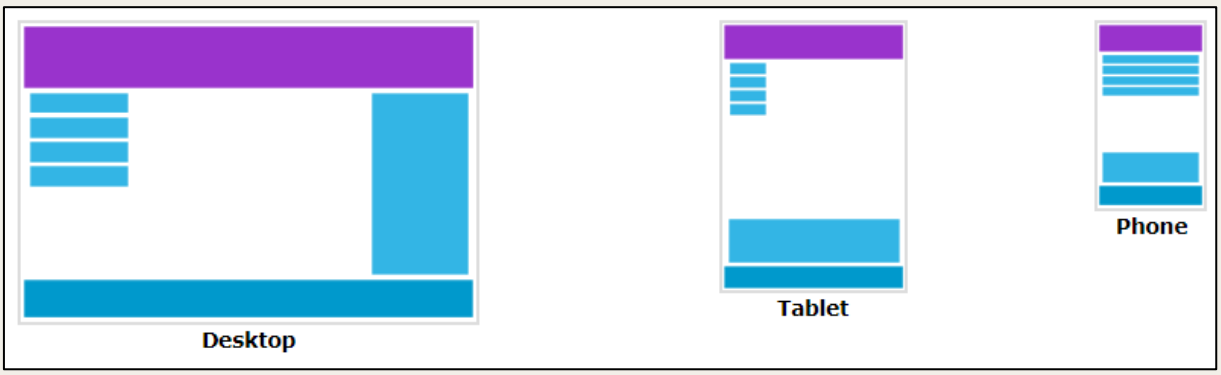
Εισαγωγή στο RWD:

Το Responsive Web Design δίνει τη δυνατότητα σχεδιασμού και κατασκευής μιας ευπαρουσίαστης ιστοσελίδας σε όλους τους υφιστάμενους τύπους συσκευών.

Το RWD χρησιμοποιεί μόνο HTML και CSS.

Μέσω του RWD, καθίσταται εφικτή η προβολή μιας ιστοσελίδας σε πολλούς διαφορετικούς τύπους συσκευών όπως οι επιτραπέζιοι υπολογιστές, τα tablet και τα κινητά τηλέφωνα.

Οι ιστοσελίδες δεν θα πρέπει να παραλείπουν πληροφορίες που είναι καταλληλότερες σε μικρότερες συσκευές, αλλά να αναδιαμορφώνουν το περιεχόμενό τους ανάλογα με το τύπο συσκευής που χρησιμοποιεί ο χρήστης ώστε να φέρουν το ίδιο άρτιο αποτέλεσμα και να προσφέρουν την ίδια εύκολη και γρήγορη εμπειρία πλοήγησης:



Εικόνα 95– *RWD (**Πηγή**: https://www.w3schools.com/css/css_rwd_intro.asp)*

Η ιδιότητα αποκριτικής ιστοσχεδίασης (Responsive Web Design/RWD) χρησιμοποιείται στους CSS και HTML για να αλλάξετε το μέγεθος, να αποκρύψετε, να συρρικνώσετε, να μεγεθύνετε ή να μετακινήσετε το περιεχόμενο ανάλογα με τις διαφορετικές ανάγκες προβολής που απορρέουν από τους διάφορους τύπους συσκευών.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Θύρα προβολής RWD (Viewpoint)

Η θύρα προβολής είναι ο ορατός χώρος μιας ιστοσελίδας από τον χρήστη.

Η θύρα προβολής ποικίλλει ανάλογα με τη συσκευή και θα είναι μικρότερη σε ένα κινητό τηλέφωνο από ό, τι σε μια οθόνη υπολογιστή.

Πριν κυκλοφορήσουν τα tablets και τα κινητά τηλέφωνα, οι ιστοσελίδες σχεδιάζονταν μόνο για οθόνες υπολογιστών και γι' αυτό συνήθως ο σχεδιασμός τους ήταν στατικός και σε σταθερό μέγεθος.

Αργότερα όμως, όταν η πλοήγηση στο διαδίκτυο επεκτάθηκε στις συσκευές tablet και στα κινητά τηλέφωνα, οι ιστοσελίδες σταθερού μεγέθους ήταν πολύ μεγάλες για να χωρέσουν στη θύρα προβολής. Για την επίλυση του εν λόγω προβλήματος, τα προγράμματα περιήγησης στις μικρότερου μεγέθους συσκευές μειώθηκαν για να μπορέσουν να χωρέσουν ολόκληρο το περιεχόμενο μιας ιστοσελίδας στην οθόνη τους.

Το αποτέλεσμα όμως δεν ήταν ακόμη άρτιο, αλλά μάλλον μια γρήγορη επίλυση του προβλήματος!!

Ρύθμιση της θύρας προβολής

Η HTML5 εισήγαγε μια μέθοδο που επιτρέπει στους σχεδιαστές ιστοσελίδων να ελέγχουν τη θύρα προβολής, μέσω της ετικέτας <meta>.

Θα πρέπει να συμπεριλάβετε την ακόλουθη ετικέτα θύρα προβολής <meta> σε όλες τις ιστοσελίδες σας:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Εικόνα 96– *Ρύθμιση της θύρας προβολής (**Πηγή**: https://www.w3schools.com/css/css_rwd_viewport.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αυτό δίνει στο πρόγραμμα περιήγησης οδηγίες σχετικά με τον τρόπο ελέγχου των διαστάσεων και αυξομείωσης του μεγέθους μιας σελίδας.

Το τμήμα width=device-width ορίζει το πλάτος της ιστοσελίδας ώστε να ακολουθεί το πλάτος της οθόνης μιας συσκευής (το οποίο θα διαφέρει ανάλογα με το μέγεθος της εκάστοτε οθόνης προβολής).

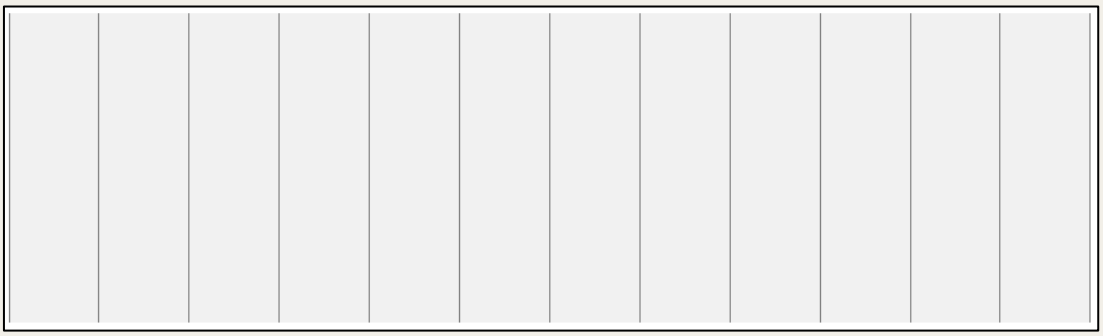
Το τμήμα initial-scale=1.0 ορίζει το αρχικό πεδίο εστίασης όταν μια ιστοσελίδα φορτώνεται σε ένα πρόγραμμα περιήγησης στον ιστό. Πιο κάτω παρατίθενται δυο παραδείγματα ιστοσελίδων, εκ των οποίων το πρώτο αποτελεί παράδειγμα ιστοσελίδας χωρίς κανόνα μετα-ετικέτας θύρας προβολής ενώ το δεύτερο αποτελεί παράδειγμα με κανόνα μετα-ετικέτας θύρας προβολής :

Εικόνα 97– *Θύρας προβολής (**Πηγή**: https://www.w3schools.com/css/css_rwd_viewport.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

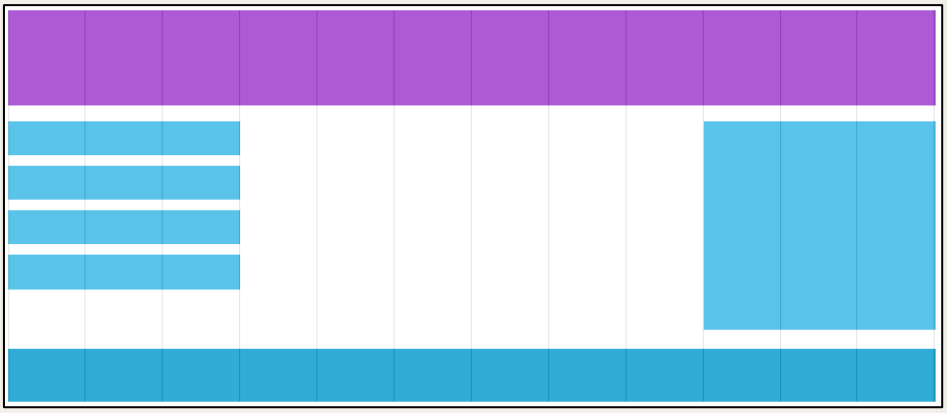
Προβολή πλέγματος RWD (Grid-View)

Πολλές ιστοσελίδες βασίζονται σε μια προβολή πλέγματος, πράγμα που σημαίνει ότι η σελίδα χωρίζεται σε στήλες:



***Εικόνα 98**– Προβολή πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_rwd_grid.asp)*

Πολλές ιστοσελίδες βασίζονται σε μια προβολή πλέγματος, πράγμα που σημαίνει ότι η σελίδα χωρίζεται σε στήλες:



***Εικόνα 99**– Προβολή πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_rwd_grid.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Κατασκευή ενός αποκριτική προβολή πλέγματος

Πρώτα βεβαιωθείτε ότι όλα τα στοιχεία HTML έχουν την ιδιότητα μεγέθους πλαισίου που έχει οριστεί στο πλαίσιο περιγράμματος (border-box). Αυτό εξασφαλίζει ότι η επένδυση και το περίγραμμα περιλαμβάνονται στο συνολικό πλάτος και ύψος των στοιχείων.

Προσθέστε τον ακόλουθο κώδικα στην CSS:

```
* {
  box-sizing: border-box;
}
```

Εικόνα 100– *Αποκριτική προβολή πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_rwd_grid.asp)*

Το ακόλουθο παράδειγμα δείχνει μια απλή αποκριτική ιστοσχεδίαση, με δύο στήλες:

25%	75%

Εικόνα 101– *Αποκριτική προβολή πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_rwd_grid.asp)*

```
.menu {
  width: 25%;
  float: left;
}
.main {
  width: 75%;
  float: left;
}
```

Εικόνα 102– *Αποκριτική προβολή πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_rwd_grid.asp)*

Ερωτήματα πολυμέσων RWD (Media Queries)

- Ένα ερώτημα πολυμέσων είναι μια τεχνική CSS που εισάχθηκε στο CSS3.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Χρησιμοποιεί τον κανόνα @media για να συμπεριλάβει ένα μπλοκ ιδιοτήτων CSS μόνο αν μια συγκεκριμένη συνθήκη είναι αληθής.

- Εάν το παράθυρο του προγράμματος περιήγησης είναι 600px ή μικρότερο, το χρώμα φόντου θα είναι γαλάζιο:

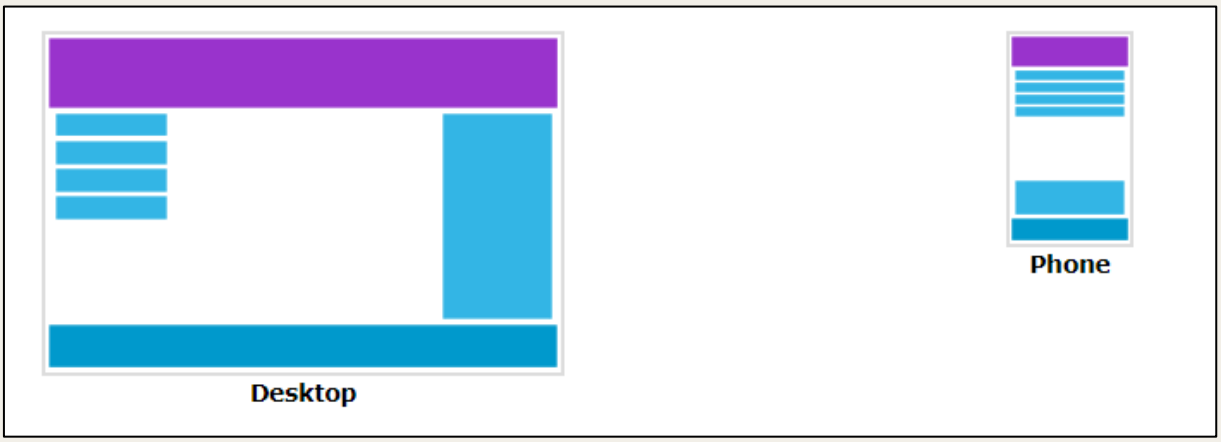
```
@media only screen and (max-width: 600px) {
  body {
    background-color: lightblue;
  }
}
```

Εικόνα 103– *Ερωτήματα πολυμέσων (**Πηγή**: https://www.w3schools.com/css/css_rwd_mediaqueries.asp)*

Προσθήκη σημείου διακοπής (breakpoint)

Νωρίτερα στον παρόντα οδηγό, κατασκευάσαμε μια ιστοσελίδα με σειρές και στήλες, η οποία ανταποκρινόταν μεν, αλλά δεν φαινόταν ευπαρουσίαστη δε, σε μια μικρή οθόνη προβολής.

Τα ερωτήματα πολυμέσων μπορούν να μας βοηθήσουν σχετικά με αυτό. Μπορούμε να προσθέσουμε ένα σημείο διακοπής όπου ορισμένα μέρη του σχεδιασμού θα συμπεριφέρονται διαφορετικά σε κάθε πλευρά του σημείου διακοπής.



Εικόνα 104– *Ερωτήματα πολυμέσων (**Πηγή**: https://www.w3schools.com/css/css_rwd_mediaqueries.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Χρησιμοποιήστε ένα ερώτημα πολυμέσων για να προσθέσετε ένα σημείο διακοπής στο 768px:

Παράδειγμα: Όταν η οθόνη (ένα παράθυρο προγράμματος περιήγησης) γίνεται μικρότερη από 768px, κάθε στήλη πρέπει να έχει πλάτος 100%:

```
/* For desktop: */
.col-1 {width: 8.33%;}
.col-2 {width: 16.66%;}
.col-3 {width: 25%;}
.col-4 {width: 33.33%;}
.col-5 {width: 41.66%;}
.col-6 {width: 50%;}
.col-7 {width: 58.33%;}
.col-8 {width: 66.66%;}
.col-9 {width: 75%;}
.col-10 {width: 83.33%;}
.col-11 {width: 91.66%;}
.col-12 {width: 100%;}

@media only screen and (max-width: 768px) {
  /* For mobile phones: */
  [class*="col-"] {
    width: 100%;
  }
}
```

Εικόνα 105– *Ερωτήματα πολυμέσων (**Πηγή**: https://www.w3schools.com/css/css_rwd_mediaqueries.asp)*

Εικόνες RWD

- Εάν η ιδιότητα πλάτους έχει οριστεί σε ένα συγκεκριμένο ποσοστό και η ιδιότητα ύψους έχει οριστεί ως “auto”, η εικόνα θα ανταποκρίνεται και θα κλιμακώνεται προς τα πάνω και προς τα κάτω:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
img {
  width: 100%;
  height: auto;
}
```

Εικόνα 106– *Εικόνα (**Πηγή**: https://www.w3schools.com/css/css_rwd_images.asp)*

- Εάν η ιδιότητα μέγιστου πλάτους έχει οριστεί στο 100%, η εικόνα θα μειωθεί αν χρειαστεί, αλλά ποτέ δεν θα αυξηθεί για να είναι μεγαλύτερη από το αρχικό της μέγεθος:

```
img {
  max-width: 100%;
  height: auto;
}
```

Εικόνα 107– *Η ιδιότητα μέγιστου πλάτους (**Πηγή**: https://www.w3schools.com/css/css_rwd_images.asp)*

Σημειώστε ότι στο παραπάνω παράδειγμα, η εικόνα μπορεί να αυξηθεί ώστε να είναι μεγαλύτερη από το αρχικό της μέγεθος. Μια καλύτερη λύση, σε πολλές περιπτώσεις, θα ήταν η χρήση της ιδιότητας μέγιστου πλάτους (max-width).

Χρήση της ιδιότητας μέγιστου πλάτους:

Η ιδιότητα μέγιστου πλάτους έχει οριστεί στο 100%, η εικόνα θα μειωθεί αν χρειαστεί, αλλά ποτέ δεν θα αυξηθεί για να είναι μεγαλύτερη από το αρχικό της μέγεθος:

```
img {
  max-width: 100%;
  height: auto;
}
```

Εικόνα 108– *Η ιδιότητα μέγιστου πλάτους (**Πηγή**: https://www.w3schools.com/css/css_rwd_images.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Βίντεο RWD

- Εάν η ιδιότητα πλάτους έχει οριστεί στο 100%, το πρόγραμμα αναπαραγωγής βίντεο θα ανταποκρίνεται και θα αυξομειώνεται προς τα πάνω και προς τα κάτω:

```
video {
  width: 100%;
  height: auto;
}
```

Εικόνα 109– *Βίντεο (**Πηγή**: https://www.w3schools.com/css/css_rwd_videos.asp)*

- Εάν η ιδιότητα μέγιστου πλάτους έχει οριστεί στο 100%, το πρόγραμμα αναπαραγωγής βίντεο θα μειωθεί αν χρειαστεί, αλλά δεν θα αυξηθεί ποτέ ώστε να είναι μεγαλύτερο από το αρχικό του μέγεθος:

```
video {
  max-width: 100%;
  height: auto;
}
```

Εικόνα 110– *Βίντεο (**Πηγή**: https://www.w3schools.com/css/css_rwd_videos.asp)*

Σημειώστε ότι στο παραπάνω παράδειγμα, το πρόγραμμα αναπαραγωγής βίντεο μπορεί να αυξηθεί ώστε να είναι μεγαλύτερο από το αρχικό του μέγεθος. Μια καλύτερη λύση, σε πολλές περιπτώσεις, θα ήταν η χρήση της ιδιότητας μέγιστου πλάτους.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Χρήση της ιδιότητας μέγιστου πλάτους (max-width):

Εάν η ιδιότητα μέγιστου πλάτους έχει οριστεί στο 100%, το πρόγραμμα αναπαραγωγής βίντεο θα μειωθεί αν χρειαστεί, αλλά δεν θα αυξηθεί ποτέ ώστε να είναι μεγαλύτερο από το αρχικό του μέγεθος:

```
video {
  max-width: 100%;
  height: auto;
}
```

Εικόνα 111– *Βίντεο (**Πηγή**: https://www.w3schools.com/css/css_rwd_videos.asp)*

Πλαίσια RWD (Frameworks)

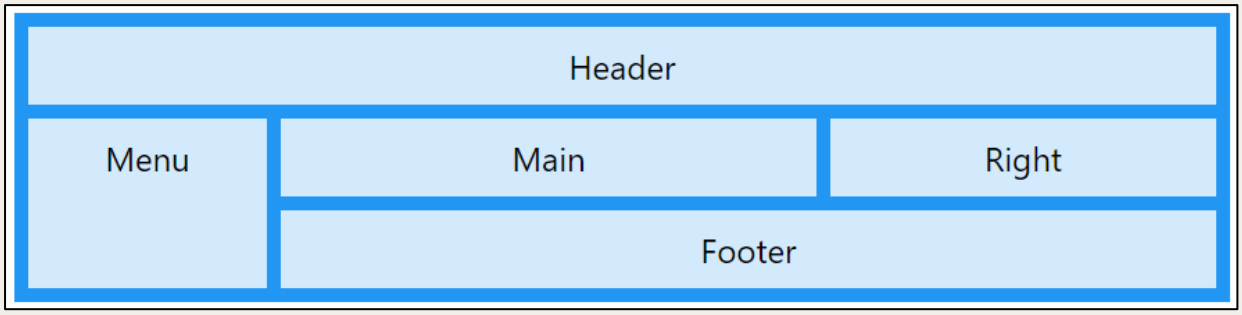
- Υπάρχουν πολλά δωρεάν CSS Frameworks που προσφέρουν αποκριτική ιστοσχεδίαση (Responsive Design).
- Ένας πολύ καλός τρόπος για να δημιουργήσετε ένα σχεδιασμό ιστοσελίδας που να ανταποκρίνεται, είναι να χρησιμοποιήσετε ένα φύλλο ύφους που επίσης ανταποκρίνεται, όπως το W3.CSS
- Το W3.CSS διευκολύνει την ανάπτυξη ιστότοπων που το περιεχόμενό τους είναι ευπαρουσίαστο σε οποιοδήποτε μέγεθος!
- Ένα άλλο δημοφιλές πλαίσιο είναι το Bootstrap. Χρησιμοποιεί την HTML και την CSS για να αναπτύξει και να σχεδιάσει ιστοσελίδες που ανταποκρίνονται

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

3.3. Πλέγμα CSS (Grid)

Εισαγωγή στο CSS Grid (Grid Layout)

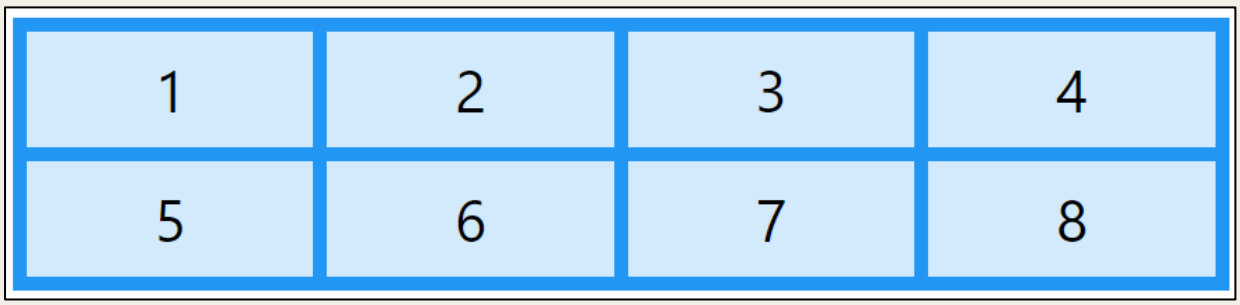
- Η ιδιότητα διάταξης πλέγματος CSS προσφέρει ένα σύστημα διάταξης ενός ιστότοπου με βάση το πλέγμα, τις σειρές και τις στήλες, καθιστώντας έτσι ευκολότερο τον σχεδιασμό και την ανάπτυξη ιστοσελίδων χωρίς να χρειάζεται η χρήση πλωτήρων και τοποθέτησης στοιχείων.



Εικόνα 112– *Πλέγμα CCS (**Πηγή**: https://www.w3schools.com/css/css_grid.asp)*

Δοχείο/κοντέινερ Πλέγματος (Grid Container)

- Για να κάνετε ένα στοιχείο HTML να συμπεριφέρεται ως Grid Container, πρέπει να ορίσετε την ιδιότητα προβολής (display) σε μια δομή πλέγματος ή δομή ενσωματωμένου πλέγματος (inline-grid)
- Ένα Grid Container αποτελείται από στοιχεία δομής πλέγματος, τοποθετημένα μέσα σε στήλες και σειρές.

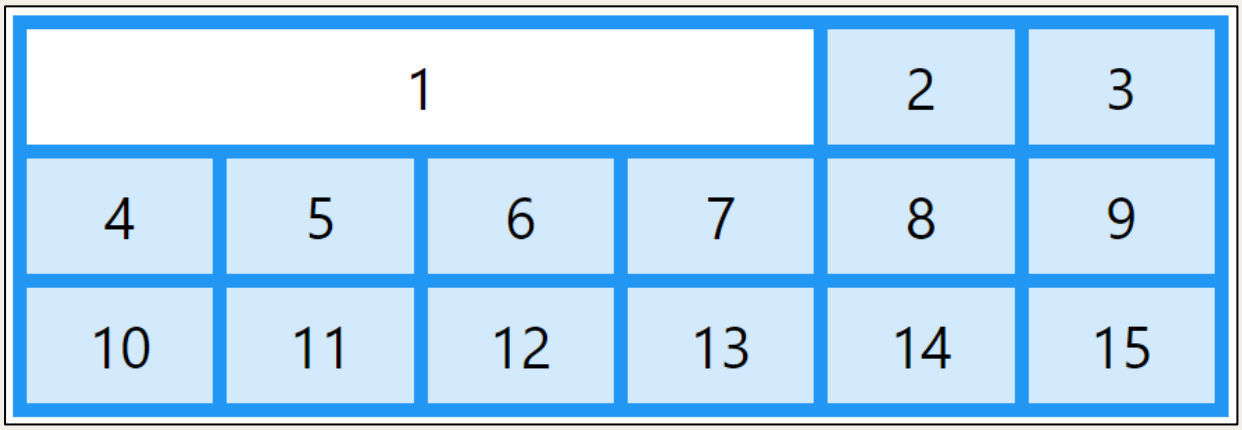


Εικόνα 113– *Δοχείο πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_grid_container.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στοιχείο πλέγματος (grid item)

- Ένα *δοχείο* πλέγματος περιέχει *στοιχεία* πλέγματος.
- Από προεπιλογή, ένα δοχείο/κοντέινερ πλέγματος έχει ένα στοιχείο πλέγματος για κάθε στήλη, σε κάθε γραμμή, αλλά μπορείτε να διαμορφώσετε τα στοιχεία πλέγματος έτσι ώστε να καλύπτουν πολλαπλές στήλες ή/και γραμμές.
- Η ιδιότητα στήλης πλέγματος (grid-column) ορίζει σε ποια/ες στήλη(ες) θα τοποθετηθεί/ουν σε ένα στοιχείο.
- Μπορείτε να ορίσετε πού θα ξεκινήσει ένα στοιχείο και πού θα τελειώσει.



Εικόνα 114– *Στοιχεία πλέγματος (**Πηγή**: https://www.w3schools.com/css/css_grid_item.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

3.5. CSS SASS

Εισαγωγή στο SASS

Τι είναι το SASS;

- SASS σημαίνει Syntactically Awesome Stylesheet
- Το SASS είναι μια επέκταση του CSS
- Το SASS είναι προ-επεξεργαστής του CSS
- Το SASS είναι πλήρως συμβατό με όλες τις εκδόσεις του CSS
- Το SASS μειώνει τις επαναλήψεις του CSS και ως εκ τούτου εξοικονομεί χρόνο
- Το SASS σχεδιάστηκε από τον Hampton Catlin και αναπτύχθηκε από τη Natalie Weizenbaum το 2006
- Το SASS μπορεί να το μεταφορτώσει και να το χρησιμοποιήσει κανείς ελεύθερα

Γιατί να χρησιμοποιήσετε το SASS;

Τα φύλλα ύφους γίνονται διαρκώς μεγαλύτερα, πιο περίπλοκα και πιο δύσκολα μπορούν να διατηρηθούν. Εδώ είναι το σημείο στο οποίο ένας προ-επεξεργαστής CSS μπορεί να βοηθήσει.

Το SASS σας επιτρέπει να χρησιμοποιείτε λειτουργίες που δεν υπάρχουν στο CSS, όπως μεταβλητές, ένθετους κανόνες, mixins, εισαγωγές, inherits, ενσωματωμένες λειτουργίες κτλ.

Ένα απλό παράδειγμα γιατί το SASS είναι χρήσιμο

Ας πούμε ότι έχουμε μια ιστοσελίδα με τρία κύρια χρώματα όπως φαίνονται πιο κάτω:

```
#a2b9bc
#b2ad7f
#878f99
```

Εικόνα 115– *Χρώματα στο SASS (**Πηγή**: https://www.w3schools.com/sass/sass_intro.phph)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πόσες φορές πρέπει να πληκτρολογήσουμε αυτές τις δεκαεξαδικές τιμές; Πολλές φορές. Και πόσες φορές πρέπει να πληκτρολογήσουμε για τις τιμές των αποχρώσεων των ίδιων χρωμάτων;

Αντί λοιπόν να πληκτρολογήσουμε τις προαναφερθέντες τιμές πολλές φορές, μπορούμε να χρησιμοποιήσετε το SASS και να γράψουμε μόνο τον κώδικα πιο κάτω:

```
/* define variables for the primary colors */
$primary_1: #a2b9bc;
$primary_2: #b2ad7f;
$primary_3: #878f99;

/* use the variables */
.main-header {
  background-color: $primary_1;
}

.menu-left {
  background-color: $primary_2;
}

.menu-right {
  background-color: $primary_3;
}
```

Εικόνα 116– *Μεταβλητές του SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Έτσι λοιπόν, όταν χρησιμοποιείτε το SASS, και το κύριο χρώμα αλλάζει, το μόνο που χρειάζεται είναι να το αλλάξετε σε ένα μέρος.

Πώς λειτουργεί το SASS;

Ένα πρόγραμμα περιήγησης δεν καταλαβαίνει τον κώδικα SASS. Ως εκ τούτου, θα χρειαστείτε έναν προ-επεξεργαστή SASS για να μετατρέψετε τον κώδικα SASS σε έναν τυπικό κώδικα τύπου CSS.

Αυτή η διαδικασία ονομάζεται μεταγλώττιση. Για τη μεταγλώττιση, θα πρέπει να χρησιμοποιήσετε έναν αναμεταδότη (ένα είδος προγράμματος) στον οποίο θα δώσετε τον κώδικα Sass για να το λάβετε πίσω στη συνέχεια υπό μορφή CSS.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τύπος αρχείου

Τα αρχεία SASS έχουν την επέκταση αρχείου ".scss".

Σχόλια SASS

Το SASS υποστηρίζει τυπικά σχόλια CSS/* comment */, και επιπλέον υποστηρίζει ενσωματωμένα (inline) σχόλια // comment:

```
/* define primary colors */
$primary_1: #a2b9bc;
$primary_2: #b2ad7f;

/* use the variables */
.main-header {
  background-color: $primary_1; // here you can put an inline comment
}
```

Εικόνα 117– *Σχόλια (**Πηγή**: https://www.w3schools.com/sass/sass_intro.php)*

Μεταβλητές SASS

Οι μεταβλητές είναι ένας τρόπος αποθήκευσης πληροφοριών που μπορείτε να χρησιμοποιήσετε αργότερα.

Με το SASS, μπορείτε να αποθηκεύσετε πληροφορίες σε μεταβλητές, όπως:

- ακολουθίες χαρακτήρων
- αριθμοί
- χρώματα
- λογικές τιμές
- λίστες

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- μηδενικές τιμές nulls

Το SASS χρησιμοποιεί το σύμβολο $, ακολουθούμενο από ένα όνομα, για να δηλώσει μεταβλητές:

```
$variablename: value;
```

Εικόνα 118– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Στο ακόλουθο παράδειγμα δηλώνονται 4 μεταβλητές που ονομάζονται myFont, myColor, myFontSize και myWidth. Αφού δηλωθούν οι μεταβλητές, μπορείτε να χρησιμοποιήσετε τις μεταβλητές όπου θέλετε:

```
$myFont: Helvetica, sans-serif;
$myColor: red;
$myFontSize: 18px;
$myWidth: 680px;

body {
  font-family: $myFont;
  font-size: $myFontSize;
  color: $myColor;
}

#container {
  width: $myWidth;
}
```

Εικόνα 119– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Έτσι, όταν το αρχείο SASS μεταγλωττίζεται, παίρνει τις μεταβλητές (myFont, myColor, κλπ.) και τις εξάγει υπό τη μορφή μεταβλητών τιμών CSS, όπως φαίνονται πιο κάτω:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: red;
}

#container {
  width: 680px;
}
```

Εικόνα 120– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Πεδίο εφαρμογής μεταβλητών SASS

Οι μεταβλητές SASS είναι διαθέσιμες μόνο στο επίπεδο ένθεσης (nesting) στο οποίο ορίζονται.

Παρατηρήστε το ακόλουθο παράδειγμα:

```
$myColor: red;

h1 {
  $myColor: green;
  color: $myColor;
}

p {
  color: $myColor;
}
```

Εικόνα 121– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Το χρώμα του κειμένου μέσα σε μια ετικέτα <p> θα είναι κόκκινο ή πράσινο; Θα είναι κόκκινο!

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο άλλος ορισμός, $ myColor: green; είναι μέσα στην ετικέτα <h1>, και θα είναι διαθέσιμος μόνο εκεί!

Έτσι, το αποτέλεσμα CSS θα είναι:

```
h1 {
  color: green;
}

p {
  color: red;
}
```

Εικόνα 122– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Αυτή είναι λοιπόν η προεπιλεγμένη συμπεριφορά για το πεδίο εφαρμογής μεταβλητών SASS.

Η χρήση της μεταβλητής SASS !global

Η προεπιλεγμένη συμπεριφορά για το πεδίο εφαρμογής μεταβλητών μπορεί να παρακαμφθεί χρησιμοποιώντας την μεταβλητή !global.

Η μεταβλητή !global δείχνει ότι μια μεταβλητή είναι καθολική, πράγμα που σημαίνει ότι είναι προσβάσιμη σε όλα τα επίπεδα.

Παρατηρήστε το ακόλουθο παράδειγμα (το ίδιο όπως παραπάνω, αλλά με την προσθήκη μεταβλητής !global):

```
$myColor: red;

h1 {
  $myColor: green !global;
  color: $myColor;
}

p {
  color: $myColor;
}
```

Εικόνα 123– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τώρα το χρώμα του κειμένου μέσα σε μια ετικέτα <p> θα είναι πράσινο!

Έτσι, η έξοδος CSS θα είναι:

```
h1 {
  color: green;
}

p {
  color: green;
}
```

Εικόνα 124– *Μεταβλητή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_variables.php)*

Ένθετοι Κανόνες SASS (Nested)

Το Sass σας επιτρέπει να τοποθετήσετε επιλογείς CSS με τον ίδιο τρόπο όπως το HTML.

Παρατηρήστε ένα παράδειγμα κώδικα SASS για την πλοήγηση ενός ιστότοπου:

```
nav {
  ul {
    margin: 0;
    padding: 0;
    list-style: none;
  }
  li {
    display: inline-block;
  }
  a {
    display: block;
    padding: 6px 12px;
    text-decoration: none;
  }
}
```

Εικόνα 125– *Ένθετοι SASS (**Πηγή**: https://www.w3schools.com/sass/sass_nesting.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παρατηρήστε ότι στο SASS, το ul, το li και οι επιλογείς είναι ένθετα μέσα στον επιλογέα πλοήγησης (nav).
Ενώ στο CSS, οι κανόνες ορίζονται έναν προς έναν (δεν είναι ένθετοι):

```
nav ul {
  margin: 0;
  padding: 0;
  list-style: none;
}
nav li {
  display: inline-block;
}
nav a {
  display: block;
  padding: 6px 12px;
  text-decoration: none;
}
```

Εικόνα 126– *Ένθετοι SASS (**Πηγή**: https://www.w3schools.com/sass/sass_nesting.php)*

Επειδή μπορείτε να μετατρέψετε τις ιδιότητες σε ένθετες στο SASS, είναι πιο εύκολα να διαβαστούν ό,τι οι τυπικές ιδιότητες CSS.

Ένθετες ιδιότητες SASS

Πολλές ιδιότητες CSS έχουν το ίδιο πρόθεμα, όπως font-family, font-size και font-weight ή text-align, text-transform και text-overflow.

Με το SASS μπορείτε να τις γράψετε ως ένθετες ιδιότητες:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
font: {
  family: Helvetica, sans-serif;
  size: 18px;
  weight: bold;
}

text: {
  align: center;
  transform: lowercase;
  overflow: hidden;
}
```

Εικόνα 127– *Ένθετοι SASS (**Πηγή**: https://www.w3schools.com/sass/sass_nesting.php)*

Ο αναμεταδότης SASS θα μετατρέψει τις παραπάνω ιδιότητες σε τυπικές ιδιότητες CSS:

```
font-family: Helvetica, sans-serif;
font-size: 18px;
font-weight: bold;

text-align: center;
text-transform: lowercase;
text-overflow: hidden;
```

Εικόνα 128– *Ένθετοι SASS (**Πηγή**: https://www.w3schools.com/sass/sass_nesting.php)*

SASS @import and Partials

Το Sass διατηρεί τον κώδικα CSS DRY (ακρώνυμο του «Don't Repat Yourself/ Να μην επαναληφθεί»). Ένας τρόπος για να γράψετε τον κώδικα DRY είναι να διατηρήσετε τον σχετικό κώδικα σε ξεχωριστά αρχεία.

Μπορείτε να δημιουργήσετε μικρά αρχεία με αποσπάσματα CSS (snippets) για να τα συμπεριλάβετε σε άλλα αρχεία SASS. Παραδείγματα τέτοιων αρχείων μπορεί να είναι: αρχείο επαναφοράς, μεταβλητές, χρώματα, γραμματοσειρές, μεγέθη γραμματοσειρών κ.λπ.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εισαγωγή αρχείων SASS

Όπως και το CSS, έτσι και το SASS υποστηρίζει επίσης την οδηγία @import.
Η οδηγία @import σας επιτρέπει να συμπεριλάβετε το περιεχόμενο ενός αρχείου σε ένα άλλο.
Η οδηγία CSS @import παρουσιάζει ένα σημαντικό μειονέκτημα λόγω κυρίως προβλημάτων που αφορούν την αποδοτικότητά τους. Αυτό γιατί κάθε φορά που την χρησιμοποιείτε δημιουργείται ένα επιπλέον αίτημα HTTP. Ωστόσο, η οδηγία SASS @import περιλαμβάνει το αρχείο στο CSS. Επομένως, δεν απαιτείται να χρησιμοποιήσετε ένα αίτημα http κατά την εκτέλεση!

```
@import filename;
```

Εικόνα 129– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Συμβουλή: Δεν χρειάζεται να καθορίσετε μια επέκταση αρχείου, εφόσον το SASS υποθέτει αυτόματα ότι εννοείτε ένα αρχείο .sass ή .scss. Μπορείτε επίσης να εισάγετε αρχεία CSS. Η οδηγία @import εισάγει το αρχείο και οποιεσδήποτε μεταβλητές ή mixins που ορίζονται στο εισαγόμενο αρχείο και μπορούν στη συνέχεια να χρησιμοποιηθούν στο κύριο αρχείο.

Μπορείτε να εισάγετε όσα αρχεία χρειάζεστε στο κύριο αρχείο:

```
@import "variables";
@import "colors";
@import "reset";
```

Εικόνα 130– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Ας δούμε ένα παράδειγμα: Ας υποθέσουμε ότι έχουμε ένα αρχείο επαναφοράς (reset file) που ονομάζεται "reset.scss", όπως φαίνεται πιο κάτω:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
html,
body,
ul,
ol {
  margin: 0;
  padding: 0;
}
```

Εικόνα 131– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Και τώρα θέλουμε να εισάγουμε το αρχείο "reset.scss" σε ένα άλλο αρχείο που ονομάζεται "standard.scss".

Να λοιπόν πώς μπορούμε να το κάνουμε: Είναι φυσιολογικό να προσθέτουμε την οδηγία @import στην κορυφή ενός αρχείου. Με αυτόν τον τρόπο, το περιεχόμενό της θα έχει ένα ευρύ πεδίο εφαρμογής:

```
@import "reset";

body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: red;
}
```

Εικόνα 132– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Έτσι, όταν το αρχείο "standard.css" μεταγλωττίζεται σε αρχείο CSS ο κώδικας θα πρέπει να ακολουθεί την πιο κάτω σύνταξη:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
html, body, ul, ol {
  margin: 0;
  padding: 0;
}

body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: red;
}
```

Εικόνα 133– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

SASS Partials

Από προεπιλογή, το SASS μεταγλωττίζει όλα τα αρχεία .scss άμεσα. Ωστόσο, όταν θέλετε να εισαγάγετε ένα αρχείο, δεν χρειάζεται το αρχείο να μεταφερθεί απευθείας.
Το SASS έχει ένα μηχανισμό για αυτό: Αν ξεκινήσετε γράφοντας το όνομα ενός αρχείου με μια κάτω παύλα, το SASS δεν θα μπορεί να το μεταγλωττίσει. Τα αρχεία που ονομάζονται με αυτόν τον τρόπο ονομάζονται «αποσπάσματα» (partials) στο Sass.

Έτσι, ένα SASS partial θα πρέπει να υπογραμμίζεται με μια κάτω παύλα:

```
_filename;
```

Εικόνα 134– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Το ακόλουθο παράδειγμα εμφανίζει ένα απόσπασμα αρχείου SASS που ονομάζεται "_colors.scss". (Αυτό το αρχείο δεν θα μεταγλωττιστεί απευθείας σε "colors.css"):

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
$myPink: #EE82EE;
$myBlue: #4169E1;
$myGreen: #8FBC8F;
```

Εικόνα 135– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Τώρα, αν εισάγετε το απόσπασμα αρχείου, παραλείψτε τον χαρακτήρα υπογράμμισης της κάτω παύλας. Το SASS κατανοεί ότι θα πρέπει να εισάγει το αρχείο "_colors.scss":

```
@import "colors";

body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: $myBlue;
}
```

Εικόνα 136– *Εισαγωγή SASS (**Πηγή**: https://www.w3schools.com/sass/sass_import.php)*

Οι εντολές @mixin και @include στο SASS

SASS Mixins

Η εντολή @mixin σας επιτρέπει να δημιουργήσετε έναν κώδικα CSS που πρόκειται να επαναχρησιμοποιηθεί σε όλη την ιστοσελίδα.
Η εντολή @include δημιουργήθηκε για να σας επιτρέπει να χρησιμοποιήσετε (include) το mixin.

Ορισμός ενός Mixin

Ένα mixin ορίζεται με την εντολή @mixin.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
@mixin name {
  property: value;
  property: value;
  ...
}
```

Εικόνα 137– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Στο ακόλουθο παράδειγμα έχουμε τη σύνταξη ενός mixin με το όνομα "important-text":

```
@mixin important-text {
  color: red;
  font-size: 25px;
  font-weight: bold;
  border: 1px solid blue;
}
```

Εικόνα 138– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Η χρήση ενός mixin

Η οδηγία @include χρησιμοποιείται για να συμπεριλάβει ένα mixin.

```
selector {
  @include mixin-name;
}
```

Εικόνα 139– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Έτσι λοιπόν, για να συμπεριλάβετε το mixin με το όνομα "important-text" που δημιουργήθηκε παραπάνω καταχωρήστε:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
.danger {
  @include important-text;
  background-color: green;
}
```

Εικόνα 140– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Ο αναμεταδότης SASS θα μεταγλωττίσει των κώδικα SASS σε τυπικό CSS:

```
.danger {
  color: red;
  font-size: 25px;
  font-weight: bold;
  border: 1px solid blue;
  background-color: green;
}
```

Εικόνα 141– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Ένα mixin μπορεί επίσης να περιλαμβάνει και άλλα mixins:

```
@mixin special-text {
  @include important-text;
  @include link;
  @include special-border;
}
```

Εικόνα 142– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Προσθήκη μεταβλητών σε ένα mixin

Τα mixins μπορούν να λάβουν οδηγίες-εντολές. Με αυτόν τον τρόπο μπορείτε να προσθέσετε μεταβλητές σε ένα mixin.

Παρατηρήστε το πιο κάτω παράδειγμα για να δείτε πώς ορίζεται ένα mixin με εντολές:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
/* Define mixin with two arguments */
@mixin bordered($color, $width) {
  border: $width solid $color;
}

.myArticle {
  @include bordered(blue, 1px);  // Call mixin with two values
}

.myNotes {
  @include bordered(red, 2px); // Call mixin with two values
}
```

Εικόνα 143– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Σημειώστε ότι οι εντολές ορίζονται ως μεταβλητές και στη συνέχεια χρησιμοποιούνται ως τιμές (χρώμα και πλάτος) της ιδιότητας των ορίων.

Μετά τη μεταγλώττιση, ο κώδικας CSS θα έχει την εξής μορφή:

```
.myArticle {
  border: 1px solid blue;
}

.myNotes {
  border: 2px solid red;
}
```

Εικόνα 144– *SASS @mixin (**Πηγή**: https://www.w3schools.com/sass/sass_mixin_include.php)*

Η εντολή @extension στο SASS

Η εντολή @extension σάς επιτρέπει να μοιράζεστε ένα σύνολο ιδιοτήτων CSS από τον ένα επιλογέα στον άλλο.
Η εντολή @extension είναι χρήσιμη εάν έχετε σχεδόν πανομοιότυπα στοιχεία που διαφέρουν μόνο σε ορισμένες μικρές λεπτομέρειες.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Το ακόλουθο παράδειγμα SASS δημιουργεί πρώτα ένα βασικό στυλ για κουμπιά (αυτό το στυλ θα χρησιμοποιηθεί για τα περισσότερα κουμπιά). Στη συνέχεια, δημιουργούμε ένα στυλ για ένα κουμπί “Report” και ένα στυλ για ένα κουμπί “Submit”. Τόσο το κουμπί “Report” όσο και το κουμπί “Submit” κληρονομούν όλες τις ιδιότητες CSS από την κλάση .button-basic, μέσω της οδηγίας @extension. Επιπλέον, έχουν τα δικά τους χρώματα που ορίζονται ως εξής:

```
.button-basic  {
  border: none;
  padding: 15px 30px;
  text-align: center;
  font-size: 16px;
  cursor: pointer;
}

.button-report  {
  @extend .button-basic;
  background-color: red;
}

.button-submit  {
  @extend .button-basic;
  background-color: green;
  color: white;
}
```

Εικόνα 145– *SASS Entend (**Πηγή**: https://www.w3schools.com/sass/sass_extend.php)*

Μετά τη μεταγλώττιση, ο κώδικας CSS θα έχει την εξής μορφή:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
.button-basic, .button-report, .button-submit {
  border: none;
  padding: 15px 30px;
  text-align: center;
  font-size: 16px;
  cursor: pointer;
}

.button-report  {
  background-color: red;
}

.button-submit  {
  background-color: green;
  color: white;
}
```

Εικόνα 146– *SASS Entend (**Πηγή**: https://www.w3schools.com/sass/sass_extend.php)*

Όταν χρησιμοποιείτε την εντολή @extension, δεν χρειάζεται να ορίζετε αρκετές κλάσεις για ένα στοιχείο στον κώδικα HTML, όπως την ετικέτα εδώ: <button class="button-basic button-report">Report this αυτού</button>. Απλά πρέπει να ορίσετε την ετικέτα .button-report. για να λάβετε και τα δύο σύνολα στυλ.

Η εντολή @extension σας επιτρέπει να διατηρείτε τον κώδικά SASS πολύ DRY.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

4. SQL - Structured Query Language

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πληροφορίες για την ενότητα

Ενότητα:

4. Γλώσσα SQL

Προαπαιτούμενες γνώσεις:

Βασικές γνώσεις υπολογιστών, εγκατεστημένο βασικό λογισμικό και βασικές γνώσεις εργασίας με αρχεία.

Φόρτος εργασίας:

10 ώρες

Περιγραφή:

Σε αυτή την ενότητα, θα καλύψουμε τα βασικά της SQL για να εξοικειώσουμε τους μαθητές με τον κόσμο του προγραμματισμού και να τους ενθαρρύνουμε να αποκτήσουν περισσότερη εμπειρία στην SQL. Εξηγούμε τα χαρακτηριστικά, τη σύνταξη και άλλους σχετικούς όρους που μπορεί να έχουν ακούσει ή να είναι εξοικειωμένοι οι εκπαιδευόμενοι και πώς αυτά ταιριάζουν στη γλώσσα προγραμματισμού. Παρέχουμε μια λεπτομερή περιγραφή της λογικής και της σύνταξης που χρησιμοποιούνται στην SQL, τη δομή της και άλλες βασικές και σημαντικές λειτουργίες.

Μαθησιακά αποτελέσματα:

- Αναγνώριση της έννοιας και της χρήσης της Structured Query Language (SQL) στα Relational Database Management Systems (RDBMS), ή Σχεσιακά Συστήματα Διαχείρισης Βάσεων Δεδομένων (ΣΣΔΒΔ)
- Διαμόρφωση βασικής σύνταξης ερωτημάτων SQL
- Χρήση της SQL για να της απόκτηση πρόσβασης, τη διαχείριση και τον χειρισμό των ΣΣΔΒΔ

Απαιτούμενα υλικά:

- Υπολογιστής ή φορητός υπολογιστής

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Σύνδεση στο Διαδίκτυο
- Διαδικτυακό πρόγραμμα μεταγλώττισης της SQL (https://www.mycompiler.io/new/sql)
- Διαδικτυακό πρόγραμμα επεξεργασίας κειμένου (https://www.w3schools.com/html/default.asp)

Σενάριο μαθήματος:

Ο συνολικός χρόνος που αφιερώνεται σε αυτή την ενότητα είναι 10 ώρες και εναπόκειται στην κρίση του εκπαιδευτικού να αποφασίσει πόσος χρόνος θα δαπανηθεί για κάθε υποενότητα. Προτείνουμε να χρησιμοποιήσετε τις παρουσιάσεις κατάρτισης PPT που δημιουργήθηκαν ρητά για αυτό το θέμα για να διευκολύνετε τη διαδικασία διδασκαλίας και να αυξήσετε την αποδοτικότητα του χρόνου. Αυτές οι παρουσιάσεις περιλαμβάνουν τα ακόλουθα:

- Προοδευτική ανάπτυξη επιμέρους θεμάτων και βασικών εννοιών για τη διατήρηση, και
- Προτεινόμενες ασκήσεις.

Ανάλογα με τις προτιμήσεις του εκπαιδευτικού, η προοδευτική ανάπτυξη των παρουσιάσεων επιτρέπει την ολοκλήρωση του μαθήματος πάνω στην SQL εντός του καθορισμένου χρόνου, δηλαδή 10 ώρες. Οι παρουσιάσεις μπορούν επίσης να διατεθούν στους εκπαιδευόμενους για αυτοδιδασκαλία.

Υποενότητες:

4.1. Τα βασικά της SQL

4.2. Βάσεις δεδομένων SQL

4.3. Παραπομπές SQL

4.4 Παραδείγματα SQL

Επιπλέον πόροι:

- W3Schools - Οδηγός για κάθε λέξη-κλειδί και λειτουργία SQL, και παραδείγματα για καθένα από αυτά

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Tutorialspoint – Ένας άλλος λεπτομερής οδηγός για τις λέξεις-κλειδιά και τις λειτουργίες SQL, και διάφορα παραδείγματα για κάθε μία από αυτές

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

4.1. Τα βασικά της SQL

Τι είναι η SQL;

Το ακρωνύμιο SQL σημαίνει Structured Query Language (Δομημένη Γλώσσα Ερωτημάτων). Η SQL είναι μια τυποποιημένη γλώσσα προγραμματισμού ειδικά σχεδιασμένη για την αποθήκευση, ανάκτηση, διαχείριση ή χειρισμό δεδομένων που βρίσκονται μέσα σε ένα Σχεσιακό Σύστημα Διαχείρισης Βάσεων Δεδομένων (ΣΣΔΒΔ). Μια σχεσιακή βάση δεδομένων είναι μια συλλογή στοιχείων δεδομένων με προκαθορισμένες σχέσεις μεταξύ τους. Τα στοιχεία αυτά οργανώνονται ως ένα σύνολο πινάκων με στήλες και σειρές.

Η SQL εντάχθηκε στο πρότυπο ISO το 1987.

Η SQL είναι η πιο ευρέως εφαρμοσμένη γλώσσα βάσης δεδομένων που υποστηρίζεται από δημοφιλή συστήματα σχεσιακών βάσεων δεδομένων, όπως MySQL, SQL Server, και Oracle. Η SQL αναπτύχθηκε αρχικά στην IBM στις αρχές της δεκαετίας του 1970. Αρχικά, ονομαζόταν SEQUEL (Structured English Query Language) και αργότερα άλλαξε σε SQL (προφέρεται ως S-Q-L).

Εφαρμογές της SQL

Η SQL είναι μία από τις πιο ευρέως χρησιμοποιούμενες γλώσσες ερωτημάτων για βάσεις δεδομένων. Μερικές από τις πολλές εφαρμογές της είναι:

- Να επιτρέπει στους χρήστες να έχουν πρόσβαση σε δεδομένα στα σχεσιακά Συστήματα Διαχείρισης Βάσεων Δεδομένων.
- Να επιτρέπει στους χρήστες να περιγράψουν τα δεδομένα.
- Να επιτρέπει στους χρήστες να ορίσουν τα δεδομένα σε μια βάση δεδομένων και να χειρίζονται τα δεδομένα αυτά.

Σύνταξη στην SQL

Οι εντολές στην SQL είναι απλές, σαν να γράφουμε προτάσεις, αλλά με συγκεκριμένη σύνταξη.

Μια εντολή στην SQL αποτελείται από μια ακολουθία λέξεων-κλειδιών, αναγνωριστικών κ.λπ., που τερματίζονται από ένα ερωτηματικό (;).

Παράδειγμα:
SELECT όνομα_εργαζομένου, ημερομηνία_πρόσληψης, μισθός FROM Εργαζόμενοι WHERE μισθός > 5000;

Για πιο εύκολη αναγνωσιμότητα, μπορείτε να γράψετε την ίδια εντολή ως εξής:

SELECT όνομα_εργαζομένου, ημερομηνία_πρόσληψης, μισθός
FROM Εργαζόμενοι
WHERE μισθός > 5000;

Χρησιμοποιήστε το ερωτηματικό του ελληνικού αλφαβήτου στο τέλος μιας εντολής SQL, καθώς τερματίζει τη εντολή ή υποβάλλει τις πληροφορίες στον διακομιστή βάσης δεδομένων.

Διάκριση πεζών-κεφαλαίων στην SQL

Εξετάστε μια άλλη εντολή στην SQL που ανακτά εγγραφές από τον πίνακα Εργαζομένων:
SELECT όνομα_εργαζομένου, ημερομηνία_πρόσληψης, μισθός FROM Εργαζόμενοι;

Η ίδια εντολή μπορεί επίσης να γραφτεί ως εξής:
select όνομα_εργαζομένου, ημερομηνία_πρόσληψης, μισθός from εργαζόμενοι;

Οι λέξεις-κλειδιά στην SQL είναι χωρίς διάκριση πεζών-κεφαλαίων, που σημαίνει ότι η εντολή SELECT είναι το ίδιο πράγμα με το select.

Ωστόσο, η βάση δεδομένων και τα ονόματα των πινάκων μπορεί να είναι, σύμφωνα με την περίπτωση, ανάλογα με το λειτουργικό σύστημα. Σε γενικές γραμμές, οι πλατφόρμες

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Unix ή Linux είναι ευαίσθητες στη διάκριση πεζών-κεφαλαίων, ενώ οι πλατφόρμες Windows όχι.

Επιλογή δεδομένων στην SQL (SELECT)

Η εντολή SELECT επιλέγει ή ανακτά δεδομένα από έναν ή περισσότερους πίνακες. Μπορείτε να χρησιμοποιήσετε αυτή την εντολή για να ανακτήσετε όλες τις σειρές από έναν πίνακα με μία κίνηση ή να ανακτήσετε μόνο εκείνες τις σειρές που ικανοποιούν μια συγκεκριμένη συνθήκη ή έναν συνδυασμό συνθηκών.

Ας υποθέσουμε ότι έχουμε έναν πίνακα με το όνομα Εργαζόμενοι στη βάση δεδομένων μας που περιέχει τις ακόλουθες εγγραφές:

```
| κωδικός_εργαζομένου| όνομα_εργαζομένου| ημερομηνία_πρόσληψης| μισθός| κωδικός_τμήματος|
+--------+--------------+------------+--------+---------+
|      1 | Ίθαν Χαντ    | 2001-05-01 |   5000 |       4 |
|      2 | Τόνι Μοντάνα | 2002-07-15 |   6500 |       1 |
|      3 | Σάρα Κόνορ | 2005-10-18 |   8000 |       5 |
|      4 | Ρικ Ντέκαρντ | 2007-01-03 |   7200 |       3 |
|      5 | Μάρτιν Μπλανκ | 2008-06-24 |   5600 |    NULL |
+--------+--------------+------------+--------+---------+
```

Επιλογή όλων από έναν πίνακα

Η ακόλουθη εντολή θα επιλέξει όλες τις σειρές από τον πίνακα των εργαζομένων.

```
>> SELECT * FROM Εργαζόμενοι;
```

Επιλογή συγκεκριμένων στηλών από τον πίνακα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εάν δεν χρειάζεστε όλα τα δεδομένα, μπορείτε να επιλέξετε συγκεκριμένες στήλες, όπως φαίνεται παρακάτω:

```
SELECT κωδικός_εργαζομένου, όνομα_εργαζομένου, ημερομηνία_πρόσληψης, μισθός
FROM Εργαζόμενοι;
```

Μετά την εκτέλεση της παραπάνω εντολής, θα λάβετε μια έξοδο όπως αυτή:

```
| κωδικός_εργαζομένου| όνομα_εργαζομένου| ημερομηνία_πρόσληψης| μισθός|
+--------+--------------+------------+--------+
|      1 | Ίθαν Χαντ | 1995-10-30 |   5000 |
|      2 | Τόνι Μοντάνα | 1990-07-15 |   6500 |
|      3 | Σάρα Κόνορ | 2011-04-13 |   5600 |
|      4 | Ρικ Ντέκαρντ | 2005-10-18 |   7200 |
|      5 | Μάρτιν Μπλανκ | 1996-05-24 |   8000 |
+--------+--------------+------------+--------+
```

Επιλογή ορισμένων δεδομένων στην SQL (SELECT DISTINCT)

Η εντολή SELECT DISTINCT παραλείπει τις διπλές τιμές όταν χρησιμοποιείται σε ένα ερώτημα.

Μπορεί να βρείτε διπλές τιμές μέσα σε έναν πίνακα, αλλά μερικές φορές μπορεί να θέλετε να δείτε τις «μοναδικές» τιμές.

Σύνταξη:

```
SELECT DISTINCT στήλη1, στήλη2, ...
FROM όνομα_πίνακα;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στα παρακάτω παραδείγματα, θα χρησιμοποιήσουμε τον πίνακα Πελατών που περιέχει δεδομένα σχετικά με τους πελάτες μας.

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Πίνακας 1– *Παράδειγμα πίνακα πελατών της εντολής SELECT DISTINCT*

(***Πηγή***: https://www.w3schools.com/sql/sql_distinct.asp)

Για να *επιλέξετε όλες τις τιμές* από τη στήλη Χώρα στον πίνακα Πελάτες, θα χρησιμοποιήσετε την ακόλουθη εντολή:

```
SELECT Χώρα FROM Πελάτες;
```

Ωστόσο, η εντολή αυτή θα περιλαμβάνει διπλές τιμές.

Εάν θέλετε να παραλείψετε τις διπλότυπες τιμές από το ερώτημά σας, χρησιμοποιήστε την SELECT DISTINCT:

```
SELECT DISTINCT Χώρα FROM Πελάτες;
```

Ας υποθέσουμε ότι θέλετε να απαριθμήσετε τον αριθμό των διαφορετικών χωρών των πελατών. Θα χρησιμοποιήσετε την ακόλουθη εντολή:

```
SELECT COUNT(DISTINCT Χώρα) FROM Πελάτες;
```

Λάβετε υπόψη ότι αυτό το παράδειγμα δεν θα λειτουργήσει στο Firefox, καθώς το COUNT(DISTINCT όνομα_στήλης) δεν υποστηρίζεται στο MS Access.

Για να λάβετε το ισοδύναμο αποτέλεσμα στο σύστημα Access, χρησιμοποιήστε αυτό:

```
>> SELECT Count(*) AS DistinctCountries
   FROM (SELECT DISTINCT Χώρα FROM Πελάτες);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Όρος WHERE στην SQL

Προηγουμένως, μάθαμε πώς να επιλέγουμε όλα τα αρχεία από έναν πίνακα ή στήλες ενός πίνακα.

Ωστόσο, σε πραγματικές περιπτώσεις, πρέπει γενικά να επιλέξουμε, να ενημερώσουμε ή να διαγράψουμε μόνο εκείνα τα αρχεία που ικανοποιούν καθορισμένες συνθήκες, όπως οι χρήστες που ανήκουν σε μια συγκεκριμένη ηλικιακή ομάδα, χώρα κ.λπ.

Ο όρος WHERE χρησιμοποιείται με τις εντολές SELECT, UPDATE, και DELETE.

Ο όρος WHERE χρησιμοποιείται με την εντολή SELECT για την εξαγωγή μόνο εκείνων των εγγραφών που ικανοποιούν καθορισμένες συνθήκες.

Η βασική σύνταξη είναι η ακόλουθη:

```
SELECT στήλη_λίστα
FROM όνομα_πίνακα
WHERE συνθήκη;
```

Τώρα, ας δούμε μερικά παραδείγματα που δείχνουν πώς λειτουργεί.

Ας υποθέσουμε ότι έχουμε έναν πίνακα που ονομάζεται Εργαζόμενοι στη βάση δεδομένων μας με τα ακόλουθα αρχεία:

```
| κωδικός_εργαζομένου| όνομα_εργαζομένου| ημερομηνία_πρόσληψης| μισθός| κωδικός_τμήματος|
+--------+--------------+------------+--------+---------+
|      1 | Ίθαν Χαντ    | 2001-05-01 |   5000 |       4 |
|      2 | Τόνι Μοντάνα | 2002-07-15 |   6500 |       1 |
|      3 | Σάρα Κόνορ | 2005-10-18 |   8000 |       5 |
|      4 | Ρικ Ντέκαρντ | 2007-01-03 |   7200 |       3 |
+--------+--------------+------------+--------+---------+
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η ακόλουθη εντολή της SQL θα επιλέξει όλους τους υπαλλήλους από τον πίνακα *των Εργαζομένων των οποίων ο μισθός είναι μεγαλύτερος από 7000*:

```
SELECT * FROM Εργαζόμενοι WHERE μισθός > 7000;
```

Ο όρος WHERE απλώς φιλτράρει τα ανεπιθύμητα δεδομένα.

Ένα άλλο παράδειγμα θα ήταν να *επιλέξετε όλους* τους υπαλλήλους με *κωδικό τμήματος =1*:

```
SELECT * FROM Εργαζόμενοι WHERE κωδικό_τμήματος=1;
```

Ο ακόλουθος πίνακας παρουσιάζει τους τελεστές που μπορούν να χρησιμοποιηθούν με τον όρο WHERE:

Operator	Description
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal. **Note:** In some versions of SQL this operator may be written as !=
BETWEEN	Between a certain range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column

Πίνακας 2– *Πίνακας των τελεστών που χρησιμοποιούνται με τον όρο WHERE*
(***Πηγή***: https://www.w3schools.com/sql/sql_where.asp)

Οι τελεστές AND, OR και NOT στην SQL

Ο όρος WHERE μπορεί να συνδυαστεί με τους τελεστές AND, OR, και NOT.

Οι τελεστές AND και OR χρησιμοποιούνται για το φιλτράρισμα εγγραφών με βάση περισσότερες από μία συνθήκες.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο τελεστής AND εμφανίζει μια εγγραφή εάν όλες οι συνθήκες που χρησιμοποιούν AND είναι TRUE.

Syntax AND:

```
SELECT στήλη1, στήλη2, ...
FROM όνομα_πίνακα
WHERE συνθήκη1 AND συνθήκη2 AND συνθήκη3...;
```

Παράδειγμα: Επιλέξτε όλα τα πεδία από τον πίνακα πελατών όπου η χώρα είναι η «Γερμανία» ΚΑΙ η πόλη είναι το «Βερολίνο».

```
SELECT * FROM Πελάτες
WHERE Χώρα='Γερμανία' AND Πόλη='Βερολίνο';
```

Ο τελεστής OR εμφανίζει μια εγγραφή εάν οποιαδήποτε από τις συνθήκες που χρησιμοποιούν το OR είναι TRUE.

Σύνταξη OR:

```
SELECT στήλη1, στήλη2, ...
FROM όνομα_πίνακα
WHERE συνθήκη1 OR συνθήκη2 OR συνθήκη3...;
```

Παράδειγμα 1: Επιλέξτε όλα τα πεδία από τον πίνακα πελατών όπου η πόλη είναι το «Βερολίνο» ή το «Μόναχο»

```
SELECT * FROM Πελάτες
WHERE Πόλη = 'Βερολίνο' OR Πόλη = 'Μόναχο';
```

Παράδειγμα 2: Επιλέξτε όλα τα πεδία από τον πίνακα πελατών όπου η χώρα είναι «Γερμανία» ή «Ισπανία».

```
SELECT * FROM Πελάτες
WHERE Χώρα='Γερμανία' OR Χώρα='Ισπανία';
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο τελεστής NOT εμφανίζει μια εγγραφή εάν οι συνθήκες είναι NOT TRUE.

Σύνταξη NOT:

```
SELECT στήλη1, στήλη2, ...
FROM όνομα_πίνακα
WHERE NOT συνθήκη;
```

Παράδειγμα: Επιλέξτε όλα τα πεδία από τον πίνακα πελατών όπου η χώρα ΔΕΝ είναι η «Γερμανία».

```
SELECT * FROM Πελάτες
WHERE NOT Χώρα='Γερμανία';
```

Συνδυασμός των AND, OR και NOT

Παράδειγμα 1: Επιλέξτε όλες τις σειρές από τον πίνακα Πελάτες όπου η χώρα είναι η Γερμανία και η πόλη πρέπει να είναι είτε το Βερολίνο είτε το Μόναχο.

```
SELECT * FROM Πελάτες
WHERE Χώρα='Γερμανία' AND (Πόλη='Βερολίνο' OR Πόλη='Μόναχο');
```

Παράδειγμα 2: Επιλέξτε όλες τις σειρές από τον πίνακα Πελάτες όπου η χώρα ΔΕΝ είναι η Γερμανία και ούτε οι ΗΠΑ.

```
SELECT * FROM Πελάτες
WHERE NOT Χώρα='Γερμανία' AND NOT Χώρα='ΗΠΑ';
```

Ταξινόμηση στην SQL (ORDER BY)

Η λέξη-κλειδί ORDER BY ταξινομεί το σύνολο των αποτελεσμάτων κατά αύξουσα ή φθίνουσα σειρά. Η λέξη-κλειδί ORDER BY ταξινομεί τις εγγραφές με αύξουσα σειρά από

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

προεπιλογή. Για να ταξινομήσετε τις εγγραφές με φθίνουσα σειρά, χρησιμοποιήστε τη λέξη-κλειδί DESC.

Σύνταξη ORDER BY:

```
SELECT στήλη1, στήλη, ...
FROM όνομα_πίνακα
ORDER BY στήλη1, στήλη2, ... ASC|DESC;
```

Στα παρακάτω παραδείγματα, θα χρησιμοποιήσουμε τις ελληνικές αντιστοιχίες από τον παρακάτω πίνακα Πελατών:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

***Πίνακας 3**– Πίνακας Πελατών με παράδειγμα ORDER BY*

*(**Πηγή**: https://www.w3schools.com/sql/sql_orderby.asp)*

Παράδειγμα 1: Επιλέγει όλους τους πελάτες από τον πίνακα Πελατών και τους ταξινομεί κατά τη στήλη Χώρα

```
SELECT * FROM Πελάτες
ORDER BY Χώρα;
```

Παράδειγμα 2: Επιλέγει όλους τους πελάτες από τον ίδιο πίνακα και τους ταξινομεί με φθίνουσα σειρά κατά τη στήλη Χώρα

```
SELECT * FROM Πελάτες
ORDER BY Χώρα DESC;
```

Παράδειγμα 3: Επιλέγει όλους τους πελάτες από τον ίδιο πίνακα και τους ταξινομεί ανά χώρα και όνομα πελάτη.

```
SELECT * FROM Πελάτες
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
ORDER BY Χώρα, ΌνομαΠελάτη;
```

Εδώ, η ταξινόμηση γίνεται αρχικά ανά χώρα. Ωστόσο, εάν υπάρχουν ορισμένες σειρές που έχουν την ίδια χώρα, τότε ταξινομούνται ανά Όνομα Πελάτη.

Παράδειγμα 4: Επιλέγει όλους τους πελάτες από τον ίδιο πίνακα και τους ταξινομεί κατά αύξουσα σειρά ανά χώρα και κατά φθίνουσα σειρά ανά όνομα πελάτη

```
SELECT * FROM Πελάτες
ORDER BY Χώρα ASC, ΌνομαΠελάτη DESC;
```

Εισαγωγή Δεδομένων στην SQL (INSERT INTO)

Η εντολή INSERT INTO εισάγει νέες εγγραφές σε έναν πίνακα.

Για να τρέξει σωστά ο κώδικάς σας, καθορίστε τα ονόματα των στηλών και τις τιμές που θα εισαχθούν.

Σύνταξη:

```
INSERT INTO όνομα_πίνακα (στήλη1, στήλη2, στήλη3, ...)
VALUES (τιμή1, τιμή2, τιμή3, ...);
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland

Πίνακας 4– *INSERT INTO*
(***Πηγή***: https://www.w3schools.com/sql/sql_insert.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για παράδειγμα, για να προσθέσετε μια νέα εγγραφή στον πίνακα «Πελάτες», χρησιμοποιήστε τα εξής:

```
INSERT INTO Πελάτες (ΌνομαΠελάτη, ΌνομαΕπικοινωνίας, Διεύθυνση, Πόλη, ΤαχυδρομικόςΚώδικας, Χώρα)
VALUES (‘Κάρντιναλ', ‘Τομ Μπ. Έριχσεν', ‘Σκάγκεν 21', ‘Στάβανγκερ', '4006', ‘Νορβηγία');
```

Κενές τιμές στην SQL (NULL)

Ένα πεδίο με τιμή NULL είναι **ένα πεδίο χωρίς τιμή**. Εάν ένα πεδίο σε έναν πίνακα είναι προαιρετικό, είναι δυνατό να εισαχθεί μια νέα εγγραφή ή να ενημερωθεί μια εγγραφή χωρίς να προστεθεί μια τιμή σε αυτό το πεδίο. Στη συνέχεια, το πεδίο θα αποθηκευτεί με τιμή NULL.

Σημείωση: Μια τιμή NULL διαφέρει από μια μηδενική τιμή ή ένα πεδίο που περιέχει κενά. Ένα πεδίο με τιμή NULL έχει **μείνει κενό** κατά τη δημιουργία εγγραφών!

Έλεγχος για ΤΙΜΕΣ NULL:

Είναι αδύνατος ο έλεγχος για τιμές NULL με τελεστές σύγκρισης, όπως =, <, ή <>.

Αντ’ αυτού, θα πρέπει να χρησιμοποιήσουμε τους τελεστές **IS NULL** και **IS NOT NULL.**

Σύνταξη IS NULL:

```
SELECT όνομα_στήλης
FROM όνομα_πίνακα
WHERE όνομα_στήλης IS NULL;
```

Σύνταξη IS NOT NULL:

```
SELECT όνομα_στήλης
FROM όνομα_πίνακα
WHERE όνομα_στήλης IS NOT NULL;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τα ακόλουθα παραδείγματα χρησιμοποιούν τον ακόλουθο πίνακα:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Πίνακας 5– *Πίνακας Πελατών με τιμές NULL*

(***Πηγή***: *https://www.w3schools.com/sql/sql_null_values.asp*)

Παράδειγμα του IS NULL

Επιλέγει όλους τους πελάτες με τιμές NULL στη στήλη Διεύθυνση

```
SELECT ΌνομαΠελάτη, ΌνομαΕπικοινωνίας, Διεύθυνση
FROM Πελάτες
WHERE Διεύθυνση IS NULL;
```

Για να αναζητήσετε κενές τιμές, χρησιμοποιείτε πάντα το IS NULL.

Παράδειγμα του IS NOT NULL

Επιλέγει όλους τους πελάτες με τιμές NOT NULL στη στήλη Διεύθυνση

```
SELECT ΌνομαΠελάτη, ΌνομαΕπικοινωνίας, Διεύθυνση
FROM Πελάτες
WHERE Διεύθυνση IS NOT NULL;
```

Ενημέρωση δεδομένων στην SQL (UPDATE)

Η εντολή UPDATE τροποποιεί τις υπάρχουσες εγγραφές ενός πίνακα.

Σύνταξη:

```
UPDATE όνομα_πίνακα
SET στήλη1 = τιμή1, στήλη2 = τιμή2, ...
WHERE συνθήκη;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Να είστε προσεκτικοί όταν ενημερώνετε εγγραφές σε έναν πίνακα! Παρατηρήστε τον όρο WHERE στην εντολή UPDATE. Ο όρος WHERE προσδιορίζει ποια εγγραφή(-ές) πρέπει να ενημερωθεί (-ούν).

Εάν **παραλείψετε** τον όρο «WHERE», θα ενημερωθούν όλες οι εγγραφές στον πίνακα!

Παράδειγμα

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Πίνακας 6– *Παράδειγμα της εντολής UPDATE*

(***Πηγή***: https://www.w3schools.com/sql/sql_update.asp)

Για να ενημερώσετε τον ΚωδικόΠελάτη=1 από τον πίνακα «Πελάτες» στη δειγματική βάση δεδομένων, χρησιμοποιήστε τα ακόλουθα:

```
UPDATE Πελάτες
SET ΌνομαΕπικοινωνίας= Άλφρεντ Σμιντ', Πόλη='Φρανκφούρτη'
WHERE ΚωδικόςΠελάτη=1;
```

Και ο πίνακας «Πελάτες» θα έχει τώρα την εξής μορφή:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Alfred Schmidt	Obere Str. 57	Frankfurt	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Πίνακας 7– *Πίνακας Πελατών με παράδειγμα της εντολής UPDATE*

(***Πηγή***: https://www.w3schools.com/sql/sql_update.asp)

Ο όρος WHERE καθορίζει πόσες εγγραφές θα ενημερωθούν.

Παράδειγμα: Ενημέρωση της στήλης ΌνομαΕπικοινωνίας σε «Χουάν» για όλες τις εγγραφές όπου η χώρα είναι «Μεξικό» στον πίνακα Πελατών.

```
UPDATE Πελάτες
SET ΌνομαΕπικοινωνίας='Χουάν'
WHERE Χώρα='Μεξικό';
```

Διαγραφή δεδομένων στην SQL (DELETE)

Η εντολή DELETE διαγράφει υπάρχουσες εγγραφές σε έναν πίνακα.
DELETE FROM table_name WHERE condition;

Λάβετε υπόψη ότι πρέπει να είστε προσεκτικοί όταν διαγράφετε καταχωρήσεις σε έναν πίνακα! Παρατηρήστε τον όρο WHERE στην εντολή DELETE. Ο όρος WHERE προσδιορίζει ποια εγγραφή(-ές) πρέπει να διαγραφεί(-ούν).

Εάν **παραλείψετε** τον όρο «WHERE», θα διαγραφούν όλες οι εγγραφές του πίνακα!

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παράδειγμα:

```
DELETE FROM Πελάτες WHERE ΌνομαΠελάτη='Αλφρεντς Φούτερκιστε';
```

Διαγραφή όλων των εγγραφών ενός πίνακα

Είναι δυνατή η **διαγραφή όλων των σειρών σε έναν πίνακα χωρίς να διαγραφεί ο ίδιος ο πίνακας**. Αυτό σημαίνει ότι η δομή, τα χαρακτηριστικά και οι δείκτες του πίνακα θα παραμείνουν άθικτα:

```
DELETE FROM όνομα_πίνακα;
```

Επιλογή συγκεκριμένου αριθμού δεδομένων στην SQL (SELECT TOP)

Ο όρος SELECT TOP χρησιμοποιείται για να καθορίσει τον αριθμό των εγγραφών που θα επιλεχθούν.

Ο όρος SELECT TOP είναι χρήσιμος σε μεγάλους πίνακες με χιλιάδες εγγραφές. Ωστόσο, η επιλογή μεγάλου αριθμού εγγραφών μπορεί να επηρεάσει την απόδοση.

Σημειώστε ότι δεν υποστηρίζουν όλα τα συστήματα βάσεων δεδομένων τον όρο SELECT TOP. Το MySQL υποστηρίζει τον όρο LIMIT για την επιλογή ενός περιορισμένου αριθμού εγγραφών, ενώ το Oracle χρησιμοποιεί τους όρους FETCH FIRST αριθμός ROWS ONLY και ROWNUM.

Σύνταξη SQL Server/MS Access:

```
SELECT TOP αριθμός|ποσοστό όνομα(τα)_στήλης(ων)
FROM όνομα_πίνακα
WHERE συνθήκη;
```

Σύνταξη MySQL:

```
SELECT όνομα(τα)_στήλης(ων)
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
FROM όνομα_πίνακα
WHERE συνθήκη
LIMIT αριθμός;
LIMIT number;
```

Σύνταξη Oracle 12:

```
SELECT όνομα(τα)_στήλης(ων)
FROM όνομα_πίνακα
ORDER BY όνομα(τα)_στήλης(ων)
FETCH FIRST αριθμός ROWS ONLY;
```

Σύνταξη σε παλαιότερη έκδοση του Oracle:

```
SELECT όνομα_στήλης(ών)
FROM όνομα_πίνακα
WHERE ROWNUM <= αριθμός;
```

Σύνταξη σε παλαιότερη έκδοση του Oracle με την εντολή ORDER BY:

```
SELECT *
FROM (SELECT όνομα_στήλης(ών) FROM όνομα_πίνακα ORDER BY όνομα_στήλης(ών))
WHERE ROWNUM <=αριθμός;
```

Θα χρησιμοποιήσουμε τις ελληνικές αντιστοιχίες από τον πίνακα «Πελάτες» που παρουσιάζεται παρακάτω στα ακόλουθα παραδείγματα.

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

***Πίνακας 8**– Πίνακας Πελατών με παραδείγματα της εντολής SELECT TOP*

*(**Πηγή**: https://www.w3schools.com/sql/sql_top.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παραδείγματα TOP, LIMIT και FETCH FIRST

Για να επιλέξετε τις τρεις πρώτες εγγραφές από τον πίνακα Πελατών στο SQL Server/MS Access, χρησιμοποιήστε τα παρακάτω:

```
SELECT TOP 3 * FROM Πελάτες;
```

Για να εκτελέσετε ένα ερώτημα με το ίδιο αποτέλεσμα όπως παραπάνω στο MySQL, χρησιμοποιήστε την ακόλουθη εντολή:

```
SELECT * FROM Πελάτες
LIMIT 3;
```

Για να εκτελέσετε ένα ερώτημα με το ίδιο αποτέλεσμα με τα δύο παραπάνω στο Oracle:

```
SELECT * FROM Πελάτες
FETCH FIRST 3 ROWS ONLY;
```

Παραδείγματα Ανώτατου Ποσοστού (TOP PERCENT)

Για να επιλέξετε το πρώτο 50% των εγγραφών που βρίσκονται στον πίνακα Πελάτες, εκτελέστε την ακόλουθη εντολή στο SQL Server/MS Access:

```
SELECT TOP 50 PERCENT * FROM Πελάτες;
```

Το αντίστοιχό της στο Oracle είναι το ακόλουθο:

```
SELECT * FROM Πελάτες
FETCH FIRST 50 PERCENT ROWS ONLY;
```

Παραδείγματα προσθήκης του όρου WHERE

Στο παρακάτω παράδειγμα, επιλέγουμε τις τρεις πρώτες εγγραφές από τον πίνακα Πελατών, όπου η Χώρα είναι η «Γερμανία» σε SQL Server/MS Access:

```
>> SELECT TOP 3 * FROM Πελάτες
 WHERE Χώρα='Γερμανία';
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Το αντίστοιχό του στο MySQL:

```
SELECT * FROM Πελάτες
WHERE Χώρα='Γερμανία'
LIMIT 3;
```

Το αντίστοιχό του στο Oracle:

```
SELECT * FROM Πελάτες
WHERE Χώρα='Γερμανία'
FETCH FIRST 3 ROWS ONLY;
```

Συναρτήσεις Min και Max στην SQL

Η συνάρτηση MIN() επιλέγει **τις μικρότερες τιμές** από τις επιλεγμένες στήλες.

Σύνταξη της συνάρτησης MIN()

```
SELECT MIN(όνομα_στήλης)
FROM όνομα_πίνακα
WHERE συνθήκη;
```

Η συνάρτηση MAX() επιλέγει **τη μεγαλύτερη τιμή** από τις επιλεγμένες στήλες.

Σύνταξη της συνάρτησης MAX()

```
SELECT MAX(όνομα_στήλης)
FROM όνομα_πίνακα
WHERE συνθήκη;
```

Στα παρακάτω παραδείγματα, θα χρησιμοποιήσουμε τον παρακάτω πίνακα με τα Προϊόντα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
1	Chais	1	1	10 boxes x 20 bags	18
2	Chang	1	1	24 - 12 oz bottles	19
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22
5	Chef Anton's Gumbo Mix	2	2	36 boxes	21.35

Πίνακας 9– *Πίνακας προϊόντων με παραδείγματα των συναρτήσεων MIN και MAX*

*(**Πηγή**: https://www.w3schools.com/sql/sql_min_max.asp)*

Παράδειγμα 1: Βρίσκει την τιμή του φθηνότερου προϊόντος

```
SELECT MIN(Τιμή) AS ΦθηνότερηΤιμή
FROM Προϊόντα;
```

Παράδειγμα 2: Βρίσκει την τιμή του ακριβότερου προϊόντος

```
SELECT MAX(Τίμη) AS ΑκριβότερηΤιμή
FROM Προϊόντα;
```

Συναρτήσεις Count, Avg, Sum στην SQL

Η συνάρτηση COUNT() επιλέγει τον αριθμό των σειρών που ταιριάζουν σε ένα καθορισμένο κριτήριο.

Σύνταξη συνάρτησης COUNT()

```
SELECT COUNT(όνομα_στήλης)
FROM όνομα_πίνακα
WHERE συνθήκη;
```

Η συνάρτηση AVG() επιλέγει τη μέση τιμή μιας αριθμητικής στήλης.

Σύνταξη συνάρτησης AVG()

```
SELECT AVG(όνομα_στήλης)
FROM όνομα_πίνακα
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
WHERE συνθήκη;
```

Η συνάρτηση SUM() επιστρέφει το συνολικό άθροισμα μιας αριθμητικής στήλης.

Σύνταξη συνάρτησης SUM()

```
SELECT SUM(όνομα_στήλης)
FROM όνομα_πίνακα
WHERE συνθήκη;
```

Θα χρησιμοποιήσουμε τον πίνακα Προϊόντα στα παρακάτω παραδείγματα.

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
1	Chais	1	1	10 boxes x 20 bags	18
2	Chang	1	1	24 - 12 oz bottles	19
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22
5	Chef Anton's Gumbo Mix	2	2	36 boxes	21.35

***Πίνακας 10**– Πίνακας προϊόντων με παραδείγματα των συναρτήσεων COUNT, AVG, και SUM*

*(**Πηγή**: https://www.w3schools.com/sql/sql_count_avg_sum.asp)*

Παράδειγμα συνάρτησης COUNT()

Εκτελέστε ένα ερώτημα για να βρείτε τον αριθμό των προϊόντων:

```
SELECT COUNT(ΚωδικόςΠροϊόντων)
FROM Προϊόντα;
```

Παράδειγμα συνάρτησης AVG()

Εκτελέστε ένα ερώτημα για να βρείτε τη μέση τιμή όλων των προϊόντων:

```
SELECT AVG(Τιμή)
FROM Προϊόντα;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

OrderDetailID	OrderID	ProductID	Quantity
1	10248	11	12
2	10248	42	10
3	10248	72	5
4	10249	14	9
5	10249	51	40

***Πίνακας 11**– Πίνακας με πληροφορίες παραγγελιών με παραδείγματα των συναρτήσεων COUNT, AVG, και SUM (**Πηγή**: https://www.w3schools.com/sql/sql_count_avg_sum.asp)*

Στο παρακάτω παράδειγμα, θα χρησιμοποιήσουμε τον πίνακα ΛεπτομέρειεςΠαραγγελιών, που φαίνεται παραπάνω, για να βρούμε το άθροισμα της «Ποσότητας»:

```
SELECT SUM(Ποσότητα)
FROM ΛεπτομέρειεςΠαραγγελιών;
```

Τελεστής LIKE στην SQL

Ο τελεστής LIKE χρησιμοποιείται σε ένα όρο WHERE για να αναζητήσει ένα καθορισμένο μοτίβο σε μια στήλη.

Σύνταξη του τελεστή LIKE

```
SELECT στήλη1, στήλη2, ...
FROM όνομα_πίνακα
WHERE στήλη LIKE μοτίβο;
```

Ακολουθούν μερικά παραδείγματα που δείχνουν διαφορετικούς τελεστές LIKE με χαρακτήρες μπαλαντέρ '%' και '_':

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

LIKE Operator	Description
WHERE CustomerName LIKE 'a%'	Finds any values that start with "a"
WHERE CustomerName LIKE '%a'	Finds any values that end with "a"
WHERE CustomerName LIKE '%or%'	Finds any values that have "or" in any position
WHERE CustomerName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CustomerName LIKE 'a_%'	Finds any values that start with "a" and are at least 2 characters in length
WHERE CustomerName LIKE 'a__%'	Finds any values that start with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

Πίνακας 12– *Πίνακας τελεστών LIKE (**Πηγή**: https://www.w3schools.com/sql/sql_like.asp)*

Θα δούμε μερικά παραδείγματα που θα χρησιμοποιούν τις ελληνικές αντιστοιχίες από τον παρακάτω πίνακα Πελατών.

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Πίνακας 13– *Πίνακας πελατών με παραδείγματα του τελεστή LIKE*
*(**Πηγή**: https://www.w3schools.com/sql/sql_like.asp)*

Παράδειγμα 1: Επιλογή όλωω των πελατών με Όνομα Πελάτη που αρχίζει με ‘α’:

```
SELECT * FROM Πελάτες
WHERE ΌνομαΠελάτη LIKE 'α%';
```

Παράδειγμα 2: Επιλογή όλων των ονομάτων πελατών που τελειώνουν με «α»:

```
SELECT * FROM Πελάτες
WHERE ΌνομαΠελάτη LIKE '%α';
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παράδειγμα 3: Επιλογή όλων των ονομάτων πελατών που περιέχουν ‘ή’ στο όνομά τους σε οποιαδήποτε θέση.

```
SELECT * FROM Πελάτες
WHERE Όνομα Πελάτη LIKE '%ή%';
```

Παράδειγμα 4: Επιλογή όλων των ονομάτων πελατών που περιέχουν ‘ρ’ στη δεύτερη θέση:

```
SELECT * FROM Πελάτες
WHERE ΌνομαΠελάτη LIKE '_ρ%';
```

Παράδειγμα 5: Επιλογή όλων των ονομάτων πελατών που αρχίζουν με ‘α’ και έχουν τουλάχιστον τρεις χαρακτήρες στο σύνολο

```
SELECT * FROM Πελάτες
WHERE ΌνομαΠελάτη LIKE 'α__%';
```

Παράδειγμα 6: Επιλογή όλων των ονομάτων πελατών που αρχίζουν με ‘α’ και τελειώνουν με ‘ο’ :

```
SELECT * FROM Πελάτες
WHERE ΌνομαΠελάτη LIKE 'α%ο';
```

Παράδειγμα 7: Επιλογή όλων των ονομάτων πελατών που δεν αρχίζουν με ‘α’.

```
SELECT * FROM Πελάτες
WHERE ΌνομαΠελάτη LIKE 'α%';
```

Χαρακτήρες Μπαλαντέρ στην SQL

Ένας χαρακτήρας μπαλαντέρ αντικαθιστά έναν ή περισσότερους χαρακτήρες σε μια συμβολοσειρά. Χρησιμοποιείται με τον τελεστή LIKE.

Ο τελεστής LIKE χρησιμοποιείται επίσης σε έναν όρο WHERE για να αναζητήσει ένα συγκεκριμένο μοτίβο σε μια στήλη, όπως έχουμε δει στην προηγούμενη υποενότητα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Χαρακτήρες Μπαλαντέρ στο MS Access

Symbol	Description	Example
*	Represents zero or more characters	bl* finds bl, black, blue, and blob
?	Represents a single character	h?t finds hot, hat, and hit
[]	Represents any single character within the brackets	h[oa]t finds hot and hat, but not hit
!	Represents any character not in the brackets	h[!oa]t finds hit, but not hot and hat
-	Represents any single character within the specified range	c[a-b]t finds cat and cbt
#	Represents any single numeric character	2#5 finds 205, 215, 225, 235, 245, 255, 265, 275, 285, and 295

Πίνακας 14– *Χαρακτήρες Μπαλαντέρ στο MS Access (**Πηγή**: https://www.w3schools.com/sql/sql_wildcards.asp)*

Χαρακτήρες Μπαλαντέρ στον διακομιστή SQL

Symbol	Description	Example
%	Represents zero or more characters	bl% finds bl, black, blue, and blob
_	Represents a single character	h_t finds hot, hat, and hit
[]	Represents any single character within the brackets	h[oa]t finds hot and hat, but not hit
^	Represents any character not in the brackets	h[^oa]t finds hit, but not hot and hat
-	Represents any single character within the specified range	c[a-b]t finds cat and cbt

Πίνακας 15– *Χαρακτήρες Μπαλαντέρ στον διακομιστή SQL*

*(**Πηγή**: https://www.w3schools.com/sql/sql_wildcards.asp)*

Οι χαρακτήρες μπαλαντέρ μπορούν να χρησιμοποιηθούν συνδυαστικά. Δείτε μερικά παραδείγματα στον παρακάτω πίνακα:

LIKE Operator	Description
WHERE CustomerName LIKE 'a%'	Finds any values that starts with "a"
WHERE CustomerName LIKE '%a'	Finds any values that ends with "a"
WHERE CustomerName LIKE '%or%'	Finds any values that have "or" in any position
WHERE CustomerName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CustomerName LIKE 'a__%'	Finds any values that starts with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that starts with "a" and ends with "o"

Πίνακας 16– *Παραδείγματα χαρακτήρων μπαλαντέρ με % και '_'*

*(**Πηγή**: https://www.w3schools.com/sql/sql_wildcards.asp)*

Ας δούμε μερικά παραδείγματα χρησιμοποιώντας τον πίνακα Πελατών.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	Martín Sommer	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app'	Laurence Lebihans	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	Elizabeth Lincoln	23 Tsawassen Blvd.	Tsawassen	T2F 8M4	Canada
11	B's Beverages	Victoria Ashworth	Fauntleroy Circus	London	EC2 5NT	UK

Πίνακας 17– *Παράδειγμα χαρακτήρων μπαλαντέρ (**Πηγή**: https://www.w3schools.com/sql/sql_wildcards.asp)*

Παραδείγματα με τον χαρακτήρα μπαλαντέρ %

Σε αυτό το παράδειγμα, επιλέγουμε όλους τους πελάτες με Πόλη που ξεκινά με «βερ»:

```
SELECT * FROM Πελάτες
WHERE Πόλη LIKE ‘βερ%';
```

Στο παρακάτω παράδειγμα, επιλέγουμε όλους τους πελάτες με Πόλη που περιέχει το «εσ»:

```
SELECT * FROM Πελάτες
WHERE Πόλη LIKE ‘%εσ%';
```

Παραδείγματα με τον χαρακτήρα μπαλαντέρ «_»

Εδώ, επιλέγουμε όλους τους πελάτες που μένουν σε Πόλη ξεκινά με οποιονδήποτε χαρακτήρα ακολουθούμενο από «ονδίνο»:

```
SELECT * FROM Πελάτες
WHERE Πόλη LIKE ‘_ονδίνο';
```

Σε αυτό το παράδειγμα, επιλέγουμε ξανά όλους τους πελάτες από Πόλη που ξεκινά με ‘Λ’, ακολουθούμενο από οποιονδήποτε χαρακτήρα, ακολουθούμενο από «νδ», ακολουθούμενο από οποιονδήποτε χαρακτήρα, ακολουθούμενο από «νο»:

```
SELECT * FROM Πελάτες
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
WHERE Πόλη LIKE ‘Λ_νδ_νο';
```

Παραδείγματα με τον χαρακτήρα μπαλαντέρ [charlist]

Εδώ, επιλέγουμε όλους τους πελάτες από Πόλη που ξεκινά από «β», «σ» ή «π»:

```
SELECT * FROM Πελάτες
WHERE Πόλη LIKE ‘[βσπ]%';
```

Στο παρακάτω παράδειγμα, επιλέγουμε όλους τους πελάτες από Πόλη που να ξεκινά από «α», «β» ή «γ»:

```
SELECT * FROM Πελάτες
WHERE Πόλη LIKE '[α-γ]%';
```

Παραδείγματα με τον χαρακτήρα μπαλαντέρ [!charlist]

Το θαυμαστικό εμφανίζει χαρακτήρες που δεν περιέχουν μια καθορισμένη συμβολοσειρά. Για παράδειγμα, θέλουμε να επιλέξουμε όλους τους πελάτες από Πόλη που δεν ξεκινά από «β», «σ» ή «π»:

```
SELECT * FROM Πελάτες
WHERE Πόλη LIKE ‘[!βσπ]%';
```

Εναλλακτικά, μπορούμε να χρησιμοποιήσουμε τα ακόλουθα:

```
SELECT * FROM Πελάτες
WHERE Πόλη NOT LIKE ‘[βσπ]%';
```

Τελεστής In στην SQL

Ο τελεστής IN χρησιμοποιείται για τον καθορισμό πολλαπλών τιμών σε έναν όρο WHERE. Μπορεί να θεωρηθεί ότι πληροί διάφορες προϋποθέσεις.

Σύνταξη 1:

```
SELECT όνομα(τα)_στήλης(ών)
FROM όνομα_πίνακα
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
WHERE όνομα_στήλης IN (τιμή1, τιμή2, ...);
```

Σύνταξη 2:

```
SELECT όνομα(τα)_στήλης(ών)
FROM όνομα_πίνακα
WHERE όνομα_στήλης IN (εντολή SELECT);
```

Υπάρχουν δύο τρόποι για να χρησιμοποιήσετε τον τελεστή IN, όπως έχουμε δει.

Ας υποθέσουμε ότι έχουμε έναν πίνακα που ονομάζεται «Πελάτες» που περιέχει τις ακόλουθες στήλες: ΚωδικόςΠελάτη, ΌνομαΠελάτη, ΌνομαΕπικοινωνίας, Διεύθυνση, Πόλη, Ταχυδρομικός Κώδικας και Χώρα.

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas	Martín Sommer	C/ Araquil, 67	Madrid	28023	Spain

***Πίνακας 18**– Πίνακας Πελατών με παραδείγματα τελεστή IN (**Πηγή**: https://www.w3schools.com/sql/sql_in.asp)*

Για παράδειγμα, θέλουμε να επιλέξουμε όλους τους πελάτες που βρίσκονται στη Γερμανία, τη Γαλλία ή το Ηνωμένο Βασίλειο:

```
SELECT * FROM Πελάτες
WHERE Χώρα IN (‘Γερμανία’, ‘Γαλλία’, ‘Ηνωμένο Βασίλειο’)
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ένα άλλο παράδειγμα είναι η επιλογή όλων των πελατών που **δεν** βρίσκονται στη Γερμανία, τη Γαλλία ή το Ηνωμένο Βασίλειο:

```
SELECT * FROM Πελάτες
WHERE Χώρα NOT IN (‘Γερμανία’, ‘Γαλλία’, ‘Ηνωμένο Βασίλειο’)
```

Ας εξετάσουμε επίσης ένα τρίτο παράδειγμα όπου θέλουμε να επιλέξουμε πελάτες που προέρχονται από τις ίδιες χώρες με τους προμηθευτές:

```
SELECT * FROM Πελάτες
WHERE Χώρα IN (SELECT Χώρα FROM Προμηθευτές)
```

Τελεστής Between στην SQL

Ο τελεστής BETWEEN παρέχει ένα εύρος τιμών από τις οποίες μπορείτε να επιλέξετε. Οι τιμές μπορεί να είναι κείμενο, αριθμοί ή ημερομηνίες. Ο τελεστής BETWEEN περιλαμβάνει τις τιμές έναρξης και λήξης.

Σύνταξη:

```
SELECT όνομα(τα)_στήλης(ών)
FROM όνομα_πίνακα
WHERE όνομα_στήλης BETWEEN τιμή1 AND τιμή2;
```

Ας υποθέσουμε ότι έχουμε τον ακόλουθο πίνακα, ο οποίος περιέχει πληροφορίες για διαφορετικά προϊόντα.

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
1	Chais	1	1	10 boxes x 20 bags	18
2	Chang	1	1	24 - 12 oz bottles	19
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10
4	Chef Anton's Cajun Seasoning	1	2	48 - 6 oz jars	22
5	Chef Anton's Gumbo Mix	1	2	36 boxes	21.35

Πίνακας 19– *Πίνακας με παραδείγματα τελεστή BETWEEN*

(***Πηγή***: https://www.w3schools.com/sql/sql_between.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Θα υπάρχει μια σειρά παραδειγμάτων με τους ακόλουθους τελεστές: BETWEEN, NOT BETWEEN, BETWEEN με IN, BETWEEN και NOT BETWEEN με τιμές κειμένου και ημερομηνίες.

Παράδειγμα BETWEEN: Εμφάνιση όλων των προϊόντων με εύρος τιμών 10 έως 20.

```
SELECT * FROM Πρόϊόντα
WHERE Τιμή BETWEEN 10 AND 20;
```

Παράδειγμα NOT BETWEEN: Εμφάνιση όλων των προϊόντων εκτός της σειράς που ορίσαμε στο προηγούμενο παράδειγμα.

```
SELECT * FROM Προϊόντα
WHERE Τιμή NOT BETWEEN 10 AND 20;
```

Παράδειγμα BETWEEN με IN: Εμφάνιση όλων των προϊόντων με εύρος τιμών 10 έως 20 και δεν εμφανίζει προϊόντα με αναγνωριστικό κατηγορίας 1, 2 ή 3.

```
SELECT * FROM Προϊόντα
WHERE Τιμή BETWEEN 10 AND 20
AND ΑριθμόςΚατηγορίας NOT IN (1,2,3);
```

Παράδειγμα BETWEEN με τιμές κειμένου: Εμφάνιση όλων των πρόϊόντων με Όνομα προϊόντος μεταξύ Carnarvon Tigers και Mozzarella di Giovanni.

```
SELECT * FROM Προϊόντα
WHERE ΌνομαΠροϊόντος BETWEEN 'Carnarvon Tigers' AND 'Mozzarella di Giovanni'
ORDER BY ΌνομαΠροϊόντος;
```

Παράδειγμα NOT BETWEEN με τιμές κειμένου: Εμφάνιση όλων των προϊόντων με Όνομα προϊόντος που δεν είναι μεταξύ Carnarvon Tigers και Mozzarella di Giovanni.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
SELECT * FROM Προϊόντα
WHERE ΌνομαΠροϊόντος NOT BETWEEN 'Carnarvon Tigers' AND 'Mozzarella di Giovanni'
ORDER BY ΌνομαΠροϊόντος;
```

Υποθέστε ότι έχουμε τον ακόλουθο πίνακα που περιέχει πληροφορίες για διαφορετικές παραγγελίες:

OrderID	CustomerID	EmployeeID	OrderDate	ShipperID
10248	90	5	7/4/1996	3
10249	81	6	7/5/1996	1
10250	34	4	7/8/1996	2
10251	84	3	7/9/1996	1
10252	76	4	7/10/1996	2

__Πίνακας 20__– Πίνακας με παράδειγμα τελεστή BETWEEN (__Πηγή__: https://www.w3schools.com/sql/sql_between.asp)

Παράδειγμα BETWEEN σε ημερομηνίες: Εμφάνιση όλων των παραγγελιών με ημερομηνία παραγγελίας μεταξύ '01-Ιουλίου-1996' και '31-Ιουλίου-1996'

Υπάρχουν δύο τρόποι για να γίνει αυτό, χρησιμοποιώντας είτε ένα hashtag (#) ή εισαγωγικά (‘’):

```
SELECT * FROM Παραγγελίες
WHERE ΗμερομηνίαΠαραγγελίας BETWEEN #07/01/1996# AND #07/31/1996#;
```

Ή

```
SELECT * FROM Παραγγελίες
WHERE ΗμερομηνίαΠαραγγελίας BETWEEN '1996-07-01' AND '1996-07-31';
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ψευδώνυμα στην SQL (Aliases)

Τα ψευδώνυμα δίνουν ένα προσωρινό όνομα σε έναν πίνακα ή μια στήλη μέσα σε έναν πίνακα. Ένα ψευδώνυμο υπάρχει μόνο για τη διάρκεια ενός ερωτήματος και χρησιμοποιείται συνήθως για να κάνει τα ονόματα των στηλών πιο ευανάγνωστα. Ένα ψευδώνυμο δημιουργείται χρησιμοποιώντας τη λέξη-κλειδί **AS.**

Σύνταξη για ψευδώνυμο στήλης:

```
SELECT όνομα_στήλης AS όνομα_ψευδώνυμου
FROM όνομα_πίνακα;
```

Σύνταξη για το ψευδώνυμο πίνακα:

```
SELECT όνομα(τα)_στήλης(ών)
FROM όνομα_πίνακα AS όνομα_ψευδώνυμου;
```

Ψευδώνυμα στηλών

Ας δούμε ένα παράδειγμα που δημιουργεί δύο ψευδώνυμα, ένα για κάθε στήλη:

```
SELECT ΚωδικόςΠελάτη AS ID, ΌνομαΠελάτη AS Πελάτης
FROM Πελάτες;
```

Ένα άλλο παράδειγμα δημιουργεί και πάλι δύο ψευδώνυμα:

```
SELECT ΌνομαΠελάτη AS Πελάτης, ΌνομαΕπικοινωνίας AS [Άτομο Επικοινωνίας]
FROM Πελάτες;
```

Σημειώστε ότι τοποθετείται σε αγκύλες ([ ]) επειδή το ψευδώνυμο περιέχει κενά. Τα εισαγωγικά μπορούν να χρησιμοποιηθούν ως εναλλακτική λύση για τις αγκύλες.

Έχετε επίσης την επιλογή να δημιουργήσετε ένα ψευδώνυμο που περιέχει μία ή περισσότερες στήλες, ας δούμε το παρακάτω παράδειγμα για να δούμε πώς λειτουργεί:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
SELECT ΌνομαΠελάτη, Διεύθυνση + ', ' + ΤαχυδρομικόςΚώδικας + ' ' + Πόλη + ', ' + Χώρα AS Διεύθυνση
FROM Πελάτες;
```

Η παραπάνω εντολή αλλάζει λίγο στο MySQL:

```
SELECT ΌνομαΠελάτη, CONCAT(Διεύθυνση,', ',ΤαχυδρομικόςΚώδικας,', ',Πόλη,', ',Χώρα) AS Διεύθυνση
FROM Πελάτες;
```

Ψευδώνυμα πίνακα

Το παρακάτω παράδειγμα επιλέγει όλες τις παραγγελίες από τον πίνακα πελατών με ΚωδικόΠελάτη =4 δίνοντας το ψευδώνυμο “Around the Horn”.

Εδώ χρησιμοποιούνται ψευδώνυμα για τη συντόμευση του ερωτήματος:

```
SELECT o.ΚωδικόςΠαραγγελίας, o.ΗμερομηνίαΠαραγγελίας, c.ΌνομαΠελάτη
FROM Πελάτες AS c, Παραγγελίες AS o
WHERE c.Όνομα Πελάτη='Around the Horn' AND c.ΚωδικόςΠελάτη=o.ΚωδικόςΠελάτη;
```

Ένα ερώτημα χωρίς ψευδώνυμα θα έμοιαζε κάπως έτσι:

```
SELECT Παραγγελίες.ΚωδικόςΠαραγγελίας, Παραγγελίες.ΗμερομηνίαΠαραγγελίας, Πελάτες.ΌνομαΠελάτη
FROM Πελάτες, Παραγγελίες
WHERE Πελάτες.ΌνομαΠελάτη='Around the Horn' AND Πελάτες.ΚωδικόςΠελάτη=Παραγγελίες.ΚωδικόςΠελάτη;
```

Συνενώσεις στην SQL (JOIN)

Μια εντολή JOIN συνδυάζει σειρές από δύο ή περισσότερους πίνακες με βάση μια σχετική στήλη που βρίσκεται και στους δύο πίνακες.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ας δούμε τον πίνακα Παραγγελίες και τον πίνακα Πελάτες:

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

Πίνακες 21 & 22– *Πίνακες Παραγγελιών και Πελατών με παραδείγματα συνένωσης JOIN (**Πηγή**: https://www.w3schools.com/sql/sql_join.asp)*

Αν κοιτάξετε τους δύο πίνακες, θα παρατηρήσετε μια κοινή στήλη που ονομάζεται ΚωδικόςΠελάτη (CustomerID).

Με βάση την κοινή στήλη, μπορούμε να δημιουργήσουμε μια εντολή SQL που χρησιμοποιεί το INNER JOIN, η οποία επιλέγει εγγραφές που έχουν τις ίδιες τιμές και στους δύο πίνακες.

```
SELECT Παραγγελίες.ΚωδικόςΠαραγγελίας, Πελάτες.ΌνομαΠελάτη, Παραγγελίες.ΗμερομηνίαΠαραγγελίας
FROM Παραγγελίες
INNER JOIN Πελάτες ON Παραγγελίες.ΚωδικόςΠελάτη = Πελάτες.ΚωδικόςΠελάτη;
```

Αυτή η εντολή θα δημιουργήσει κάτι σαν τον ακόλουθο πίνακα:

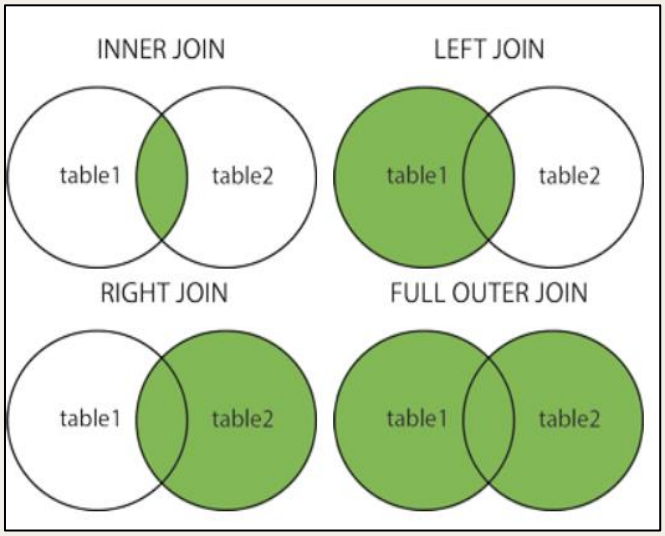
OrderID	CustomerName	OrderDate
10308	Ana Trujillo Emparedados y helados	9/18/1996
10365	Antonio Moreno Taquería	11/27/1996
10383	Around the Horn	12/16/1996
10355	Around the Horn	11/15/1996
10278	Berglunds snabbköp	8/12/1996

Πίνακας 23– *Πίνακας με παράδειγμα συνένωσης JOIN (**Πηγή**: https://www.w3schools.com/sql/sql_join.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Υπάρχουν τέσσερις διαφορετικές εντολές JOIN στην SQL:

1. (INNER) JOIN: Επιλέγει εγγραφές που έχουν αντίστοιχες τιμές και στους δύο πίνακες.
2. LEFT (OUTER) JOIN: Επιλέγει όλες τις εγγραφές από τον πίνακα στα αριστερά και τις αντίστοιχες αντιστοιχισμένες εγγραφές από τον πίνακα στα δεξιά.
3. RIGHT (OUTER) JOIN: Επιλέγει όλες τις εγγραφές από τον πίνακα στα δεξιά και τις αντίστοιχες αντιστοιχισμένες εγγραφές από τον πίνακα στα αριστερά.
4. FULL (OUTER) JOIN: Επιλέγει όλες τις εγγραφές όταν υπάρχει αντιστοιχία είτε στον πίνακα στα αριστερά ή στα δεξιά.



***Εικόνα 1**– Διαφορετικοί τύποι εντολών JOIN (**Πηγή**: https://www.w3schools.com/sql/sql_join.asp)*

Συνένωση Inner join στην SQL

Η λέξη-κλειδί INNER JOIN επιλέγει εγγραφές που έχουν ίδιες τιμές και από τους δύο πίνακες.

Σύνταξη:

```
SELECT όνομα(τα)_στήλης(ών)
FROM πίνακας1
INNER JOIN πίνακας2
ON πίνακας1.όνομα_στήλης = πίνακας2.όνομα_στήλης;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Οι ίδιοι πίνακες με το παράδειγμα της προηγούμενης υποενότητας χρησιμοποιούνται για την εκτέλεση μιας εσωτερικής ένωσης.

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

Πίνακες 24 & 25– *Πίνακες Παραγγελιών και Πελατών με παράδειγμα συνένωσης JOIN*

(***Πηγή***: https://www.w3schools.com/sql/sql_join.asp)

Σε αυτό το παράδειγμα, θέλουμε να ανακτήσουμε τα ονόματα των πελατών και τους αντίστοιχους κωδικούς της παραγγελίας τους:

```
SELECT Παραγγελίες.ΚωδικόςΠαραγγελίας, Πελάτες.ΌνομαΠελάτη
FROM Παραγγελίες
INNER JOIN Πελάτες ON Παραγγελίες.ΚωδικόςΠελάτη = Πελάτες.ΚωδικόςΠελάτη;
```

Σημειώστε ότι η λέξη κλειδί INNER JOIN θα επιλέξει όλες τις σειρές και από τους δύο πίνακες που έχουν τις ίδιες τιμές. Εάν οι εγγραφές στον πίνακα Παραγγελίες δεν έχουν αντιστοιχίσεις στον πίνακα Πελάτες, δεν θα επιλεγούν.

Στο παρακάτω παράδειγμα, θα δούμε πώς μπορείτε να ενώσετε τρεις πίνακες που περιέχουν πληροφορίες πελατών και αποστολέων:

```
SELECT Παραγγελίες.ΚωδικόςΠαραγγελίας, Πελάτες.ΌνομαΠελάτη,
Αποστολέας.ΌνομαΑποστολέα
FROM ((Παραγγελίες
INNER JOIN Πελάτες ON Παραγγελίες.ΚωδικόςΠελάτη = Πελάτες.ΚωδικόςΠελάτη)
INNER JOIN Αποστολείς ON Παραγγελίες.ΚωδικόςΑποστολέα =
Αποστολείς.ΚωδικόςΑποστολέα);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Συνένωση Left Join στην SQL

Η λέξη-κλειδί LEFT JOIN επιλέγει όλες τις εγγραφές από τον πίνακα στα αριστερά και τις αντίστοιχες εγγραφές από τον πίνακα στα δεξιά. Εάν δεν βρεθούν αντιστοιχίσεις, θα εμφανιστούν μηδενικές εγγραφές από τον πίνακα στα δεξιά.

Σύνταξη:

```
SELECT όνομα(τα)_στήλης(ών)
FROM πίνακας1
LEFT JOIN πίνακας2
ON πίνακας1.όνομα_στήλης = πίνακας2.όνομα_στήλης;
```

Σημειώστε ότι η εντολή LEFT JOIN ονομάζεται LEFT OUTER JOIN σε ορισμένες βάσεις δεδομένων.

Για παράδειγμα, ας επιλέξουμε όλους τους πελάτες και τυχόν παραγγελίες που μπορεί να έχουν αυτοί οι πελάτες:

```
SELECT Πελάτες.ΌνομαΠελάτη, Παραγγελίες.ΚωδικόςΠαραγγελίας
FROM Πελάτες
LEFT JOIN Παραγγελίες ON Πελάτες.ΚωδικόςΠελάτη = Παραγγελίες.ΚωδικόςΠελάτη
ORDER BY Πελάτες.ΌνομαΠελάτη;
```

Σημειώστε ότι θα εμφανιστούν όλες οι εγγραφές από τον αριστερό πίνακα με όνομα Πελάτες, ακόμη και αν δεν υπάρχουν αντιστοιχίσεις στον δεξιό πίνακα με όνομα Παραγγελίες.

Συνένωση Right join στην SQL

Η λέξη-κλειδί RIGHT JOIN ακολουθεί ουσιαστικά την ίδια λογική ξεκινώντας από τη δεξιά πλευρά αντί για την αριστερή όπως περιγράφεται στην προηγούμενη υποενότητα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η εντολή RIGHT JOIN θα εμφανίσει όλες τις εγγραφές από τον πίνακα στα δεξιά και τις αντίστοιχες εγγραφές από τον πίνακα στα αριστερά, εάν υπάρχουν.

Σύνταξη:

```
SELECT όνομα(τα)_στήλης(ών)
FROM πίνακας1
RIGHT JOIN πίνακας2
ON πίνακας1.όνομα_στήλης = πίνακας2.όνομα_στήλης;
```

Εξετάστε τους ακόλουθους δύο πίνακες, τους πίνακες Παραγγελιών και Εργαζομένων:

OrderID	CustomerID	EmployeeID	OrderDate	ShipperID
10308	2	7	1996-09-18	3
10309	37	3	1996-09-19	1
10310	77	8	1996-09-20	2

Πίνακας 26– *Πίνακας Παραγγελιών με παράδειγμα συνένωσης RIGHT JOIN*
*(**Πηγή**: https://www.w3schools.com/sql/sql_join_right.asp)*

EmployeeID	LastName	FirstName	BirthDate	Photo
1	Davolio	Nancy	12/8/1968	EmpID1.pic
2	Fuller	Andrew	2/19/1952	EmpID2.pic
3	Leverling	Janet	8/30/1963	EmpID3.pic

Πίνακας 27– *Πίνακας Εργαζομένων με παράδειγμα συνένωσης RIGHT JOIN*
*(**Πηγή**: https://www.w3schools.com/sql/sql_join_right.asp)*

Το ακόλουθο παράδειγμα θα εμφανίσει όλους τους εργαζομένους και τυχόν παραγγελίες που μπορεί να έχουν κάνει:

```
SELECT Παραγγελίες.ΚωδικόςΠαραγγελίας, Εργαζόμενοι.Επίθετο, Εργαζόμενοι.Όνομα
FROM Παραγγελίες
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
RIGHT JOIN Εργαζόμενοι ON Παραγγελίες.ΚωδικόςΕργαζομένου = Εργαζόμενοι.ΚωδικόςΕργαζομένου
ORDER BY Παραγγελίες.ΚωδικόςΠαραγγελίας;
```

Σημειώστε ότι η λέξη κλειδί RIGHT JOIN εμφανίζει όλα τα αρχεία από τον πίνακα στα δεξιά με όνομα Εργαζόμενοι, ακόμα και αν δεν βρεθούν αντιστοιχίες στον αριστερό πίνακα με όνομα Παραγγελίες. Το ίδιο ισχύει και για την εντολή LEFT JOIN που είδαμε νωρίτερα.

Συνένωση Full Join στην SQL

Η λέξη-κλειδί FULL JOIN επιλέγει όλες τις εγγραφές όταν οι εγγραφές που είναι ίδιες βρίσκονται είτε στον πίνακα στα δεξιά είτε στα αριστερά.

Σημειώστε ότι οι εντολές FULL OUTER JOIN και FULL JOIN είναι το ίδιο πράγμα.

Σύνταξη:

```
SELECT όνομα(τα)_στήλης(ών)
FROM πίνακας1
FULL OUTER JOIN πίνακας2
ON πίνακας1.όνομα_στήλης = πίνακας2.όνομα_στήλης
WHERE συνθήκη;
```

Λάβετε υπόψη ότι μια εντολή FULL JOIN μπορεί δυνητικά να επιστρέψει μεγάλα σύνολα αποτελεσμάτων.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παρατηρήστε τους ακόλουθους δύο πίνακες, τον πίνακα Παραγγελιών και τον πίνακα Πελατών:

OrderID	CustomerID	EmployeeID	OrderDate	ShipperID
10308	2	7	1996-09-18	3
10309	37	3	1996-09-19	1
10310	77	8	1996-09-20	2

Πίνακας 28– *Πίνακας Παραγγελιών με παράδειγμα συνένωσης FULL JOIN*

*(**Πηγή**: https://www.w3schools.com/sql/sql_join_full.asp)*

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

Πίνακας 29– *Πίνακας Πελατών με παράδειγμα συνένωσης FULL JOIN*

*(**Πηγή**: https://www.w3schools.com/sql/sql_join_full.asp)*

Ένα παράδειγμα που επιλέγει όλους τους πελάτες και τις παραγγελίες:

```
SELECT Πελάτες.ΌνομαΠελάτη, Παραγγελλίες.ΚωδικόςΠαραγγελίας
FROM Πελάτες
FULL OUTER JOIN Παραγγελίες ON Πελάτες.ΚωδικόςΠελάτη
=Παραγγελίες.ΚωδικόςΠελάτη
ORDER BY Πελάτες.ΌνομαΠελάτη;
```

Το αποτέλεσμα της εντολής FULL JOIN μπορεί να μοιάζει κάπως έτσι:

CustomerName	OrderID
Null	10309
Null	10310
Alfreds Futterkiste	*Null*
Ana Trujillo Emparedados y helados	10308
Antonio Moreno Taquería	*Null*

Πίνακας 30– *Εντολή FULL JOIN σε πίνακες Πελατών και Παραγγελιών - παράδειγμα συνένωσης FULL JOIN*

*(**Πηγή**: https://www.w3schools.com/sql/sql_join_full.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εδώ μπορούμε να δούμε ότι εμφανίζει όλες τις εγγραφές που ταιριάζουν και από τους δύο πίνακες, ακόμη και αν δεν υπάρχουν κοινές αντιστοιχίες μεταξύ των δύο πινάκων. Σε περίπτωση που δεν υπάρχουν κοινές αντιστοιχίσεις, ορίζεται μηδενική τιμή.

Συνένωση Self-Join στην SQL

Η συνένωση SELF-JOIN θεωρείται ως μια κανονική συνένωση JOIN, ωστόσο οι εγγραφές στον πίνακα συσχετίζονται αυτόματα.

Σύνταξη:

```
SELECT όνομα(τα)_στήλης(ών)
FROM πίνακας1 T1, πίνακας1 T2
WHERE συνθήκη;
```

Τα T1 και T2 είναι ψευδώνυμα που χρησιμοποιούνται για τον ίδιο πίνακα.

Ας πάρουμε τον πίνακα Πελατών ως παράδειγμα:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

Πίνακας 31– *Πίνακας Πελατών με παράδειγμα συνένωσης SELF JOIN*
*(**Πηγή**: https://www.w3schools.com/sql/sql_join_self.asp)*

Εδώ, θέλουμε να επιλέξουμε πελάτες που προέρχονται από την ίδια πόλη:

```
SELECT Α.ΌνομαΠελάτη AS ΌνομαΠελάτη1, Β.ΌνομαΠελάτη AS ΌνομαΠελάτη2, Α.Πόλη
FROM Πελάτες Α, Πελάτες Β
WHERE Α.ΚωδικόςΠελάτη <> Β.ΚωδικόςΠελάτη
AND Α.Πόλη = Β.Πόλη
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
ORDER BY A.Πόλη;
```

Ένωση στην SQL (UNION)

Ο τελεστής UNION χρησιμοποιείται για να συνδυάσει το σύνολο δύο ή περισσότερων εντολών SELECT.

Υπάρχουν ορισμένες απαιτήσεις για να καταστεί δυνατός ένας τελεστής UNION:

1. Κάθε εντολή SELECT εντός του τελεστή UNION πρέπει να έχει τον ίδιο αριθμό στηλών
2. Οι στήλες πρέπει να έχουν παρόμοιους τύπους δεδομένων.
3. Οι στήλες σε κάθε εντολή SELECT πρέπει να είναι στην ίδια σειρά.

Σύνταξη:

```
SELECT όνομα_στήλης(ών) FROM πίνακας1
UNION
SELECT όνομα_στήλης(ών) FROM πίνακας2;
```

* Σημειώστε ότι η λειτουργία UNION επιλέγει μόνο διακριτές τιμές από προεπιλογή.

Για να επιτρέψετε τις διπλές τιμές, χρησιμοποιήστε την λειτουργία UNION ALL:

```
SELECT όνομα(τα)_στήλης(ών) FROM πίνακας1
UNION ALL
SELECT όνομα(τα)_στήλης(ών) FROM πίνακας2;
```

* Σημειώστε ότι τα ονόματα των στηλών στις δύο εντολές SELECT είναι συνήθως ίσα.

Τώρα ας δούμε μερικά παραδείγματα της ένωσης UNION, της UNION ALL, και UNION με εντολές που χρησιμοποιούν τον όρο WHERE για να καταλάβουμε λίγο καλύτερα πώς μπορούμε να το χρησιμοποιήσουμε.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

Πίνακας 32– *Πίνακας Πελατών με παράδειγμα τελεστή UNION*

(***Πηγή***: https://www.w3schools.com/sql/sql_union.asp)

SupplierID	SupplierName	ContactName	Address	City	PostalCode	Country
1	Exotic Liquid	Charlotte Cooper	49 Gilbert St.	London	EC1 4SD	UK
2	New Orleans Cajun Delights	Shelley Burke	P.O. Box 78934	New Orleans	70117	USA
3	Grandma Kelly's Homestead	Regina Murphy	707 Oxford Rd.	Ann Arbor	48104	USA

Πίνακας 33– *Πίνακας Προμηθευτών με παράδειγμα τελεστή UNION*

(***Πηγή***: https://www.w3schools.com/sql/sql_union.asp)

Το πρώτο παράδειγμα χρησιμοποιείται για την επιλογή ξεχωριστών πόλεων από τους δύο πίνακες που παρουσιάζονται παραπάνω:

```
SELECT Πόλη FROM Πελάτες
UNION
SELECT Πόλη FROM Προμηθευτές
ORDER BY Πόλη;
```

Εφόσον χρησιμοποιούμε τον τελεστή UNION, οι προμηθευτές από την ίδια πόλη θα αναφέρονται μόνο μία φορά. Αν θέλετε να δείτε τις διπλές τιμές, χρησιμοποιήστε τον τελεστή UNION ALL.

Το ακόλουθο παράδειγμα θα κάνει ακριβώς αυτό και θα εμφανίσει τυχόν διπλές τιμές και από τους δύο πίνακες:

```
SELECT Πόλη FROM Πελάτες
UNION ALL
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
SELECT Πόλη FROM Προμηθευτές
ORDER BY Πόλη;
```

Το ακόλουθο παράδειγμα θα εμφανίσει τις ξεχωριστές γερμανικές πόλεις από τους πίνακες «Πελάτες» και «Προμηθευτές» με τη χρήση του όρου WHERE:

```
SELECT Πόλη, Χώρα FROM Πελάτες
WHERE Χώρα='Γερμανία'
UNION
SELECT Πόλη, Χώρα FROM Πελάτες
WHERE Χώρα = 'Γερμανία'
ORDER BY Πόλη;
```

Αυτό το παράδειγμα είναι παρόμοιο με το προηγούμενο, ωστόσο εδώ εμφανίζονται πιθανές διπλές τιμές:

```
SELECT Πόλη, Χώρα FROM Πελάτες
WHERE Χώρα = 'Γερμανία'
UNION ALL
SELECT Πόλη, Χώρα FROM Πελάτες
WHERE Χώρα = 'Γερμανία'
ORDER BY Πόλη;
```

Ένα άλλο παράδειγμα θα απαριθμήσει όλους τους πελάτες και τους προμηθευτές:

```
SELECT Πελάτης AS Είδος, ΌνομαΕπικοινωνίας, Πόλη, Χώρα
FROM Πελάτες
UNION
SELECT 'Προμηθευτής', ΌνομαΕπικοινωνίας, Πόλη, Χώρα
FROM Προμηθευτές;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Όπως βλέπετε, εδώ χρησιμοποιήσαμε τον όρο AS για να δημιουργήσουμε ένα ψευδώνυμο για το δεδομένο ερώτημα που θα εξαφανιστεί μετά την ολοκλήρωσή του.

Ομαδοποίηση στην SQL (Group By)

Η εντολή GROUP BY ομαδοποιεί τις σειρές με τις ίδιες τιμές σε συνοπτικές γραμμές. Για παράδειγμα, σκεφτείτε ότι θέλετε να βρείτε τον αριθμό των πελατών σε κάθε χώρα. Επίσης, η εντολή GROUP BY χρησιμοποιείται συχνά με αθροιστικές συναρτήσεις όπως COUNT(), MAX(), MIN(), SUM(), AVG() για την ομαδοποίηση του αποτελέσματος κατά μία ή περισσότερες στήλες.

Σύνταξη:

```
SELECT όνομα_στήλης(ών)
FROM όνομα_πίνακα
WHERE συνθήκη
GROUP BY όνομα_στήλης(ών)
ORDER BY όνομα_στήλης(ών);
```

Θα χρησιμοποιήσουμε τον πίνακα Πελατών, που φαίνεται παρακάτω, στα παραδείγματα μας.

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

Πίνακας 34– *Πίνακας Πελατών με παράδειγμα εντολής GROUP BY*

(***Πηγή***: https://www.w3schools.com/sql/sql_groupby.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ως πρώτο παράδειγμα, θα απαριθμήσουμε τον αριθμό των πελατών που βρέθηκαν σε κάθε χώρα:

```
SELECT COUNT(ΚωδικόςΠελάτη), Χώρα
FROM Πελάτες
GROUP BY Χώρα;
```

Ως δεύτερο παράδειγμα, θα απαριθμήσουμε και πάλι τον αριθμό των πελατών σε κάθε χώρα, αλλά με φθίνουσα σειρά.

```
SELECT COUNT(ΚωδικόςΠελάτη), Χώρα
FROM Πελάτες
GROUP BY Χώρα;
ORDER BY COUNT(ΚωδικόςΠελάτη) DESC;
```

Στο παρακάτω παράδειγμα, θα χρησιμοποιήσουμε την εντολή GROUP BY με τον τελεστή JOIN χρησιμοποιώντας τους πίνακες Παραγγελιών και Αποστολέων.

OrderID	CustomerID	EmployeeID	OrderDate	ShipperID
10248	90	5	1996-07-04	3
10249	81	6	1996-07-05	1
10250	34	4	1996-07-08	2

Πίνακας 35– *Πίνακας Παραγγελιών με παράδειγμα εντολής GROUP BY*

(***Πηγή***: https://www.w3schools.com/sql/sql_groupby.asp)

ShipperID	ShipperName
1	Speedy Express
2	United Package
3	Federal Shipping

Πίνακας 36– *Πίνακας Αποστολέων με παράδειγμα εντολής GROUP BY*

(***Πηγή***: https://www.w3schools.com/sql/sql_groupby.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σε αυτό το παράδειγμα, θα αναφέρουμε τον αριθμό των παραγγελιών που αποστέλλονται από κάθε αποστολέα:

```
SELECT Αποστολέας.ΌνομαΑποστολέα, COUNT(Παραγγελίες.ΚωδικόςΠαραγγελίας) AS ΑριθμόςΠαραγγελιών
FROM Παραγγελίες
LEFT JOIN Αποστολείς ON Παραγγελίες.ΚωδικόςΑποστολέα= Αποστολείς.ΚωδικόςΑποστολέα
GROUP BY ΌνομαΑποστολέα;
```

Όπως φαίνεται στο παραπάνω παράδειγμα, ξεκινήσαμε επιλέγοντας τα ονόματα των Αποστολέων από τον πίνακα Αποστολείς και απαριθμήσαμε τις παραγγελίες με βάση τον κωδικό παραγγελίας τους που αποθηκεύτηκε ως ψευδώνυμο.

Στη συνέχεια, χρησιμοποιήσαμε τον τελεστή LEFT JOIN για να συνδυάσουμε τον πίνακα Αποστολέων (πίνακας 2) και τον πίνακα Παραγγελιών (πίνακας 1) και να τους ομαδοποιήσουμε με βάση το όνομα του Αποστολέα.

Ο όρος HAVING στην SQL

Ο όρος HAVING προστέθηκε στην SQL επειδή ο όρος WHERE δεν μπορεί να χρησιμοποιηθεί με αθροιστικές συναρτήσεις.

Σύνταξη:

```
SELECT όνομα_στήλης(ών)
FROM όνομα_πίνακα
WHERE συνθήκη
GROUP BY όνομα_στήλης(ών)
HAVING συνθήκη
ORDER BY όνομα_στήλης(ών);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

Πίνακας 37– *Πίνακας Πελατών με παράδειγμα του όρου HAVING*

(***Πηγή***: https://www.w3schools.com/sql/sql_having.asp)

Σε αυτό το παράδειγμα, θα χρησιμοποιήσουμε ξανά τον πίνακα Πελατών. Εδώ, θέλουμε να απαριθμήσουμε τον αριθμό των πελατών που βρίσκονται σε κάθε χώρα, αλλά θέλουμε επίσης να συμπεριλάβουμε χώρες που έχουν περισσότερους από πέντε πελάτες.

```
SELECT COUNT(ΚωδικόςΠελάτη), Χώρα
FROM Πελάτες
GROUP BY Χώρα
HAVING COUNT(ΚωδικόςΠελάτη) > 5;
```

Σε αυτό το παράδειγμα, θέλουμε να απαριθμήσουμε ξανά τον αριθμό των πελατών ανά χώρα και να συμπεριλάβουμε χώρες με περισσότερους από πέντε πελάτες σε φθίνουσα σειρά.

```
SELECT COUNT(ΚωδικόςΠελάτη), Χώρα
FROM Πελάτες
GROUP BY Χώρα
HAVING COUNT(ΚωδικόςΠελάτη) > 5;
ORDER BY COUNT(ΚωδικόςΠελάτη) DESC;
```

Ας δοκιμάσουμε μερικά άλλα παραδείγματα συνδυάζοντας αυτά που έχουμε μάθει μέχρι στιγμής.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Θα χρησιμοποιήσουμε τους πίνακες Παραγγελιών και Εργαζομένων στα ακόλουθα δύο παραδείγματα.

OrderID	CustomerID	EmployeeID	OrderDate	ShipperID
10248	90	5	1996-07-04	3
10249	81	6	1996-07-05	1
10250	34	4	1996-07-08	2

Πίνακας 38– *Πίνακας Πελατών με παράδειγμα HAVING*
*(**Πηγή**: https://www.w3schools.com/sql/sql_having.asp)*

EmployeeID	LastName	FirstName	BirthDate	Photo	Notes
1	Davolio	Nancy	1968-12-08	EmpID1.pic	Education includes a BA....
2	Fuller	Andrew	1952-02-19	EmpID2.pic	Andrew received his BTS....
3	Leverling	Janet	1963-08-30	EmpID3.pic	Janet has a BS degree....

Πίνακας 39– *Πίνακας Πελατών με παράδειγμα HAVING*
*(**Πηγή**: https://www.w3schools.com/sql/sql_having.asp)*

Το ακόλουθο παράδειγμα θα απαριθμήσει τους εργαζομένους που έχουν καταχωρίσει περισσότερες από δέκα παραγγελίες:

```
SELECT Εργαζόμενοι.Επίθετο, COUNT(Παραγγελίες.ΚωδικόςΠαραγγελιών) AS ΑριθμόςΠαραγγελιών
FROM (Παραγγελίες
INNER JOIN Εργαζόμενοι ON Παραγγελίες.ΚωδικόςΕργαζομένου = Εργαζόμενοι.ΚωδικόςΕργαζομένου)
GROUP BY Επίθετο
HAVING COUNT(Παραγγελίες.ΚωδικόςΠαραγγελιών) > 10;
```

Σε αυτό το παράδειγμα, θα απαριθμήσουμε τους εργαζόμενους επίθετο «Νταβόλιο» ή «Φούλερ» εάν έχουν καταχωρίσει παραγγελίες περισσότερες από 25 φορές:

```
SELECT Εργαζόμενοι.Επίθετο, COUNT(Παραγγελίες.ΚωδικόςΠαραγγελίας) AS ΑριθμόςΠαραγγελιών
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
FROM (Παραγγελίες
INNER JOIN Εργαζόμενοι ON Παραγγελίες.ΚωδικόςΕργαζομένου = Εργαζόμενοι.ΚωδικόςΕργαζομένων)
WHERE Επίθετο = 'Νταβόλιο' OR Επίθετο = 'Φούλερ'
GROUP BY Επίθετο
HAVING COUNT(Παραγγελίες.ΚωδικόςΠαραγγελιών) > 25;
```

Εντολή Select Into στην SQL

Η εντολή SELECT INTO αντιγράφει τα δεδομένα από έναν πίνακα σε έναν νέο πίνακα.

Σύνταξη για αντιγραφή όλων των στηλών σε νέο πίνακα:

```
SELECT *
INTO νέοςπίνακας [IN externaldb]
FROM παλιόςπίνακας
WHERE συνθήκη;
```

Σύνταξη για αντιγραφή μόνο μερικών στηλών σε νέο πίνακα:

```
SELECT στήλη1, στήλη2, ...
INTO νέοςπίνακας [IN externaldb]
FROM παλιόςπίνακας
WHERE συνθήκη;
```

Ο νέος πίνακας θα διατηρήσει τα ονόματα και τους τύπους των στηλών όπως και ο παλιός πίνακας. Μπορείτε να δημιουργήσετε νέες στήλες με τον όρο AS.

Παράδειγμα δημιουργίας εφεδρικού αντιγράφου της στήλης Πελατών:

```
SELECT * INTO ΑντίγραφοΠελατών2017
FROM Πελάτες;
```

Παράδειγμα χρήσης του όρου IN για την αντιγραφή του πίνακα σε νέο πίνακα σε άλλη βάση δεδομένων:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
SELECT * INTO ΑντίγραφοΠελατών2017 IN 'Backup.mdb'
FROM Πελάτες;
```

Παράδειγμα αντιγραφής μόνο μερικών στηλών σε νέο πίνακα:

```
SELECT ΌνομαΠελάτη, ΌνομαΕπικοινωνίας INTO ΑντίγραφοΠελατών2017
FROM Πελάτες;
```

Παράδειγμα αντιγραφής μόνο των Γερμανών πελατών σε νέο πίνακα:

```
SELECT * INTO ΠελάτεςΓερμανία
FROM Πελάτες
WHERE Χώρα = ‘Γερμανία';
```

Παράδειγμα αντιγραφής δεδομένων από πολλαπλούς πίνακες σε νέο πίνακα:

```
SELECT Πελάτες.ΌνομαΠελάτη, Παραγγελία.ΚωδικόςΠαραγγελίας
INTO ΑντίγραφοΠαραγγελιώνΠελατών2017
FROM Πελάτες
LEFT JOIN Παραγγελίες ON Πελάτες.ΚωδικόςΠελάτη= Παραγγελίες.ΚωδικόςΠελάτη;
```

Η εντολή SELECT INTO μπορεί επίσης να χρησιμοποιηθεί για τη δημιουργία ενός νέου, κενού πίνακα χρησιμοποιώντας το σχήμα ενός άλλου.

Για να το κάνετε αυτό, προσθέστε έναν όρο WHERE που δεν εμφανίζει δεδομένα:

```
SELECT * INTO νέοςπίνακας
FROM παλιόςπίνακας
WHERE 1 = 0;
```

Η εντολή Insert Into Select στην SQL

Η εντολή INSERT INTO SELECT αντιγράφει δεδομένα από έναν πίνακα και τα εισάγει σε έναν άλλον. Απαιτεί την αντιστοίχιση των τύπων δεδομένων στον πίνακα πηγής και στόχου.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σύνταξη για αντιγραφή όλων των στηλών από έναν πίνακα σε έναν άλλο:

```
INSERT INTO πίνακας2
SELECT * FROM πίνακας1
WHERE συνθήκη;
```

Σύνταξη για αντιγραφή μόνο μερικών στηλών από έναν πίνακα σε έναν άλλο:

```
INSERT INTO πίνακας2 (στήλη1, στήλη2, στήλη3, ...)
SELECT στήλη1, στήλη2, στήλη3, ...
FROM πίνακας1
WHERE συνθήκη;
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

Πίνακας 40– *Πίνακας Πελατών με παράδειγμα εντολής INSERT INTO SELECT*

(***Πηγή***: https://www.w3schools.com/sql/sql_insert_into_select.asp)

SupplierID	SupplierName	ContactName	Address	City	Postal Code	Country
1	Exotic Liquid	Charlotte Cooper	49 Gilbert St.	Londona	EC1 4SD	UK
2	New Orleans Cajun Delights	Shelley Burke	P.O. Box 78934	New Orleans	70117	USA
3	Grandma Kelly's Homestead	Regina Murphy	707 Oxford Rd.	Ann Arbor	48104	USA

Πίνακας 41– *Πίνακας Προμηθευτών με παράδειγμα εντολής INSERT INTO SELECT*

(***Πηγή***: https://www.w3schools.com/sql/sql_insert_into_select.asp)

Θα χρησιμοποιήσουμε τους πίνακες Πελατών και Προμηθευτών, που παρουσιάζονται παραπάνω, στα ακόλουθα παραδείγματα.

Το πρώτο παράδειγμα αντιγράφει Προμηθευτές στον πίνακα Πελατών (σημειώστε ότι οι στήλες που δεν έχουν δεδομένα θα περιέχουν μηδενικές τιμές):

```
INSERT INTO Πελάτες (ΌνομαΠελάτη, Πόλη, Χώρα)
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
SELECT ΌνομαΠρομηθευτή, Πόλη, Χώρα FROM Προμηθευτές;
```

Αυτό το παράδειγμα αντιγράφει Προμηθευτές στον πίνακα Πελατών για να συμπληρώσει όλες τις στήλες:

```
INSERT INTO Πελάτες (ΌνομαΠελάτη, ΌνομαΕπικοινωνίας, Διεύθυνση, Πόλη, ΤαχυδρομικόςΚώδικας, Χώρα)
SELECT ΌνομαΠρομηθευτή, ΌνομαΕπικοινωνίας, Διεύθυνση, Πόλη, ΤαχυδρομικόςΚώδικας, Χώρα FROM Προμηθευτές;
```

Το τρίτο παράδειγμα αντιγράφει μόνο τους Γερμανούς προμηθευτές στον πίνακα Πελατών:

```
INSERT INTO Πελάτες (ΌνομαΠελάτη, Πόλη, Χώρα)
SELECT ΌνομαΠρομηθευτή, Πόλη, Χώρα FROM Προμηθευτές
WHERE NOT Χώρα='Γερμανία';
```

Εντολή Case στην SQL

Η εντολή CASE περνά από μια σειρά προϋποθέσεων και επιστρέφει μια τιμή όταν πληρούται η πρώτη προϋπόθεση. Σκεφτείτε το σαν μια εντολή με τα if, then, else.

Όταν ένας όρος ισχύει, θα σταματήσει να περνάει μέσα από τον βρόχο. Εάν δεν ισχύει, θα εμφανίσει την τιμή στον όρο ELSE.

Σημειώστε ότι αν δεν υπάρχει ο όρος ELSE και δεν υπάρχουν όροι που να ισχύουν, θα εμφανίσει το NULL.

Σύνταξη:

```
CASE
WHEN συνθήκη1 THEN αποτέλεσμα1
WHEN συνθήκη2 THEN αποτέλεσμα2
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
WHEN συθήκηΝ THEN αποτέλεσμαΝ
ELSE αποτέλεσμα
END;
```

OrderDetailID	OrderID	ProductID	Quantity
1	10248	11	12
2	10248	42	10
3	10248	72	5
4	10249	14	9
5	10249	51	40

***Πίνακας 42**– Πίνακας Παραγγελιών με παράδειγμα του όρου CASE*

*(**Πηγή**: https://www.w3schools.com/sql/sql_case.asp)*

Στα παρακάτω παραδείγματα, θα χρησιμοποιήσουμε τον πίνακα Παραγγελιών.

Το πρώτο παράδειγμα θα περάσει από μια σειρά συνθηκών και θα εμφανίσει μια τιμή όταν η πρώτη συνθήκη πληρείται:

```
SELECT ΚωδικόςΠαραγγελίας, Ποσότητα,
CASE
WHEN Ποσότητα > 30 THEN ‘Ο ποσότητα είναι μεγαλύτερη από 30'
WHEN Ποσότητα = 30 THEN ‘Η ποσότητα είναι ίση με 30’
ELSE ‘Η ποσότητα είναι μικρότερη από 30'
END AS ΑριθμόςΠοσότητας
FROM ΛεπτομέριεςΠαραγγελίας;
```

Σε αυτό το δεύτερο παράδειγμα, θα ταξινομήσουμε τους πελάτες ανά πόλη. Σημειώστε ότι αν η Πόλη έχει την τιμή NULL, θα ταξινομηθούν ανά Χώρα.

```
SELECT ΌνομαΠελάτη, Πόλη, Χώρα
FROM Πελάτες
ORDER BY
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
    (CASE
        WHEN Πόλη IS NULL THEN Χώρα
        ELSE Πόλη
END);
```

Λειτουργίες κενών τιμών στην SQL (Null Functions)

Οι λειτουργίες NULL περιλαμβάνουν τα ακόλουθα: IFNULL(), ISNULL(), COALESCE(), and NVL().

Εδώ, θα χρησιμοποιήσουμε τον πίνακα «Προϊόντα»:

P_Id	ProductName	UnitPrice	UnitsInStock	UnitsOnOrder
1	Jarlsberg	10.45	16	15
2	Mascarpone	32.56	23	
3	Gorgonzola	15.67	9	20

Πίνακας 43– *Πίνακας Προϊόντων με παράδειγμα λειτουργιών NULL*

(***Πηγή***: *https://www.w3schools.com/sql/sql_isnull.asp*)

Ας υποθέσουμε ότι η στήλη «ΤεμάχιαΠαραγγελίας» είναι προαιρετική και μπορεί να περιέχει κενές τιμές (NULL).

Παράδειγμα:

```
SELECT ΌνομαΠροϊόντος, ΤιμήΤεμαχίου * (ΤεμάχιαΣεΑπόθεμα + ΤεμάχιαΠαραγγελίας)
FROM Προϊόντα;
```

Εδώ μπορούμε να δούμε ότι εάν οποιαδήποτε από τις τιμές ΤεμάχιαΠαραγγελίας είναι κενή, το αποτέλεσμα θα είναι επίσης κενό (NULL).

Ας δούμε πώς μπορούμε να λύσουμε αυτό το ζήτημα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στο σύστημα MySQL, μπορούμε να χρησιμοποιήσουμε τη συνάρτηση ISNULL() που μας επιτρέπει να επιλέξουμε μια εναλλακτική τιμή αν μια συνάρτηση είναι μηδενική:

```
SELECT ΌνομαΠροϊόντος, ΤιμήΤεμαχίου * (ΤεμάχιαΣεΑπόθεμα + IFNULL(ΤεμάχιαΠαραγγελίας, 0))
FROM Προϊόντα;
```

Ή μπορούμε να χρησιμοποιήσουμε τη λειτουργία COALESCE() :

```
SELECT ΌνομαΠροϊόντος, ΤιμήΤεμαχίου * (ΤεμάχιαΣεΑπόθεμα + COALESCE(ΤεμάχιαΠαραγγελίας, 0))
FROM Προϊόντα;
```

Στον διακομιστή SQL, η λειτουργία ISNULL() κάνει το ίδιο πράγμα όπως στο MySQL:

```
SELECT ΌνομαΠροϊόντος, ΤιμήΤεμαχίου * (ΤεμάχιαΣεΑπόθεμα + IFNULL(ΤεμάχιαΠαραγγελίας, 0))
FROM Προϊόντα;
```

Στη λειτουργία IsNull() στο MS Access η συνάρτηση TRUE(-1) προκύπτει αν η έκφραση είναι μια κενή τιμή, αλλιώς προκύπτει η συνάρτηση FALSE (0):

```
SELECT ΌνομαΠροϊόντος, ΤιμήΤεμαχίου * (ΤεμάχιαΣεΑπόθεμα + IIF(IsNull(ΤεμάχιαΠαραγγελίας), 0, ΤεμάχιαΠαραγγελίας))
FROM Προϊόντα;
```

Στο σύστημα Oracle, η λειτουργία NVL() κάνει το ίδιο πράγμα:

```
SELECT ΌνομαΠροϊόντος, ΤιμήΤεμαχίου * (ΤεμάχιαΣεΑπόθεμα + NVL(ΤεμάχιαΠαραγγελίας, 0))
FROM Προϊόντα;
```

Σχόλια στην SQL (Comments)

Τα σχόλια SQL εξηγούν τις ενότητες εντολών στην SQL ή εμποδίζουν την εκτέλεσή τους.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σημειώστε ότι τα παραδείγματα σε αυτήν την ενότητα δεν υποστηρίζονται στο Firefox και το Microsoft Edge, τα οποία είναι βάσεις δεδομένων της Microsoft Access. Τα σχόλια (comments) γενικά δεν υποστηρίζονται στις βάσεις δεδομένων της Microsoft Access.

Τα σχόλια μιας γραμμής στην SQL ξεκινούν με - - (δύο παύλες):

```
--Select all:
SELECT * FROM Πελάτες;
```

Ή μπορεί να χρησιμοποιηθεί με αυτόν τον τρόπο για να αγνοήσει το τέλος της γραμμής:

```
SELECT * FROM Πελάτες -- WHERE Πόλη='Βερολίνο';
```

Ή για να αγνοήσετε μια πρόταση:

```
--SELECT * FROM Πελάτες;
SELECT * FROM Προϊόντα;
```

Τα σχόλια πολλαπλών γραμμών ξεκινούν με /* και τελειώνουν με */. Οποιοδήποτε κείμενο γραφτεί μεταξύ αυτών των δύο θα αγνοηθεί.

Παράδειγμα:

```
/*Επιλογή όλων των στηλών
όλων των εγγραφών
στον Πίνακα Πελατών:*/
SELECT * FROM Πελάτες;
```

Για να αγνοήσετε μέρος μιας πρότασης, μπορείτε επίσης να χρησιμοποιήσετε το/**/.

Παράδειγμα 1:

```
SELECT ΌνομαΠελάτη, /*Πόλη,*/ Χώρα FROM Πελάτες;
```

Παράδειγμα 2:

```
SELECT * FROM Πελάτες WHERE (ΌνομαΠελάτη LIKE 'Λ%'
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
OR ΌνομαΠελάτη LIKE 'P%' /*OR ΌνομαΠελάτη LIKE 'Σ%'
OR ΌνομαΠελάτη LIKE 'T%'*/ OR ΌνομαΠελάτη LIKE 'Ω%')
AND Χώρα='ΗΠΑ'
ORDER BY ΌνομαΠελάτη;
```

Τελεστές στην SQL

Αριθμητικοί τελεστές που χρησιμοποιούνται στην SQL:

Operator	Description
+	Add
-	Subtract
*	Multiply
/	Divide
%	Modulo

Πίνακας 44– *Πίνακας Αριθμητικών τελεστών (**Πηγή**: https://www.w3schools.com/sql/sql_operators.asp)*

Οι τελεστές Bitwise που χρησιμοποιούνται στην SQL:

Operator	Description
&	Bitwise AND
|	Bitwise OR
^	Bitwise exclusive OR

Πίνακας 45– *Πίνακας τελεστών Bitwise (**Πηγή**: https://www.w3schools.com/sql/sql_operators.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Οι τελεστές σύγκρισης που χρησιμοποιούνται στην SQL:

Operator	Description
=	Equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
<>	Not equal to

***Πίνακας 46**– Πίνακας τελεστών σύγκρισης (**Πηγή**: https://www.w3schools.com/sql/sql_operators.asp)*

Σύνθετοι τελεστές που χρησιμοποιούνται στην SQL:

Operator	Description
+=	Add equals
-=	Subtract equals
*=	Multiply equals
/=	Divide equals
%=	Modulo equals
&=	Bitwise AND equals
^-=	Bitwise exclusive equals
|*=	Bitwise OR equals

***Πίνακας 47**– Πίνακας σύνθετων τελεστών (**Πηγή**: https://www.w3schools.com/sql/sql_operators.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Λογικοί τελεστές που χρησιμοποιούνται στην SQL:

Operator	Description
ALL	TRUE if all of the subquery values meet the condition
AND	TRUE if all the conditions separated by AND is TRUE
ANY	TRUE if any of the subquery values meet the condition
BETWEEN	TRUE if the operand is within the range of comparisons
EXISTS	TRUE if the subquery returns one or more records
IN	TRUE if the operand is equal to one of a list of expressions
LIKE	TRUE if the operand matches a pattern
NOT	Displays a record if the condition(s) is NOT TRUE
OR	TRUE if any of the conditions separated by OR is TRUE
SOME	TRUE if any of the subquery values meet the condition

***Πίνακας 48**– Πίνακας λογικών τελεστών (**Πηγή**: https://www.w3schools.com/sql/sql_operators.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

4.2. Βάση δεδομένων στην SQL

Όπως έχουμε αναφέρει στην προηγούμενη υποενότητα που ήταν αφιερωμένη στις βασικές λειτουργίες που χρησιμοποιούνται στην SQL, αυτή η γλώσσα προγραμματισμού χρησιμοποιείται κυρίως για σχεσιακές βάσεις δεδομένων. Ως εκ τούτου, σε αυτή την υποενότητα, θα μάθουμε πώς να δημιουργήσουμε μια βάση δεδομένων, να την τροποποιήσουμε και να την χειριστούμε μέσω της SQL.

Ας ξεκινήσουμε από τις απλές λειτουργίες, προχωρώντας σε ελαφρώς πιο περίπλοκες.

Δημιουργία Βάσης Δεδομένων στην SQL

Η εντολή CREATE DATABASE δημιουργεί μια νέα βάση δεδομένων στην SQL.

Σύνταξη:

```
CREATE DATABASE ΌνομαΒάσηςΔεδομένων;
```

Σημείωση: Να θυμάστε πάντα ότι το όνομα της βάσης δεδομένων πρέπει να είναι μοναδικό εντός του σχεσιακού Συστήματος Διαχείρισης Βάσεων Δεδομένων (ΣΒΔΒ) που χρησιμοποιείτε και βεβαιωθείτε ότι έχετε δικαιώματα διαχειριστή πριν δημιουργήσετε οποιαδήποτε βάση δεδομένων.

Ας πούμε ότι θέλετε να δημιουργήσετε μια δοκιμαστική βάση δεδομένων. Θα χρησιμοποιούσατε την ακόλουθη εντολή:

```
CREATE DATABASE δοκιμαστικήΒΔ;
```

Διαγραφή Βάσης Δεδομένων στην SQL

Η εντολή DROP DATABASE διαγράφει μια υπάρχουσα βάση δεδομένων στην SQL.

```
DROP DATABASE ΌνομαΒάσηςΔεδομένων;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πριν διαγράψετε τη βάση δεδομένων, βεβαιωθείτε ότι δεν χρειάζεστε καμία από τις πληροφορίες που περιέχει, καθώς τη διαγράφει εντελώς.

Θυμάστε τη βάση δεδομένων που μόλις δημιουργήσαμε με την ονομασία «δοκιμαστικήΒΔ»; Τώρα θα την διαγράψουμε.

Παράδειγμα:

```
DROP DATABASE δοκιμαστικήΒΔ;
```

Δημιουργία αντιγράφου της Βάσης Δεδομένων στην SQL

Η εντολή BACKUP DATABASE δημιουργεί ένα πλήρες εφεδρικό αντίγραφο σε μια υπάρχουσα βάση δεδομένων SQL.

Για να χρησιμοποιήσετε αυτή την εντολή, θα πρέπει να γράψετε δύο πράγματα: το όνομα της βάσης δεδομένων και τη διαδρομή του αρχείου.

```
BACKUP DATABASE ΌνομαΒάσηςΔεδομένων
TO DISK = ‘διαδρομήαρχείου’
```

Παράδειγμα:

```
BACKUP DATABASE δοκιμαστικήΒΔ
TO DISK = 'D:\αντίγραφα\δοκιμαστικήΒΔ.bak';
```

Σημείωση: Για να αποφύγετε τεχνικά προβλήματα, είναι καλύτερο να δημιουργήσετε εφεδρικά αντίγραφα της βάσης δεδομένων σε μια διαφορετική μονάδα δίσκου από εκείνη στην οποία βρίσκεται η υπάρχουσα βάση δεδομένων.

Υπάρχει επίσης μια άλλη επιλογή όπου μπορείτε να εκτελέσετε ένα διαφορικό εφεδρικό αντίγραφο με βάση τις αλλαγές που έχουν γίνει από το τελευταίο πλήρες εφεδρικό αντίγραφο της βάσης δεδομένων. Αυτός ο τύπος εφεδρικών αντιγράφων μειώνει επίσης τον χρόνο δημιουργίας εφεδρικών αντιγράφων.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για να το κάνετε αυτό, ακολουθήστε αυτήν τη σύνταξη:

```
BACKUP DATABASE ΌνομαΒάσηςΔεδομένων
TO DISK = 'διαδρομήαρχείου'
WITH DIFFERENTIAL;
```

Παράδειγμα:

```
BACKUP DATABASE δοκιμαστικήΔΒ
TO DISK = 'D:\αντίγραφα\δοκιμαστικήΔΒ.bak'
WITH DIFFERENTIAL;
```

Δημιουργία πίνακα στην SQL

Η εντολή CREATE TABLE δημιουργεί έναν νέο πίνακα σε μια βάση δεδομένων.

Σύνταξη:

```
CREATE TABLE όνομα_πίνακα (
                    στήλη1 τύποςδεδομένων,
                στήλη2 τύποςδεδομένων,
                    στήλη3 τύποςδεδομένων,
                …….
);
```

Σε αυτή τη λειτουργία, θα πρέπει να καθορίσετε τα **ονόματα των στηλών** και τον **τύπο των δεδομένων** που θα περιέχει η στήλη.

Υπάρχουν πολλοί τύποι δεδομένων όπως integer, date ή varchar. Ανάλογα με τον τύπο των δεδομένων που θέλετε να αποθηκεύσετε, επιλέγετε την πιο κατάλληλη επιλογή. Για παράδειγμα, αν έχετε μια στήλη με το όνομα «Ημερομηνία Γέννησης», τότε πιθανότατα θα επιλέγατε την Ημερομηνία ως τον τύπο δεδομένων.

Παράδειγμα:

```
CREATE TABLE Άτομα (
                    ΚωδΑτόμου int,
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
                Επίθετο varchar(255),
                Όνομα varchar(255),
                Διεύθυνση varchar(255),
                Πόλη varchar(255)
);
```

Αυτό το παράδειγμα θα δημιουργήσει έναν πίνακα με το όνομα Άτομα και θα περιέχει 5 στήλες.

Η στήλη ΚωδΑτόμου θα περιέχει έναν ακέραιο αριθμό (int). Οι στήλες Επίθετο, Όνομα, Διεύθυνση και Πόλη θα περιέχουν μέχρι και 255 χαρακτήρες.

Ο πίνακας χωρίς δεδομένα θα μοιάζει κάπως έτσι:

PersonID	LastName	FirstName	Address	City

Πίνακας 49– *Κενός πίνακας από παράδειγμα CREATE TABLE*
*(**Πηγή**: https://www.w3schools.com/sql/sql_create_table.asp)*

Μπορείτε επίσης να δημιουργήσετε έναν πίνακα χρησιμοποιώντας έναν άλλο πίνακα και επιλέγοντας ποιες στήλες θέλετε στον νέο πίνακα. Λάβετε υπόψη ότι τα δεδομένα του υπάρχοντος πίνακα θα συμπληρώσουν τις εγγραφές του νέου πίνακα.

Η σύνταξη έχει ως εξής:

```
CREATE TABLE όνομα_νέου_πίνακα AS
SELECT στήλη1, στήλη2,...
FROM όνομα_υπάρχοντος_πίνακα
WHERE ....;
```

Όπως μάθαμε στην προηγούμενη ενότητα:

- Η εντολή SELECT επιλέγει τις στήλες από τον υπάρχοντα πίνακα,
- Η εντολή FROM καθορίζει το όνομα του υπάρχοντος πίνακα, και

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Ο όρος WHERE μπορεί να χρησιμοποιηθεί αν θέλετε να επιλέξετε ένα σύνολο εγγραφών που πληρούν μια καθορισμένη συνθήκη.

Παράδειγμα:

```
CREATE TABLE ΔοκιμαστικόςΠίνακας AS
SELECT ΌνομαΠελάτη, ΌνομαΕπικοινωνίας
FROM Πελάτες;
```

Διαγραφή πίνακα στην SQL

Είναι παρόμοια με την εντολή DROP DATABASE που είδαμε νωρίτερα, όμως η εντολή DROP TABLE διαγράφει έναν υπάρχοντα πίνακα σε μια βάση δεδομένων.

Να θυμάστε ότι θα πρέπει να βεβαιωθείτε ότι δεν χρειάζεστε καμία από τις πληροφορίες που περιέχονται σε έναν πίνακα πριν τον διαγράψετε.

Σύνταξη:

```
DROP TABLE ΌνομαΠίνακα;
```

Παράδειγμα:

```
DROP TABLE Άτομα;
```

Μπορείτε επίσης να επιλέξετε να διαγράψετε τα δεδομένα που περιέχονται σε έναν πίνακα, αλλά όχι και τον ίδιο τον πίνακα.

Μπορεί να δημιουργήσατε έναν νέο πίνακα από έναν υπάρχοντα πίνακα που έχει τη δομή που θέλετε, αλλά να θέλετε να προσθέσετε νέες εγγραφές εξ ολοκλήρου. Σε αυτή την περίπτωση, μπορείτε να χρησιμοποιήσετε την εντολή TRUNCATE TABLE.

Σύνταξη:

```
TRUNCATE TABLE ΌνομαΠίνακα;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παράδειγμα:

```
TRUNCATE TABLE Άτομα;
```

Τροποποίηση πίνακα στην SQL (ALTER TABLE)

Η λειτουργία ALTER TABLE μπορεί να προσθέσει, να διαγράψει ή να τροποποιήσει στήλες σε έναν υπάρχοντα πίνακα. Επίσης, μπορεί να χρησιμοποιηθεί για την προσθήκη και την διαγραφή περιορισμών σε έναν υπάρχοντα πίνακα.

Ας δούμε πρώτα τη σύνταξη της προσθήκης μιας στήλης:

```
ALTER TABLE ΌνομαΠίνακα
ADD όνομα_στήλης τύπος δεδομένων;
```

Αυτή η λειτουργία μοιάζει με τη λειτουργία που χρησιμοποιήσαμε για να δημιουργήσουμε έναν πίνακα, καθώς πρέπει να καθορίσουμε το όνομα της στήλης και τον τύπο των δεδομένων που θα περιέχονται σε αυτή την στήλη.

Παράδειγμα:

```
ALTER TABLE Πελάτες
ADD Email varchar(255);
```

Για να διαγράψουμε μια στήλη σε έναν πίνακα, όπως έχουμε δει στο παρελθόν, χρησιμοποιούμε την εντολή DROP.

Σημειώστε ότι ορισμένα συστήματα βάσεων δεδομένων δεν επιτρέπουν στους χρήστες να διαγράψουν μια στήλη.

Σύνταξη:

```
ALTER TABLE ΌνομαΠίνακα
DROP COLUMN ΌνομαΣτήλης;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για παράδειγμα, ας διαγράψουμε τη στήλη που δημιουργήσαμε:

```
ALTER TABLE Πελάτες
DROP COLUMN Email;
```

Για να αλλάξετε τον τύπο δεδομένων μιας στήλης, μπορείτε να χρησιμοποιήσετε τις ακόλουθες λειτουργίες ανάλογα με το Σχεσιακό Σύστημα Διαχείρισης Βάσεων Δεδομένων που χρησιμοποιείτε:

- ALTER COLUMN (για SQL Server/MS Access);
- MODIFY COLUMN (για My SQL/ Oracle πριν την έκδοση 10G);
- MODIFY (για την έκδοση 10G Oracle και μεταγενέστερες εκδόσεις).

Σύνταξη:

```
ALTER TABLE ΌνομαΠίνακα
ALTER COLUMN ΌνομαΣτήλης τύποςδεδομένων;
```

Σημειώστε ότι η δεύτερη εντολή είναι αυτή που αλλάζει σύμφωνα με το ΣΣΔΒΔ που χρησιμοποιείτε, μεταξύ ALTER COLUMN, MODIFY COLUMN και MODIFY. Τα υπόλοιπα παραμένουν τα ίδια.

Ας δούμε ένα παράδειγμα για να κατανοήσουμε καλύτερα αυτή την εντολή. Ο παρακάτω πίνακας είναι ο πίνακας «Άτομα» και περιέχει πληροφορίες για διαφορετικούς ανθρώπους.

ID	LastName	FirstName	Address	City
1	Hansen	Ola	Timoteivn 10	Sandnes
2	Svendson	Tove	Borgvn 23	Sandnes
3	Pettersen	Kari	Storgt 20	Stavanger

Πίνακας 50– *Πίνακας από παράδειγμα τροποποίησης (ALTER TABLE)*

*(**Πηγή**: https://www.w3schools.com/sql/sql_alter.asp)*

Για παράδειγμα, ας πούμε ότι θέλαμε να προσθέσουμε μια στήλη με το όνομα «ΗμερομηνίαΓέννησης» σε αυτόν τον πίνακα. Θα χρησιμοποιήσουμε την ακόλουθη εντολή:

```
ALTER TABLE Άτομα
ADD ΗμερομηνίαΓέννησης ημερομηνία;
```

Η νέα στήλη που προσθέσαμε στον πίνακα έχει τον τύπο δεδομένων της ημερομηνίας, που σημαίνει ότι αποθηκεύει δεδομένα σε μορφή ημερομηνίας. Παρακάτω, μπορείτε να δείτε τον πίνακα με την προσθήκη της νέας στήλης.

ID	LastName	FirstName	Address	City	DateOfBirth
1	Hansen	Ola	Timoteivn 10	Sandnes	
2	Svendson	Tove	Borgvn 23	Sandnes	
3	Pettersen	Kari	Storgt 20	Stavanger	

Πίνακας 51– *Παράδειγμα τροποποίησης πίνακα (ALTER TABLE)*
*(**Πηγή**: https://www.w3schools.com/sql/sql_alter.asp)*

Ωστόσο, αν αλλάξατε γνώμη και θέλετε να αλλάξετε τον τύπο δεδομένων της νέας στήλης, τότε μπορείτε να χρησιμοποιήσετε τη λειτουργία ALTER COLUMN.

Για παράδειγμα, μπορούμε να αλλάξουμε τον τύπο της στήλης που προστέθηκε πρόσφατα από ημερομηνία σε έτος. Για να το κάνετε αυτό, χρησιμοποιήστε την ακόλουθη πρόταση:

```
ALTER TABLE Άτομα
ALTER COLUMN ΗμερομηνίαΓέννησης Έτος;
```

Ο τύπος δεδομένων έτους περιέχει ένα έτος σε διψήφιο ή τετραψήφιο μορφότυπο.

Για να διαγράψουμε τη στήλη που μόλις τροποποιήσαμε, χρησιμοποιούμε την εντολή DROP COLUMN.

```
ALTER TABLE Άτομα
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
DROP COLUMN ΗμερομηνίαΓέννησης;
```

Ο πίνακας μας θα πάρει τη μορφή που είχε στην αρχή.

ID	LastName	FirstName	Address	City
1	Hansen	Ola	Timoteivn 10	Sandnes
2	Svendson	Tove	Borgvn 23	Sandnes
3	Pettersen	Kari	Storgt 20	Stavanger

Πίνακας 52– *Παράδειγμα τροποποίησης πίνακα (ALTER TABLE)*

(***Πηγή***: https://www.w3schools.com/sql/sql_alter.asp)

Περιορισμοί στην SQL (Constraints)

Οι περιορισμοί SQL χρησιμοποιούνται όταν ο πίνακας δημιουργείται με την εντολή CREATE TABLE ή την εντολή ALTER TABLE κατά την τροποποίησή του.

Σύνταξη:

```
CREATE TABLE όνομα_πίνακα(
        στήλη1 τύποςδεδομένων περιορισμός,
        στήλη2 τύποςδεδομένων περιορισμός,
        στήλη3 τύποςδεδομένων περιορισμός,
        ....
);
```

Οι περιορισμοί χρησιμοποιούνται για τον καθορισμό ενός συνόλου κανόνων και περιορισμών που ισχύουν για μια στήλη ή έναν πίνακα. Χρησιμοποιούνται για τη διασφάλιση της ακεραιότητας, της ακρίβειας και της αξιοπιστίας των δεδομένων. Εάν εφαρμόζονται περιορισμοί σε έναν πίνακα, τότε όλες οι στήλες πρέπει να διαμορφώνονται με βάση αυτούς τους περιορισμούς.

Οι ακόλουθοι περιορισμοί είναι αυτοί που χρησιμοποιούνται τις περισσότερες φορές:

- NOT NULL

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- UNIQUE
- PRIMARY KEY
- FOREIGN KEY
- CHECK
- DEFAULT
- CREATE INDEX

Θα εξετάσουμε καθέναν από αυτούς τους περιορισμούς για να εξηγήσουμε τη χρήση και τη σύνταξή τους με παραδείγματα.

Περιορισμοί ύπαρξης τιμής στην SQL (NOT NULL)

Στην SQL, οι στήλες μπορούν να έχουν κενές τιμές από προεπιλογή. Ο περιορισμός NOT NULL χρησιμοποιείται για την αποφυγή των κενών τιμών σε στήλες. Αυτό είναι ιδιαίτερα σημαντικό για να εξασφαλιστεί ότι, όταν προστίθεται μια νέα εγγραφή σε έναν πίνακα, συμπληρώνονται όλα τα απαραίτητα πεδία.

Για παράδειγμα, ας πούμε ότι θέλουμε να δημιουργήσουμε έναν πίνακα με το όνομα «Άτομα» και θέλουμε να διασφαλίσουμε ότι οι στήλες «ΚωδΑτόμου», «Επίθετο» και «Όνομα» δεν θα περιέχουν κενές τιμές:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255) NOT NULL,
        Ηλικία int
);
```

Εάν για κάποιο λόγο, θέλετε να τροποποιήσετε έναν ήδη υπάρχοντα πίνακα για να προσθέσετε περιορισμούς, μπορείτε να χρησιμοποιήσετε την ακόλουθη εντολή:

```
ALTER TABLE Άτομα
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
MODIFY Ηλικία int NOT NULL;
```

Μοναδικές τιμές στην SQL (Unique)

Ο περιορισμός UNIQUE χρησιμοποιείται για να διασφαλιστεί ότι όλες οι τιμές που αποθηκεύονται σε μια στήλη είναι μοναδικές μεταξύ των σειρών ενός πίνακα. Για να γίνει αυτό σαφέστερο, σκεφτείτε τη μεταβλητή «ΚωδΑτόμου». Για να μην έχουν δύο άτομα την ίδια ταυτότητα, χρησιμοποιούμε τον περιορισμό UNIQUE.

SQL Server / Oracle / MS Access:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL UNIQUE,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int
);
```

My SQL:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
UNQUE (Ταυτότητα)
);
```

Όπως βλέπετε υπάρχουν προσαρμογές σχετικά με το που τοποθετείται ο περιορισμός UNIQUE στον κώδικα, ανάλογα με το ΣΣΔΒΔ που χρησιμοποιείτε.

Εάν θέλετε να ονομάσετε ή να θέσετε έναν περιορισμό UNIQUE σε πολλαπλές στήλες, χρησιμοποιήστε τα ακόλουθα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
CONSTRAINT ΜΠ_Άτομα UNIQUE (Ταυτότητα, Επίθετο)
);
```

Μπορείτε επίσης να προσθέσετε έναν περιορισμό UNIQUE μετά τη δημιουργία του πίνακα χρησιμοποιώντας την εντολή ALTER TABLE που μάθαμε νωρίτερα.

MySQL / SQL Server / Oracle / MS Access:

```
ALTER TABLE Άτομα
ADD UNIQUE (ΚωδΑτόμου);
```

Εάν θέλετε επίσης να ονομάσετε και να θέσετε έναν περιορισμό UNIQUE σε πολλές ήδη υπάρχουσες στήλες, χρησιμοποιείτε την ακόλουθη εντολή:

```
ALTER TABLE Άτομα
ADD CONSTRAINT ΜΠ_Άτομα UNIQUE (ΚωδΑτόμου, Επίθετο);
```

Για να διαγράψετε τον περιορισμό UNIQUE, μπορείτε να χρησιμοποιήσετε την ακόλουθη εντολή:

My SQL:

```
ALTER TABLE Άτομα
DROP INDEX ΜΠ_Άτομα;
```

SQL Server/Oracle/ MS Access:

```
ALTER TABLE Άτομα
DROP CONSTRAINT ΜΠ_Άτομα;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πρωτεύον Κλειδί στην SQL (Primary Key)

Ο περιορισμός PRIMARY KEY χρησιμοποιείται για τη μοναδική αναγνώριση κάθε σειράς ή εγγραφής σε έναν πίνακα. Σημειώστε ότι οι περιορισμοί πρέπει να περιέχουν μοναδικές τιμές, ωστόσο δεν μπορούν να περιέχουν κενές τιμές.

Ένας πίνακας μπορεί να έχει μόνο ΕΝΑ πρωτεύον κλειδί το οποίο αυτό μπορεί να αποτελείται από μία ή περισσότερες στήλες.

SQL Server / Oracle / MS Access:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL PRIMARY KEY,
        Επίθετο varchar(255) NOT NULL,
        Όνομα archar(255),
        Ηλικία int
);
```

MySQL:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
    PRIMARY KEY (Ταυτότητα)
);
```

Το ακόλουθο παράδειγμα σας επιτρέπει να ονομάσετε και να θέσετε έναν περιορισμό PRIMARY KEY σε πολλαπλές στήλες:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
    CONSTRAINT ΠΚ_Άτομα PRIMARY KEY (Ταυότητα,Επίθετο)
);
```

Σημειώστε ότι υπάρχει ένα μόνο PRIMARY KEY, ωστόσο η τιμή του περιλαμβάνει δύο στήλες.

Μπορείτε επίσης να δημιουργήσετε έναν περιορισμό PRIMARY KEY σε έναν υπάρχοντα πίνακα χρησιμοποιώντας την ακόλουθη εντολή:

```
ALTER TABLE Άτομα
ADD PRIMARY KEY (ΚωδΑτόμου);
```

Για να προσθέσετε και να θέσετε έναν περιορισμό PRIMARY KEY σε έναν υπάρχοντα πίνακα, χρησιμοποιήστε την ακόλουθη εντολή:

```
ALTER TABLE Άτομα
ADD CONSTRAINT ΠΚ_Άτομα PRIMARY KEY (ΚωδΑτόμου, Επίθετο);
```

Για να διαγράψετε έναν περιορισμό PRIMARY KEY, χρησιμοποιήστε τις ακόλουθες εντολές σύμφωνα με το ΣΣΔΒΔ που χρησιμοποιείτε.

MySQL:

```
ALTER TABLE Άτομα
DROP PRIMARY KEY;
```

SQL Server / Oracle / MS Access:

```
ALTER TABLE Άτομα
DROP PRIMARY KEY;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ξένο Κλειδί στην SQL (Foreign Key)

Το FOREING KEY αντιπροσωπεύει τις στήλες ενός πίνακα που συνδέονται με έναν περιορισμό PRIMARY KEY σε έναν άλλο πίνακα. Ο πίνακας που έχει τον περιορισμό FOREIGN KEY ονομάζεται child table, ενώ ο πίνακας που έχει το primary key ονομάζεται πίνακας αναφοράς ή parent table.

Αυτός ο τύπος περιορισμού χρησιμοποιείται για να αποτρέψει οποιεσδήποτε ενέργειες που θα κατέστρεφαν τους δεσμούς μεταξύ του parent table και του child table.

Ας εξετάσουμε τους δύο παρακάτω πίνακες:

PersonID	LastName	FirstName	Age
1	Hansen	Ola	30
2	Svendson	Tove	23
3	Pettersen	Kari	20

Πίνακας 53– *Πίνακας Ατόμων από παράδειγμα περιορισμού FOREING KEY*
*(**Πηγή**: https://www.w3schools.com/sql/sql_foreignkey.asp)*

OrderID	OrderNumber	PersonID
1	77895	3
2	44678	3
3	22456	2
4	24562	1

Πίνακας 54– *Πίνακας Παραγγελιών από παράδειγμα περιορισμού FOREING KEY*
*(**Πηγή**: https://www.w3schools.com/sql/sql_foreignkey.asp)*

Αυτοί οι δύο πίνακες συνδέονται με τη στήλη «ΑναγνωριστικόΑτόμου» (PersonID) που βρίσκεται και στους δύο πίνακες. Τώρα, το primary key βρίσκεται στον πίνακα ΄Ατομα και το foreign key στη στήλη «ΑναγνωριστικόΑτόμου» στον πίνακα Παραγγελίες.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο περιορισμός FOREIGN KEY λειτουργεί εμποδίζοντας την εισαγωγή μη έγκυρων δεδομένων στη στήλη του FOREIGN KEY επειδή συνδέεται με τον parent table και οι τιμές του πρέπει να είναι πανομοιότυπες.

Για να χρησιμοποιήσετε τον περιορισμό FOREIGN KEY κατά τη δημιουργία ενός πίνακα, μπορείτε να χρησιμοποιήσετε την ακόλουθη εντολή ανάλογα με το ΣΣΔΒΔ που χρησιμοποιείτε.

SQL Server / Oracle / MS Access:

```
CREATE TABLE Παραγγελίες (
        ΚωδικόςΠαραγγελίας int NOT NULL PRIMARY KEY,
        ΑριθμόςΠαραγγελίας int NOT NULL,
        ΚωδΑτόμου int FOREIGN KEY REFERENCES Άτομα(ΚωδΑτόμου)
);
```

My SQL:

```
CREATE TABLE Παραγγελίες (
        ΚωδικόςΠαραγγελίας int NOT NULL,
        ΑριθμόςΠαραγγελίας int NOT NULL,
        ΚωδΑτόμου int,
        PRIMARY KEY (ΚωδικόςΠαραγγελίας),
        FOREIGN KEY (ΚωδΑτόμου) REFERENCES Άτομα(ΚωδΑτόμου)
);
```

Αυτή η εντολή συνέδεσε τον πίνακα Παραγγελιών με τον πίνακα Ατόμων χρησιμοποιώντας τον περιορισμό FOREIGN KEY με βάση τη στήλη ΚωδΑτόμου.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Περιορισμός ελέγχου τιμής στην SQL (CHECK)

Ο περιορισμός CHECK χρησιμοποιείται για να καθορίσει τις τιμές που επιτρέπονται σε μια στήλη ή σε ορισμένες στήλες ενός πίνακα με βάση τις τιμές που βρίσκονται σε άλλες στήλες της ίδιας σειράς.

Παράδειγμα περιορισμού CHECK στη λειτουργία CREATE TABLE

Το ακόλουθο παράδειγμα χρησιμοποιείται για να διασφαλιστεί ότι ένα άτομο δεν είναι κάτω από την ηλικία των 18 ετών, προσθέτοντας τον περιορισμό CHECK στη στήλη «Ηλικία».

MySQL:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
    CHECK (Age>=18)
);
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int CHECK (Age>=18)
);
```

Εάν θέλετε να καθορίσετε έναν περιορισμό CHECK και να χρησιμοποιήσετε τον περιορισμό σε πολλαπλές στήλες, μπορείτε να χρησιμοποιήσετε την ακόλουθη εντολή:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

MySQL / SQL Server / Oracle / MS Access:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
        Πόλη varchar(255),
        CONSTRAINT ΕΛΓΧ_Άτομο CHECK (Ηλικία>=18 AND Πόλη= 'Σάντνες')
);
```

Παράδειγμα περιορισμού CHECK στην εντολή ALTER TABLE

Για να καθορίσετε έναν περιορισμό σε έναν ήδη υπάρχοντα πίνακα, χρησιμοποιήστε την ακόλουθη εντολή.

MySQL / SQL Server / Oracle / MS Access:

```
ALTER TABLE Άτομα
ADD CHECK (Ηλικία>=18);
```

Για να καθορίσετε έναν περιορισμό και να τον προσθέσετε σε πολλαπλές στήλες, μπορείτε να χρησιμοποιήσετε:

```
ALTER TABLE Άτομα
ADD CONSTRAINT ΕΛΓΧ_Άτομα CHECK (Ηλικία>=18 AND Πόλη= 'Σάντνες');
```

Παράδειγμα εντολής DROP ενός περιορισμού CHECK

Για να διαγράψετε έναν περιορισμό CHECK, μπορείτε να χρησιμοποιήσετε τα ακόλουθα σύμφωνα με το ΣΣΔΒΔ που χρησιμοποιείτε.

SQL Server / Oracle / MS Access:

```
ALTER TABLE Άτομα
DROP CONSTRAINT ΕΛΓΧ_ΗλικίαΑτόμου;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

MySQL:

```
ALTER TABLE Άτομα
DROP CHECK ΕΛΓΧ_ΗλικίαΑτόμου;
```

Περιορισμός προεπιλεγμένων τιμών στην SQL (DEFAULT)

Ο περιορισμός DEFAULT χρησιμοποιείται για να καθορίσει μια προεπιλεγμένη τιμή για μια στήλη. Εάν δεν έχουν καθοριστεί άλλες τιμές, η προεπιλεγμένη τιμή θα προστεθεί σε όλες τις νέες εγγραφές.

Παράδειγμα περιορισμού DEFAULT στη λειτουργία CREATE TABLE

Το ακόλουθο παράδειγμα προσθέτει μια προεπιλεγμένη τιμή στη στήλη Πόλη όταν δημιουργείται ο πίνακας Άτομα:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL,
        Επίθετο varchar(255) NOT NULL,
        Όνομα varchar(255),
        Ηλικία int,
        Πόλη varchar(255) DEFAULT ‘Σάντνες'
);
```

Αυτός ο περιορισμός μπορεί επίσης να χρησιμοποιηθεί για την εισαγωγή τιμών με τελεστές όπως το GETDATE():

```
CREATE TABLE Παραγγελίες (
        ΚωδικόςΠαραγγελίας int NOT NULL,
        ΑρΠαραγγελίας int NOT NULL,
        ΗμερομηνίαΠαραγγελίας ημερομηνία DEFAULT GETDATE()
);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Παράδειγμα περιορισμού DEFAULT με την εντολή ALTER TABLE

Σε αυτό το παράδειγμα, η στήλη «Πόλη» χρησιμοποιείται για τον καθορισμό ενός περιορισμού DEFAULT όταν τροποποιούμε έναν ήδη υπάρχοντα πίνακα.

MySQL:

```
ALTER TABLE Άτομα
ALTER Πόλη SET DEFAULT ‘Σάντνες';
```

SQL Server:

```
ALTER TABLE Άτομα
ADD CONSTRAINT προεπ_Πόλη
DEFAULT ‘Σάντνες' FOR Πόλη;
```

MS Access:

```
ALTER TABLE Άτομα
ALTER COLUMN ΄Πόλη SET DEFAULT ‘Σάντνες’;
```

Oracle:

```
ALTER TABLE Άτομα
MODIFY Πόλη DEFAULT ‘Σάντνες';
```

Παράδειγμα κατάργησης ενός περιορισμού DEFAULT με την εντολή DROP

MySQL:

```
ALTER TABLE Άτομα
ALTER Πόλη DROP DEFAULT;
```

SQL Server / Oracle / MS Access:

```
ALTER TABLE Άτομα
ALTER COLUMN Πόλη DROP DEFAULT;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Δείκτες στην SQL (Index)

Η εντολή CREATE INDEX δημιουργεί έναν δείκτη σε έναν πίνακα. Οι δείκτες είναι χρήσιμοι όταν θέλετε να ανακτήσετε δεδομένα πιο γρήγορα.

Σημειώστε ότι οι πίνακες με δείκτες χρειάζονται περισσότερο χρόνο για να ενημερωθούν σε σύγκριση με τους πίνακες χωρίς δείκτες. Ως εκ τούτου, προτείνεται να δημιουργηθούν μόνο δείκτες σε στήλες όπου αναζητούνται δεδομένα συχνά.

Για την εντολή CREATE INDEX σε έναν πίνακα όπου επιτρέπονται διπλές τιμές, χρησιμοποιήστε την ακόλουθη σύνταξη:

```
CREATE INDEX όνομα_δείκτη
ON όνομα_πίνακα (στήλη1, στήλη2, ...);
```

Για την εντολή CREATE UNIQUE INDEX σε έναν πίνακα όπου επιτρέπονται διπλές τιμές, χρησιμοποιήστε την ακόλουθη σύνταξη:

```
CREATE UNIQUE INDEX όνομα_δείκτη
ON όνομα_πίνακα (στήλη1, στήλη2, ...);
```

Λάβετε υπόψη ότι η δημιουργία δεικτών ποικίλλει από βάση δεδομένων σε βάση δεδομένων, οπότε πάντα να ελέγχετε τη σύνταξη όταν δημιουργείτε έναν δείκτη στη βάση δεδομένων σας.

Παραδείγματα της εντολής CREATE INDEX

Σε αυτό το παράδειγμα, δημιουργούμε έναν δείκτη στη στήλη Επίθετο καθορίζοντας το όνομα δκτς_επίθετο:

```
CREATE INDEX δκτς_επίθετο
ON Άτομα (Επίθετο);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για να δημιουργήσετε έναν δείκτη σε ένα συνδυασμό στηλών, χρησιμοποιήστε την ακόλουθη εντολή:

```
CREATE INDEX δκτς_όνομα
ON Άτομα (Επίθετο, Όνομα);
```

Αν θέλετε, μπορείτε να προσθέσετε περισσότερες στήλες στην παρένθεση.

Παραδείγματα της εντολής DROP INDEX για τη διαγραφή ενός δείκτη

Εάν θέλετε να διαγράψετε έναν δείκτη, χρησιμοποιήστε την ακόλουθη εντολή σύμφωνα με το ΣΣΔΒΔ που χρησιμοποιείτε.

MS Access:

```
DROP INDEX όνομα_δείκτη ON όνομα_πίνακα;
```

SQL Server:

```
DROP INDEX όνομα_πίνακα.όνομα_δείκτη;
```

DB2/Oracle:

```
DROP INDEX όνομα_δείκτη;
```

MySQL:

```
ALTER TABLE όνομα_πίνακα
DROP INDEX όνομα_δείκτη;
```

Αυτόματη Αύξηση Τιμής στην SQL (Auto-Increment)

Η αυτόματη αύξηση χρησιμοποιείται για την αυτόματη δημιουργία μοναδικών αριθμών όταν εισάγεται μια νέα εγγραφή σε έναν πίνακα. Αυτό χρησιμοποιείται συνήθως σαν Primary key προκειμένου να διασφαλιστεί ότι κανένα άτομο δεν έχει την ίδια ταυτότητα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αυτή η λειτουργία χρησιμοποιεί διαφορετική σύνταξη στα συστήματα MySQL, SQL Server, Access και Oracle. Ως εκ τούτου, θα εξετάσουμε καθένα από αυτά για να εξηγήσουμε πώς να χρησιμοποιήσετε την αυτόματη αύξηση τιμών.

MySQL:

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int NOT NULL AUTO_INCREMENT,
         Επίθετο varchar(255) NOT NULL,
         Όνομα varchar(255),
         Ηλικία int,
        PRIMARY KEY (ΚωδΑτόμου)
);
```

Στο σύστημα MySQL, η εντολή AUTO_INCREMENT προσθέτει τη λειτουργία αυτόματης αύξησης και η τιμή ορίζεται 1 από προεπιλογή και αυξάνεται κατά 1 κάθε φορά.

Εάν θέλετε η ακολουθία να ξεκινήσει από διαφορετική τιμή, χρησιμοποιήστε την ακόλουθη εντολή:

```
ALTER TABLE Άτομα AUTO_INCREMENT=100;
```

Εάν εισαγάγετε μια νέα εγγραφή στον πίνακα «Άτομα», δεν θα χρειαστεί να καθορίσετε μια τιμή για τη στήλη «ΚωδΑτόμου», καθώς θα δημιουργηθεί αυτόματα:

```
INSERT INTO Άτομα (Όνομα, Επίθετο)
VALUES (‘Λαρς','Μόνσεν');
```

SQL Server

Ακολουθούμε το ίδιο παράδειγμα όπως είδαμε στις προηγούμενες διαφάνειες για το MySQL, όπου χρησιμοποιούμε τη στήλη «ΚωδΑτόμου» ως το primary key στον πίνακα «Άτομα»:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
CREATE TABLE Άτομα (
        ΚωδΑτόμου int IDENTITY(1,1) PRIMARY KEY,
         Επίθετο varchar(255) NOT NULL,
         Όνομα varchar(255),
         Ηλικία int
);
```

Στο σύστημα SQL Server, η εντολή της αυτόματης αύξησης τιμών χρησιμοποιεί τη λέξη-κλειδί IDENTIFY για να ενεργοποιηθεί. Οι δύο τιμές στην παρένθεση υποδεικνύουν (την αρχική τιμή, προσθέτοντας μια νέα τιμή για κάθε νέα εγγραφή). Ξεκινά από το 1 και αυξάνεται κατά 1 κάθε φορά που εισάγεται μια νέα εγγραφή.

Σε περίπτωση που θέλετε να αλλάξετε την αρχική τιμή σε 10 και να προσθέτετε 5 κάθε φορά που προστίθεται μια νέα εγγραφή, θα πρέπει να την γράψετε ως εξής: IDENTIFY (10,5).

Κατά την εισαγωγή νέων εγγραφών, δεν χρειάζεται να καθορίσετε την τιμή στη στήλη ΚωδΑτόμου. Θα εμφανιστεί αυτόματα όπως στο παραπάνω παράδειγμα.

MS Access

```
CREATE TABLE Άτομα (
ΚωδΑτόμου AUTOINCREMENT PRIMARY KEY,
         Επίθετο varchar(255) NOT NULL,
         Όνομα varchar(255),
         Ηλικία int
);
```

Το MS Access χρησιμοποιεί τη λέξη-κλειδί AUTOINCREMENT για να ενεργοποιήσει τη λειτουργία αυτόματης αύξησης τιμών. Η αρχική τιμή είναι το 1 και αυξάνεται κατά 1 κάθε φορά που προστίθεται μια νέα εγγραφή, όπως και στις άλλες δύο περιπτώσεις.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μπορείτε να καθορίσετε διαφορετικές τιμές όπως 10 για την αρχική τιμή και να αυξάνεται κατά 5 με κάθε προσθήκη με την εντολή AUTOINCREMENT(10,5).

Και πάλι, σημειώστε ότι κάθε φορά που προσθέτουμε μια νέα εγγραφή, δεν χρειάζεται να καθορίσουμε τον Κωδικό Ατόμου. Εμφανίζεται αυτόματα.

Oracle

Στο σύστημα Oracle, ο κώδικας για την αυτόματη αύξηση τιμών είναι λίγο πιο περίπλοκος. Για να δημιουργήσετε ένα πεδίο αυτόματης αύξησης, πρέπει να δημιουργήσετε μια ακολουθία αριθμών:

```
CREATE SEQUENCE ακ_ατόμου
MINVALUE 1
START WITH 1
INCREMENT BY 1
CACHE 10;
```

Αυτή η ακολουθία δημιουργεί ένα αντικείμενο ακολουθίας που ονομάζεται «ακ_ατόμου», ορίζει την ελάχιστη τιμή από την οποία ξεκινά (η οποία είναι 1 σε αυτήν την περίπτωση), έπειτα καθορίζει την αύξηση κατά 1. Η προσωρινή μνήμη καθορίζει πόσες τιμές ακολουθίας θα πρέπει να αποθηκεύονται στη μνήμη για γρηγορότερη πρόσβαση.

Σε αντίθεση με τα προηγούμενα παραδείγματα, για να εισαγάγετε μια νέα εγγραφή στον πίνακα Άτομα, θα πρέπει να χρησιμοποιήσετε τη λειτουργία nextval. Αυτή η λειτουργία χρησιμοποιείται για να ανακτήσει την επόμενη τιμή από το αντικείμενο ακολουθίας που δημιουργήσαμε.

```
INSERT INTO Άτομα (ΚωδΑτόμου, Όνομα, Επίθετο)
VALUES (ακ_ατόμου.nextval,'Λαρς','Μόνσεν');
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εδώ, μπορούμε να δούμε ότι η στήλη ΚωδΑτόμου επιλέγεται για να εκχωρηθεί ο επόμενος αριθμός από το αντικείμενο ακολουθίας που δημιουργήσαμε που ονομάζεται «ακ_ατόμου».

Ημερομηνίες στην SQL

Μια από τις μεγαλύτερες προκλήσεις όταν εργάζεστε με ημερομηνίες (date) είναι να διασφαλίσετε ότι η μορφή της ημερομηνίας που προσπαθείτε να εισάγετε είναι η ίδια με τη μορφή της στήλης ημερομηνίας στη βάση δεδομένων.

Είναι σημαντικό να σημειωθεί ότι τα δεδομένα που περιέχουν μόνο ημερομηνίες θα λειτουργήσουν όπως αναμένεται στα ερωτήματα. Ωστόσο, αν συμπεριλαμβάνεται και η ώρα (time), τα πράγματα γίνονται λίγο πιο περίπλοκα.

Τύποι χρονολογικών δεδομένων στο MySQL:

- DATE (Ημερομηνία) - μορφότυπο ΕΕΕΕ-ΜΜ-ΗΗ
- DATETIME (Ημερομηνία και Ώρα) - μορφότυπο: ΕΕΕΕ-ΜΜ-ΗΗ ΩΩ:ΛΛ:ΔΔ
- TIMESTAMP (Χρονοσήμανση) - μορφή: ΕΕΕΕ-ΜΜ-ΗΗ ΩΩ:ΛΛ:ΔΔ
- YEAR (Έτος) - Μορφή ΕΕΕΕ ή ΕΕ

Τύποι χρονολογικών δεδομένων στο SQL Server:

- DATE – μορφή: ΕΕΕΕ-ΜΜ-ΗΗ
- DATETIME- μορφή: ΕΕΕΕ-ΜΜ-ΗΗ ΩΩ: ΛΛ:ΔΔ
- SMALLDATETIME - μορφή: ΕΕΕΕ-ΜΜ-ΗΗ ΩΩ: ΛΛ:ΔΔ
- TIMESTAMP - μορφή: ένας μοναδικός αριθμός

Λάβετε υπόψη ότι οι τύποι δεδομένων επιλέγονται όταν δημιουργείτε έναν νέο πίνακα στη βάση δεδομένων σας.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

OrderId	ProductName	OrderDate
1	Geitost	2008-11-11
2	Camembert Pierrot	2008-11-09
3	Mozzarella di Giovanni	2008-11-11
4	Mascarpone Fabioli	2008-10-29

Πίνακας 55– *Πίνακας Παραγγελιών από παράδειγμα ημερομηνιών*
(***Πηγή***: https://www.w3schools.com/sql/sql_dates.asp)

Θα χρησιμοποιήσουμε τον πίνακα Παραγγελιών στο παράδειγμά μας για να επιλέξουμε τις εγγραφές με ημερομηνία παραγγελίας “2008-11-11”.

Παράδειγμα:

```
SELECT *
FROM Παραγγελίες
WHERE ΗμερομηνίαΠαραγγελίας='2008-11-11';
```

Το αναμενόμενο αποτέλεσμα θα είναι κάπως έτσι:

OrderId	ProductName	OrderDate
1	Geitost	2008-11-11
3	Mozzarella di Giovanni	2008-11-11

Πίνακας 56– *Αποτέλεσμα ερωτήματος Ημερομηνιών Παραγγελίας από παράδειγμα ημερομηνιών*
(***Πηγή***: https://www.w3schools.com/sql/sql_dates.asp)

Σημειώστε ότι δύο ημερομηνίες μπορούν να συγκριθούν εύκολα όταν δεν υπάρχει Χρονοσήμανση (TIMESTAMP).

Ας υποθέσουμε ότι έχετε τον πίνακα Παραγγελιών, αλλά με μια χρονική σήμανση στη στήλη Ημερομηνία Παραγγελίεας (OrderDate).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

OrderId	ProductName	OrderDate
1	Geitost	2008-11-11 13:23:44
2	Camembert Pierrot	2008-11-09 15:45:21
3	Mozzarella di Giovanni	2008-11-11 11:12:01
4	Mascarpone Fabioli	2008-10-29 14:56:59

Πίνακας 57– *Πίνακας παραγγελιών με χρονοσήμανση από παράδειγμα Ημερομηνιών (**Πηγή**: https://www.w3schools.com/sql/sql_dates.asp)*

Εδώ, εάν προσπαθήσατε να χρησιμοποιήσετε το ίδιο ερώτημα με αυτό που χρησιμοποιήσαμε παραπάνω θα προκύψει το παρακάτω:

```
SELECT *
FROM Παραγγελίες
WHERE Ημερομηνία Παραγγελίας='2008-11-11';
```

Δεν θα πάρετε κανένα αποτέλεσμα, επειδή το ερώτημα δεν λαμβάνει υπόψη τη χρονοσήμανση. Συνιστάται να μην χρησιμοποιείτε χρονοσημάνσεις, εκτός αν είναι απολύτως απαραίτητο.

Προβολές στην SQL (VIEWS)

Στην SQL, μια προβολή (view) είναι ένας εικονικός πίνακας του συνόλου αποτελεσμάτων που δημιουργείται από ένα συγκεκριμένο ερώτημα. Μια προβολή είναι χρήσιμη όταν θέλετε να προβάλετε και να παρουσιάσετε δεδομένα μέσω ενός συνδυασμού πινάκων.

Σύνταξη:

```
CREATE VIEW όνομα_προβολής AS
SELECT στήλη1, στήλη2, ...
FROM όνομα_πίνακα
WHERE συνθήκη;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σημειώστε ότι μια προβολή εμφανίζει πάντα ενημερωμένα δεδομένα, καθώς η βάση δεδομένων αναδημιουργεί τον εικονικό πίνακα, κάθε φορά που οι χρήστες δημιουργούν το αντίστοιχο ερώτημα.

Παράδειγμα για να δημιουργήσετε ερώτημα με όλους τους πελάτες από τη Βραζιλία:

```
CREATE VIEW [Πελάτες από Βαζιλία] AS
SELECT ΌνομαΠελάτη, ΌνομαΕπικοινωνίας
FROM Πελάτες
WHERE Country = ‘Βραζιλία';
```

Για να προβάλετε το ερώτημα:

```
SELECT * FROM [Πελάτες από Βραζιλία];
```

Παράδειγμα δημιουργίας προβολής που επιλέγει κάθε προϊόν στον πίνακα Προϊόντων με τιμή υψηλότερη από τη μέση τιμή:

```
CREATE VIEW [Προϊόντα με υψηλότερη τιμή από τη μέση τιμή] AS
SELECT ΌνομαΠροϊόντος, Τιμή
FROM Προϊόντα
WHERE Τιμή > (SELECT AVG(Τιμή) FROM Προϊόντα);
```

Για να υποβάλετε ένα ερώτημα σχετικά με την παραπάνω προβολή:

```
SELECT * FROM [Προϊόντα με υψηλότερη τιμή από τη μέση τιμή];
```

Για να ενημερώσετε μια προβολή, χρησιμοποιήστε την εντολή CREATE OR REPLACE VIEW:

```
CREATE OR REPLACE VIEW όνομα_προβολής AS
SELECT στήλη1, στήλη2, ...
FROM όνομα_πίνακα
WHERE συνθήκη;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για να προσθέσετε τη στήλη «Πόλη» στην προβολή με όνομα Πελάτες από Βραζιλία που δημιουργήσαμε νωρίτερα:

```
CREATE OR REPLACE VIEW [Πελάτες από Βραζιλία] AS
SELECT ΌνομαΠελάτη, ΌνομαΕπικοινωνίας, Πόλη
FROM Πελάτες
WHERE Χώρα = ‘Βραζιλία';
```

Για να διαγράψετε μια προβολή, χρησιμοποιήστε την εντολή DROP VIEW:

```
DROP VIEW όνομα_προβολής;
```

Για να διαγράψετε την προβολή «Πελάτες από Βραζιλία»:

```
DROP VIEW [Πελάτες από Βραζιλία];
```

Τύποι δεδομένων στην SQL

Γενικά, κάθε στήλη σε έναν πίνακα απαιτεί ένα όνομα και έναν τύπο δεδομένων.

Ένας προγραμματιστής της SQL θα πρέπει να αποφασίσει τον τύπο των δεδομένων που θα αποθηκευτούν μέσα σε κάθε στήλη κατά τη δημιουργία ενός πίνακα. Ο τύπος δεδομένων χρησιμοποιείται στην SQL για την κατανόηση των δεδομένων που θα περιέχονται σε κάθε στήλη και επίσης πώς θα αλληλοεπιδρά με αυτά τα δεδομένα.

Λάβετε υπόψη ότι οι τύποι δεδομένων μπορεί να έχουν διαφορετικά ονόματα σε κάθε βάση δεδομένων. Πρέπει πάντοτε να ελέγχετε την τεκμηρίωση, ακόμη και αν το όνομα είναι το ίδιο, καθώς άλλες λεπτομέρειες μπορεί να είναι διαφορετικές, όπως το μέγεθος.

Τύποι δεδομένων στο σύστημα MySQL (Έκδοση 8.0)

Το σύστημα MySQL έχει τρεις βασικούς τύπους δεδομένων: συμβολοσειρά, αριθμητικός τύπος και ημερομηνίας/ώρας.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τύποι δεδομένων συμβολοσειράς

Data type	Description
CHAR(size)	A FIXED length string (can contain letters, numbers, and special characters). The *size* parameter specifies the column length in characters - can be from 0 to 255. Default is 1
VARCHAR(size)	A VARIABLE length string (can contain letters, numbers, and special characters). The *size* parameter specifies the maximum column length in characters - can be from 0 to 65535
BINARY(size)	Equal to CHAR(), but stores binary byte strings. The *size* parameter specifies the column length in bytes. Default is 1
VARBINARY(size)	Equal to VARCHAR(), but stores binary byte strings. The *size* parameter specifies the maximum column length in bytes.
TINYBLOB	For BLOBs (Binary Large Objects). Max length: 255 bytes
TINYTEXT	Holds a string with a maximum length of 255 characters
TEXT(size)	Holds a string with a maximum length of 65,535 bytes
BLOB(size)	For BLOBs (Binary Large Objects). Holds up to 65,535 bytes of data
MEDIUMTEXT	Holds a string with a maximum length of 16,777,215 characters
MEDIUMBLOB	For BLOBs (Binary Large Objects). Holds up to 16,777,215 bytes of data
LONGTEXT	Holds a string with a maximum length of 4,294,967,295 characters
LONGBLOB	For BLOBs (Binary Large Objects). Holds up to 4,294,967,295 bytes of data
ENUM(val1, val2, val3, ...)	A string object that can have only one value, chosen from a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. The values are sorted in the order you enter them
SET(val1, val2, val3, ...)	A string object that can have 0 or more values, chosen from a list of possible values. You can list up to 64 values in a SET list

Πίνακας 58– *Τύποι δεδομένων συμβολοσειράς (**Πηγή**: https://www.w3schools.com/sql/sql_datatypes.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αριθμητικοί τύποι δεδομένων

Data type	Description
BIT(*size*)	A bit-value type. The number of bits per value is specified in *size*. The *size* parameter can hold a value from 1 to 64. The default value for *size* is 1.
TINYINT(*size*)	A very small integer. Signed range is from -128 to 127. Unsigned range is from 0 to 255. The *size* parameter specifies the maximum display width (which is 255)
BOOL	Zero is considered as false, nonzero values are considered as true.
BOOLEAN	Equal to BOOL
SMALLINT(*size*)	A small integer. Signed range is from -32768 to 32767. Unsigned range is from 0 to 65535. The *size* parameter specifies the maximum display width (which is 255)
MEDIUMINT(*size*)	A medium integer. Signed range is from -8388608 to 8388607. Unsigned range is from 0 to 16777215. The *size* parameter specifies the maximum display width (which is 255)
INT(*size*)	A medium integer. Signed range is from -2147483648 to 2147483647. Unsigned range is from 0 to 4294967295. The *size* parameter specifies the maximum display width (which is 255)
INTEGER(*size*)	Equal to INT(size)
BIGINT(*size*)	A large integer. Signed range is from -9223372036854775808 to 9223372036854775807. Unsigned range is from 0 to 18446744073709551615. The *size* parameter specifies the maximum display width (which is 255)
FLOAT(*size*, *d*)	A floating point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter. This syntax is deprecated in MySQL 8.0.17, and it will be removed in future MySQL versions
FLOAT(*p*)	A floating point number. MySQL uses the *p* value to determine whether to use FLOAT or DOUBLE for the resulting data type. If *p* is from 0 to 24, the data type becomes FLOAT(). If *p* is from 25 to 53, the data type becomes DOUBLE()
DOUBLE(*size*, *d*)	A normal-size floating point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter
DOUBLE PRECISION(*size*, *d*)	
DECIMAL(*size*, *d*)	An exact fixed-point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter. The maximum number for *size* is 65. The maximum number for *d* is 30. The default value for *size* is 10. The default value for *d* is 0.
DEC(*size*, *d*)	Equal to DECIMAL(size,d)

Πίνακας 59– *Αριθμητικοί τύποι δεδομένων (MySQL) (**Πηγή**: https://www.w3schools.com/sql/sql_datatypes.asp)*

Τύποι δεδομένων ημερομηνίας και ώρας

Data type	Description
DATE	A date. Format: YYYY-MM-DD. The supported range is from '1000-01-01' to '9999-12-31'
DATETIME(*fsp*)	A date and time combination. Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. Adding DEFAULT and ON UPDATE in the column definition to get automatic initialization and updating to the current date and time
TIMESTAMP(*fsp*)	A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC. Automatic initialization and updating to the current date and time can be specified using DEFAULT CURRENT_TIMESTAMP and ON UPDATE CURRENT_TIMESTAMP in the column definition
TIME(*fsp*)	A time. Format: hh:mm:ss. The supported range is from '-838:59:59' to '838:59:59'
YEAR	A year in four-digit format. Values allowed in four-digit format: 1901 to 2155, and 0000. MySQL 8.0 does not support year in two-digit format.

Πίνακας 60– *Τύποι δεδομένων ημερομηνίας και ώρας (MySQL)*

*(**Πηγή**: https://www.w3schools.com/sql/sql_datatypes.asp)*

Τύποι δεδομένων στο σύστημα SQL Server:

Τύποι δεδομένων συμβολοσειράς

Data type	Description	Max size	Storage
char(n)	Fixed width character string	8,000 characters	Defined width
varchar(n)	Variable width character string	8,000 characters	2 bytes + number of chars
varchar(max)	Variable width character string	1,073,741,824 characters	2 bytes + number of chars
text	Variable width character string	2GB of text data	4 bytes + number of chars
nchar	Fixed width Unicode string	4,000 characters	Defined width x 2
nvarchar	Variable width Unicode string	4,000 characters	
nvarchar(max)	Variable width Unicode string	536,870,912 characters	
ntext	Variable width Unicode string	2GB of text data	
binary(n)	Fixed width binary string	8,000 bytes	
varbinary	Variable width binary string	8,000 bytes	
varbinary(max)	Variable width binary string	2GB	
image	Variable width binary string	2GB	

Πίνακας 61– *Τύποι δεδομένων συμβολοσειράς (SQL Server)*

(***Πηγή***: https://www.w3schools.com/sql/sql_datatypes.asp)

Αριθμητικοί τύποι δεδομένων

Data type	Description	Storage
bit	Integer that can be 0, 1, or NULL	
tinyint	Allows whole numbers from 0 to 255	1 byte
smallint	Allows whole numbers between -32,768 and 32,767	2 bytes
int	Allows whole numbers between -2,147,483,648 and 2,147,483,647	4 bytes
bigint	Allows whole numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807	8 bytes
decimal(p,s)	Fixed precision and scale numbers. Allows numbers from -10^38 +1 to 10^38 –1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0	5-17 bytes
numeric(p,s)	Fixed precision and scale numbers. Allows numbers from -10^38 +1 to 10^38 –1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0	5-17 bytes
smallmoney	Monetary data from -214,748.3648 to 214,748.3647	4 bytes
money	Monetary data from -922,337,203,685,477.5808 to 922,337,203,685,477.5807	8 bytes
float(n)	Floating precision number data from -1.79E + 308 to 1.79E + 308. The n parameter indicates whether the field should hold 4 or 8 bytes. float(24) holds a 4-byte field and float(53) holds an 8-byte field. Default value of n is 53.	4 or 8 bytes
real	Floating precision number data from -3.40E + 38 to 3.40E + 38	4 bytes

Πίνακας 62– *Αριθμητικοί τύποι δεδομένων (SQL Server)*

(***Πηγή***: https://www.w3schools.com/sql/sql_datatypes.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τύποι δεδομένων ημερομηνίας και ώρας

Data type	Description	Storage
datetime	From January 1, 1753 to December 31, 9999 with an accuracy of 3.33 milliseconds	8 bytes
datetime2	From January 1, 0001 to December 31, 9999 with an accuracy of 100 nanoseconds	6-8 bytes
smalldatetime	From January 1, 1900 to June 6, 2079 with an accuracy of 1 minute	4 bytes
date	Store a date only. From January 1, 0001 to December 31, 9999	3 bytes
time	Store a time only to an accuracy of 100 nanoseconds	3-5 bytes
datetimeoffset	The same as datetime2 with the addition of a time zone offset	8-10 bytes
timestamp	Stores a unique number that gets updated every time a row gets created or modified. The timestamp value is based upon an internal clock and does not correspond to real time. Each table may have only one timestamp variable	

Πίνακας 63– *Τύποι δεδομένων ημερομηνίας και ώρας (SQL Server)*

*(**Πηγή**: https://www.w3schools.com/sql/sql_datatypes.asp)*

Άλλοι τύποι δεδομένων

Data type	Description
sql_variant	Stores up to 8,000 bytes of data of various data types, except text, ntext, and timestamp
uniqueidentifier	Stores a globally unique identifier (GUID)
xml	Stores XML formatted data. Maximum 2GB
cursor	Stores a reference to a cursor used for database operations
table	Stores a result-set for later processing

Πίνακας 64– *Τύποι δεδομένων ημερομηνίας και ώρας (SQL Server)*

*(**Πηγή**: https://www.w3schools.com/sql/sql_datatypes.asp)*

Τύποι δεδομένων στο σύστημα MS Access

Data type	Description	Storage
Text	Use for text or combinations of text and numbers. 255 characters maximum	
Memo	Memo is used for larger amounts of text. Stores up to 65,536 characters. **Note:** You cannot sort a memo field. However, they are searchable	
Byte	Allows whole numbers from 0 to 255	1 byte
Integer	Allows whole numbers between -32,768 and 32,767	2 bytes
Long	Allows whole numbers between -2,147,483,648 and 2,147,483,647	4 bytes
Single	Single precision floating-point. Will handle most decimals	4 bytes
Double	Double precision floating-point. Will handle most decimals	8 bytes
Currency	Use for currency. Holds up to 15 digits of whole dollars, plus 4 decimal places. **Tip:** You can choose which country's currency to use	8 bytes
AutoNumber	AutoNumber fields automatically give each record its own number, usually starting at 1	4 bytes
Date/Time	Use for dates and times	8 bytes
Yes/No	A logical field can be displayed as Yes/No, True/False, or On/Off. In code, use the constants True and False (equivalent to -1 and 0). **Note:** Null values are not allowed in Yes/No fields	1 bit
Ole Object	Can store pictures, audio, video, or other BLOBs (Binary Large Objects)	up to 1GB
Hyperlink	Contain links to other files, including web pages	
Lookup Wizard	Let you type a list of options, which can then be chosen from a drop-down list	4 bytes

Πίνακας 65– *Τύποι δεδομένων (MS Access) (**Πηγή**: https://www.w3schools.com/sql/sql_datatypes.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

4.3. Αναφορές στην SQL (References)

Λέξεις-κλειδιά στην SQL

Λέξη-κλειδί	Περιγραφή
ADD	Προσθέτει μια στήλη σε έναν υπάρχοντα πίνακα
ADD CONSTRAINT	Προσθέτει έναν περιορισμό μετά την δημιουργία ενός πίνακα
ALL	Εμφανίζεται TRUE αν όλες οι τιμές ενός δευτερεύοντος ερωτήματος πληρούν την προϋπόθεση
ALTER	Προσθέτει, διαγράφει ή τροποποιεί στήλες σε έναν πίνακα ή αλλάζει τον τύπο δεδομένων μιας στήλης σε έναν πίνακα
ALTER COLUMN	Αλλάζει τον τύπο δεδομένων μιας στήλης σε έναν πίνακα
ALTER TABLE	Προσθέτει, διαγράφει ή τροποποιεί στήλες σε έναν πίνακα
AND	Περιλαμβάνει μόνο γραμμές όπου και οι δύο συνθήκες είναι αληθείς
ANY	Εμφανίζεται TRUE αν όλες οι τιμές των υποερωτημάτων πληρούν την προϋπόθεση

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

AS	Μετονομασία στήλης ή πίνακα με ψευδώνυμο
ASC	Ταξινόμηση του αποτελέσματος σε αύξουσα σειρά
BACKUP DATABASE	Δημιουργεί ένα εφεδρικό αντίγραφο μιας υπάρχουσας βάσης δεδομένων
BETWEEN	Επιλογή τιμών εντός ενός δοσμένου εύρους
CASE	Εμφανίζει διαφορετικά αποτελέσματα με βάση τις συνθήκες
CHECK	Ένας περιορισμός που θέτει ένα όριο στην τιμή που μπορεί να τοποθετηθεί σε μια στήλη
COLUMN	Αλλάζει τον τύπο δεδομένων μιας στήλης ή διαγράφει μια στήλη σε έναν πίνακα
CONSTRAINT	Προσθέτει ή διαγράφει έναν περιορισμό
CREATE	Δημιουργεί μια βάση δεδομένων, δείκτη, προβολή, πίνακα ή διαδικασία
CREATE DATABASE	Δημιουργεί μια νέα βάση δεδομένων SQL

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

CREATE INDEX	Δημιουργεί έναν δείκτη σε έναν πίνακα (επιτρέπει διπλότυπες τιμές)
CREATE OR REPLACE VIEW	Ενημερώνει μια προβολή
CREATE TABLE	Δημιουργεί έναν νέο πίνακα στη βάση δεδομένων
CREATE PROCEDURE	Δημιουργεί μια αποθηκευμένη διαδικασία
CREATE UNIQUE INDEX	Δημιουργεί ένα μοναδικό δείκτη σε έναν πίνακα (χωρίς διπλές τιμές)
CREATE VIEW	Δημιουργεί μια προβολή με βάση το σύνολο αποτελεσμάτων μιας εντολής SELECT
DATABASE	Δημιουργεί ή διαγράφει μια βάση δεδομένων SQL
DEFAULT	Ένας περιορισμός που παρέχει μια προεπιλεγμένη τιμή για μια στήλη
DELETE	Διαγράφει σειρές από έναν πίνακα
DESC	Ταξινόμηση των αποτελεσμάτων κατά φθίνουσα σειρά

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

DISTINCT	Επιλέγει μόνο διακριτές (διαφορετικές) τιμές
DROP	Διαγράφει μια στήλη, περιορισμό, βάση δεδομένων, δείκτη, πίνακα ή προβολή
DROP COLUMN	Διαγράφει στήλες από έναν πίνακα
DROP CONSTRAINT	Διαγράφει έναν περιορισμό UNIQUE, PRIMARY KEY, FOREIGN KEY, ή CHECK
DROP DATABASE	Διαγράφει μια υπάρχουσα βάση δεδομένων SQL
DROP DEFAULT	Διαγράφει έναν περιορισμό DEFAULT
DROP INDEX	Διαγράφει έναν δείκτη σε έναν πίνακα
DROP TABLE	Διαγράφει έναν υπάρχοντα πίνακα στη βάση δεδομένων
DROP VIEW	Διαγράφει μια προβολή
EXEC	Εκτελεί μια αποθηκευμένη διαδικασία
EXISTS	Εξετάζει την ύπαρξη οποιασδήποτε εγγραφής σε ένα υποερώτημα

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

FOREIGN KEY	Ένας περιορισμός που αποτελεί το κλειδί που χρησιμοποιείται για τη σύνδεση δύο πινάκων
FROM	Καθορίζει από ποιον πίνακα θα επιλεγούν ή θα διαγραφούν δεδομένα
FULL OUTER JOIN	Εμφανίζει όλες τις σειρές όταν υπάρχει αντιστοιχία είτε στον πίνακα στα αριστερά είτε στα δεξιά
GROUP BY	Ομαδοποιεί το σύνολο αποτελεσμάτων (χρησιμοποιείται με αθροιστικές συναρτήσεις: COUNT, MAX, MIN, SUM, AVG)
HAVING	Χρησιμοποιείται αντί για το WHERE με αθροιστικές συναρτήσεις
IN	Σας επιτρέπει να καθορίσετε πολλαπλές τιμές σε έναν όρο WHERE
INDEX	Δημιουργεί ή διαγράφει έναν δείκτη σε έναν πίνακα
INNER JOIN	Επιλέγει σειρές που έχουν τιμές που είναι ίδιες και στους δύο πίνακες
INSERT INTO	Εισάγει νέες σειρές σε έναν πίνακα

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

INSERT INTO SELECT	Αντιγράφει δεδομένα από έναν πίνακα σε άλλον πίνακα
IS NULL	Εξετάζει το ενδεχόμενο κενών τιμών
IS NOT NULL	Εξετάζει το ενδεχόμενο μη κενών τιμών
JOIN	Συγχωνεύει πίνακες
LEFT JOIN	Εμφανίζει όλες τις σειρές από τον αριστερό πίνακα και τις σειρές που είναι ίδιες από τον δεξί πίνακα
LIKE	Αναζητά ένα συγκεκριμένο μοτίβο σε μια στήλη
LIMIT	Καθορίζει τον αριθμό των εγγραφών που θα εμφανιστούν στο σύνολο αποτελεσμάτων
NOT	Περιλαμβάνει μόνο σειρές όπου μια συνθήκη δεν είναι TRUE
NOT NULL	Ένας περιορισμός που επιβάλλει σε μια στήλη να μην δέχεται κενές τιμές
OR	Περιλαμβάνει σειρές όπου οποιαδήποτε από τις συνθήκες είναι TRUE

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ORDER BY	Ταξινόμηση των αποτελεσμάτων κατά αύξουσα ή φθίνουσα σειρά
OUTER JOIN	Εμφανίζει όλες τις σειρές όπου υπάρχει αντιστοιχία είτε στον αριστερό είτε στον δεξί πίνακα
PRIMARY KEY	Ένας περιορισμός που προσδιορίζει ως μοναδική κάθε εγγραφή σε έναν πίνακα βάσης δεδομένων
PROCEDURE	Μια αποθηκευμένη δοκιμασία
RIGHT JOIN	Εμφανίζει όλες τις σειρές από το δεξί πίνακα και τις σειρές που είναι ίδιες με τον αριστερό πίνακα
ROWNUM	Καθορίζει τον αριθμό των εγγραφών που θα εμφανιστούν στο σύνολο αποτελεσμάτων
ΕΠΙΛΟΓΗ	Επιλογή δεδομένων από μια βάση δεδομένων
SELECT DISTINCT	Επιλέγει μόνο διακριτές (διαφορετικές) τιμές
SELECT INTO	Αντιγράφει δεδομένα από έναν πίνακα σε έναν νέο πίνακα
SELECT TOP	Καθορίζει τον αριθμό των εγγραφών που θα εμφανιστούν στο σύνολο αποτελεσμάτων

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

SET	Καθορίζει ποιες στήλες και τιμές θα πρέπει να ενημερωθούν σε έναν πίνακα
TABLE	Δημιουργεί έναν πίνακα ή προσθέτει, διαγράφει ή τροποποιεί στήλες σε έναν πίνακα ή διαγράφει έναν πίνακα ή τα δεδομένα μέσα σε αυτόν
TOP	Καθορίζει τον αριθμό των εγγραφών που θα εμφανιστούν στο σύνολο αποτελεσμάτων
TRUNCATE TABLE	Διαγράφει τα δεδομένα μέσα σε έναν πίνακα, αλλά όχι τον ίδιο τον πίνακα
UNION	Συνδυάζει το σύνολο των αποτελεσμάτων δύο ή περισσότερων εντολών SELECT (μόνο διακριτές τιμές)
UNION ALL	Συνδυάζει το σύνολο αποτελεσμάτων δύο ή περισσότερων εντολών SELECT (επιτρέπει διπλότυπες τιμές)
UNIQUE	Ένας περιορισμός που διασφαλίζει ότι όλες οι τιμές σε μια στήλη είναι μοναδικές
UPDATE	Ενημερώνει τις υπάρχουσες σειρές σε έναν πίνακα
VALUES	Καθορίζει τις τιμές μιας εντολής INSERT INTO

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

VIEW	Δημιουργεί, ενημερώνει ή διαγράφει μια προβολή

***Πίνακας 66**– Λέξεις-κλειδιά στην SQL και περιγραφές (**Πηγή**: https://www.w3schools.com/sql/sql_ref_keywords.asp)*

Λειτουργίες στο σύστημα MySQL

Για περισσότερες λεπτομέρειες και μια ολοκληρωμένη λίστα με συγκεκριμένες λειτουργίες που χρησιμοποιούνται στο σύστημα MySQL, οι εκπαιδευόμενοι μπορούν να ανατρέξουν σε αυτόν τον σύνδεσμο.

Λειτουργίες στο σύστημα SQL Server

Για περισσότερες λεπτομέρειες και μια ολοκληρωμένη λίστα με συγκεκριμένες λειτουργίες που χρησιμοποιούνται στο σύστημα SQL Server, οι εκπαιδευόμενοι μπορούν να ανατρέξουν σε αυτόν τον σύνδεσμο.

Λειτουργίες στο σύστημα MS Access

Για περισσότερες λεπτομέρειες και μια ολοκληρωμένη λίστα με συγκεκριμένες λειτουργίες που χρησιμοποιούνται στο σύστημα MS Access, οι εκπαιδευόμενοι μπορούν να ανατρέξουν σε αυτόν τον σύνδεσμο.

Σύνοψη της γλώσσας SQL

Για έναν πλήρη κατάλογο των εντολών της SQL και της αντίστοιχης σύνταξής τους, οι εκπαιδευόμενοι μπορούν να ανατρέξουν σε αυτόν τον σύνδεσμο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

4.4. Παραδείγματα της SQL

Παραδείγματα της SQL

Υπάρχει ένας περιεκτικός κατάλογος παραδειγμάτων στην ιστοσελίδα του W3Schools, τα οποία μπορούν να χρησιμοποιήσουν οι εκπαιδευόμενοι για να μελετήσουν μόνοι τους και να εξασκήσουν περαιτέρω τις δεξιότητές τους στη γλώσσα SQL.

Κουίζ για την SQL

Για τους εκπαιδευόμενους που θέλουν να αξιολογήσουν τις γνώσεις και τις δεξιότητές τους στην SQL, ανατρέξτε σε έναν από τους παρακάτω ιστότοπους:

- W3Schools
- Tutorialspoint

Ασκήσεις στην SQL

Για μια ολοκληρωμένη λίστα των ασκήσεων, οι εκπαιδευόμενοι μπορούν να χρησιμοποιήσουν την ιστοσελίδα του W3Schools για να εξασκήσουν τις δεξιότητές τους στην SQL.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

5. JavaScript

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πληροφορίες για την ενότητα

Ενότητα:

5. JavaScript

Προαπαιτούμενες γνώσεις:

Για να μάθουν την JavaScript, οι εκπαιδευόμενοι πρέπει να γνωρίζουν τα βασικά της HTML και της CSS. Εαν οι εκπαιδευόμενοι επιθυμούν να αποκτήσουν πρακτικές γνώσεις της JavaScript και των περισσότερων έργων που βασίζονται στο διαδίκτυο, τότε η γνώση που προσφέρεται στο κεφάλαιο αυτό θα είναι επαρκής. Για πιο προηγμένα έργα και δεξιότητες, συνιστάται να γνωρίζετε βασικές έννοιες του OOP και μια γλώσσα προγραμματισμού βασισμένη σε OOP (όπως η Java).

Φόρτος εργασίας:

15 Ώρες.

Περιγραφή:

Η JavaScript είναι μια γλώσσα προγραμματισμού που επιτρέπει στον προγραμματιστή να εκτελεί αλλαγές στο περιεχόμενο μιας ιστοσελίδας με δυναμικό τρόπο. Η JavaScript είναι μία από τις πιο δυνατές και ευέλικτες γλώσσες προγραμματισμού του διαδικτύου.

Το κεφάλαιο αυτό για την JavaScript καλύπτει όλες τις θεμελιώδεις έννοιες προγραμματισμού, συμπεριλαμβανομένων των τύπων δεδομένων, των χειριστών, της δημιουργίας και της χρήσης μεταβλητών, της παραγωγής αποτελεσμάτων, της διάρθρωσης του κώδικα για τη λήψη αποφάσεων σε προγράμματα ή για την επανάληψη του ίδιου μπλοκ κώδικα πολλές φορές με τη δήλωση ελέγχου βρόχου, της δημιουργίας και επεξεργασίας συμβολοσειρών και συστοιχιών, του καθορισμού και της κλήσης συναρτήσεων κ.λπ.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μόλις οι εκπαιδευόμενοι εξοικειωθούν με τα βασικά, θα προχωρήσουν στο επόμενο επίπεδο που εξηγεί την ιδέα των αντικειμένων, το Μοντέλο Αντικειμένων Εγγράφων (DOM/ Document Object Model) και το Μοντέλο Αντικειμένων Περιηγητή (BOM/ Browser Object Model), καθώς και πώς να κάνουν χρήση των εγγενών αντικειμένων JavaScript όπως Ημερομηνία, Μαθηματικά κ.λπ., και να εκτελούν μετατροπές τύπου.

Επιπλέον, θα διερευνηθούν ορισμένες σύνθετες έννοιες όπως το event listeners, event propagation, οι μέθοδοι δανεισμού από άλλα αντικείμενα, η συμπεριφορά ανύψωσης JavaScript, η κωδικοποίηση και η αποκωδικοποίηση δεδομένων JSON, καθώς και η λεπτομερής επισκόπηση των νέων χαρακτηριστικών που εισήχθησαν στο ECMAScript 6 (ή ES6).

Μαθησιακά αποτελέσματα:

Οι εκπαιδευόμενοι θα καταλάβουν πώς η JavaScript επιτρέπει στον προγραμματιστή να αλλάξει το περιεχόμενο μιας ιστοσελίδας με δυναμικούς τρόπους και να τροποποιεί τις πληροφορίες HTML ή CSS από την πλευρά του πελάτη, όταν η ιστοσελίδα εμφανίζεται στο χρήστη. Ως εκ τούτου, οι εκπαιδευόμενοι θα μελετήσουν τις δομές του κώδικα JavaScript, θα γνωρίσουν πώς να αλλάζουν τον κώδικα HTML/CSS κατά τη φόρτωση της σελίδας και επίσης θα μάθουν πώς να δημιουργούν μια λειτουργία και να την επισυνάψουν σε μια εκδήλωση.

Απαιτούμενα υλικά:

- Υπολογιστής ή φορητός υπολογιστής
- Σύνδεση στο Διαδίκτυο
- Επεξεργαστής κειμένου (online ή offline): Sublime Text/Brackets/W3Schools online editor

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σενάριο μαθήματος:

Ο συνολικός χρόνος για το θέμα αυτό είναι 10 ώρες και θα εναπόκειται στον εκπαιδευτή να αποφασίσει πόσο χρόνο θα αφιερώσει στη διδασκαλία κάθε υποθέματος. Προκειμένου να αξιοποιήσουμε στο έπακρο όλο τον διαθέσιμο χρόνο, προτείνουμε τη χρήση του εκπαιδευτικού υλικού που συντάχθηκε στο πλαίσιο του έργου (παρουσιάσεις PPT), το οποίο σχεδιάστηκε με γνώμονα την αποτελεσματική χρήση του χρόνου. Αυτές οι παρουσιάσεις αποτελούνται από τα ακόλουθα στοιχεία:

- Ανάπτυξη του υποθέματος και των βασικών ιδεών που πρέπει να διατηρηθούν.
- Προτεινόμενες Δραστηριότητες/Ασκήσεις.

Επομένως, εάν ο εκπαιδευτής ακολουθήσει τη λογική σειρά των PPTs, σίγουρα θα μπορέσει να ολοκληρώσει τη συνεδρία εντός του καθορισμένου χρονικού ορίου. Αυτές οι παρουσιάσεις μπορούν επίσης να διατεθούν στους μαθητές για ατομική μελέτη.

Υποθέματα:

- Βασικές γνώσεις JavaScript
- JavaScript & DOM
- JavaScript & BOM
- JavaScript για προχωρημένους
- Αναφορά JavaScript

Επιπλέον πόροι:

- Οδηγός JavaScript: w3schools
- Διαδικτυακά μαθήματα στην JavaScript: CodeAcademy

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

5.1. JavaScript Basic (CEPROF)

Η JavaScript (JS) είναι η πιο διαδεδομένη γλώσσα προγραμματισμού σεναρίων (scripting language) *από την πλευρά του πελάτη* (η γλώσσα προγραμματισμού σεναρίων, από την πλευρά του πελάτη, σχετίζεται με τα σενάρια που εκτελούνται μέσα σε ένα πρόγραμμα περιήγησης ιστού). Η JS προσθέτει διαδραστικότητα και δυναμικά εφέ στις ιστοσελίδες, επεξεργάζοντας το περιεχόμενο που επιστρέφεται από έναν διακομιστή ιστού.

Η JavaScript αναπτύχθηκε για πρώτη φορά ως LiveScript από τον προγραμματιστή υπολογιστών της Netscape, Brendan Eich, το 1995. Αργότερα μετονομάστηκε σε JavaScript και έγινε πρότυπο ECMA (European Computer Manufacturers Association - Ευρωπαϊκή Ένωση Κατασκευαστών Υπολογιστών) το 1997. Σήμερα, η JavaScript είναι η τυπική γλώσσα σεναρίου από την πλευρά του πελάτη για διαδικτυακές εφαρμογές και υποστηρίζεται από όλους σχεδόν του περιηγητές ιστού (Chrome, Firefox, Safari κ.λπ.).

Η JavaScript είναι μια αντικειμενοστραφής γλώσσα, και έχει επίσης κάποιες ομοιότητες στη σύνταξη με τη γλώσσα προγραμματισμού Java, παρόλο που δεν σχετίζεται καθόλου με την Java.

Η JavaScript μπορεί να χρησιμοποιηθεί για ποικίλους σκοπούς, όπως:

- Για την τροποποίηση του περιεχομένου μιας ιστοσελίδας προσθέτοντας ή αφαιρώντας στοιχεία·
- Για την αλλαγή του στυλ και της θέση των στοιχείων σε μια ιστοσελίδα·
- Για την παρακολούθηση των ενεργειών, όπως το κλικ του ποντικιού, το κείμενο κατάδειξης (hover), κ.λπ. και την αντίδραση σε αυτά·
- Για την πραγματοποίηση και τον έλεγχο των μεταβάσεων και κινούμενων εικόνων·
- Για την δημιουργία αναδυόμενων παραθύρων ειδοποίησης για την εμφάνιση πληροφοριών ή προειδοποιητικών μηνυμάτων στον χρήστη.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Για την ολοκλήρωση των λειτουργιών με βάση τα στοιχεία εισαγωγής από τον χρήστη και εμφάνιση των αποτελεσμάτων.
- Για την επικύρωση των στοιχείων εισαγωγής από τον χρήστη πριν τα υποβάλει στον διακομιστή.
- Και πολλούς άλλους ενδιαφέροντες λόγους που θα εξεταστούν αργότερα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ας ξεκινήσουμε λοιπόν με την JavaScript!

Ξεκινώντας από το σημείο αυτό, οι εκπαιδευόμενοι θα αντιληφθούν στην πορεία πόσο απλό μπορεί να είναι να προσθέσουν διαδραστικότητα σε μια ιστοσελίδα χρησιμοποιώντας την JavaScript.

Υπάρχουν 3 χαρακτηριστικοί τρόποι προσθήκης της JS σε μια ιστοσελίδα:

1) Με την ενσωμάτωση του κώδικα JavaScript μεταξύ μιας ετικέτας <script> και μιας ετικέτας </script>.
2) Με την δημιουργία εξωτερικού αρχείου JavaScript με την επέκταση .js και στη συνέχεια τη φόρτωσή του μέσα στη σελίδα μέσω της ιδιότητας src της ετικέτας <script>.
3) Με την τοποθέτηση του κώδικα JavaScript απευθείας μέσα σε μια ετικέτα HTML χρησιμοποιώντας τα ειδικά χαρακτηριστικά ετικέτας όπως onclick , onmouseover, onkeypress, onload κ.λπ.

α) Ενσωμάτωση του κώδικα JavaScript μεταξύ μιας ετικέτας <script> και μιας ετικέτας </script> ;

Ο κώδικας της JavaScript μπορεί να ενσωματωθεί απευθείας μέσα σε μια ιστοσελίδα τοποθετώντας τον μεταξύ των ετικετών <script> και </script>. Η ετικέτα <script> υποδεικνύει στο πρόγραμμα περιήγησης ότι οι δηλώσεις που περιέχονται πρέπει να ερμηνεύονται ως εκτελέσιμο σενάριο και όχι ως HTML, όπως παρουσιάζεται στο παρακάτω παράδειγμα:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Embedding JavaScript</title>
</head>
<body>
    <script>
    var greet = "Hello World!";
    document.write(greet); // Prints: Hello World!
    </script>
</body>
</html>
```

Hello World!

Εικόνα 1– *Ενσωμάτωση κωδικού JS μεταξύ μιας <script> and a </script> ετικέτας*

(***Πηγή***: https://www.w3schools.com/css/css_rwd_images.asp)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

β) Δημιουργία εξωτερικού αρχείου JavaScript με την επέκταση .js και στη συνέχεια η φόρτωσή του μέσα στη σελίδα μέσω της ιδιότητας src της ετικέτας <script>.

Ένας κώδικας JavaScript μπορεί επίσης να τοποθετηθεί σε ένα ξεχωριστό αρχείο με επέκταση .js, που στη συνέχεια καλείται σε αυτό το αρχείο στο ίδιο έγγραφο μέσω του χαρακτηριστικού src της ετικέτας <script>, ως εξής:

```
<script src="js/hello.js"></script>
```

Αυτός ο κώδικας μπορεί να φανεί ιδιαίτερα χρήσιμος εάν ο προγραμματιστής θέλει τα ίδια σενάρια να είναι διαθέσιμα σε πολλαπλά έγγραφα. Μετά από τη διαδικασία αυτή, θα αποφύγει να επαναλάβει την ίδια εργασία ξανά και ξανά, και αυτό καθιστά πολύ πιο απλή τη διαδικασία συντήρησης της ιστοσελίδας του/της.

Με την προώθηση της παραπάνω δήλωσης, θα δημιουργηθεί ένα αρχείο JavaScript με το όνομα "hello.js" και θα εισαχθεί σε αυτό ο ακόλουθος κώδικας:

```
1  // A function to display a message
2  function sayHello() {
3      alert("Hello World!");
4  }
5
6  // Call function on click of the button
7  document.getElementById("myBtn").onclick = sayHello;
```

Εικόνα 2– *Δημιουργία αρχείου και επίκληση μιας συνάρτησης στο JS*

(***Πηγή***: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Στο σημείο αυτό, ο προγραμματιστής μπορεί να επικαλεστεί το παραπάνω αρχείο JS μέσα σε μια ιστοσελίδα χρησιμοποιώντας την ετικέτα <script>, όπως φαίνεται στην παρακάτω εικόνα:

Εικόνα 3– *Δημιουργία αρχείου και επίκληση συνάρτησης στο JS*

(***Πηγή***: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

γ) Η τοποθέτηση του κώδικα JavaScript απευθείας μέσα σε μια ετικέτα HTML χρησιμοποιώντας τα ειδικά χαρακτηριστικά ετικέτας όπως onclick, onmouseover, onkeypress, onload κ.λπ.

Ο κώδικας JavaScript μπορεί να εισαχθεί ενσωματωμένος εισάγοντάς τον απευθείας μέσα στην ετικέτα HTML μέσω των ειδικών χαρακτηριστικών της ετικέτας όπως onclick, onmouseover, onkeypress, onload, κλπ.

Παρ 'όλα αυτά, δεν συστήνεται η προσθήκη μεγάλου κώδικα JavaScript μέσα στον html κώδικα inline, καθώς μπορεί να προκαλέσει δυσλειτουργίες μεταξύ HTML με JavaScript, καθιστώντας δύσκολη τη διατήρηση του κώδικα JS. *Το Σχήμα 4* παρουσιάζει ένα παράδειγμα (στην περίπτωση αυτή, εμφανίζεται ένα μήνυμα ειδοποίησης όταν κάνετε κλικ στο στοιχείο κουμπιού):

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εικόνα 4 *– Τοποθέτηση/Πρόσθεση του κώδικα JS σε σειρά*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Το στοιχείο <script> μπορεί να τοποθετηθεί στο < head> ή στο τμήμα <body> ενός εγγράφου HTML. Ωστόσο, τα σενάρια θα πρέπει να τοποθετούνται κατά προτίμηση στο τέλος του τμήματος του σώματος, πριν από την </body> ετικέτα κλεισίματος. Αυτή η διαδικασία θα επιτρέψει στις ιστοσελίδες να φορτωθούν γρηγορότερα, δεδομένου ότι θα αποφευχθεί η παρεμπόδιση της απόκρισης της αρχικής σελίδας. Κάθε ετικέτα <script> αποκλείει τη διαδικασία απόκρισης μιας σελίδας μέχρι να κατεβάσει και να εκτελέσει πλήρως τον κώδικα JavaScript, οπότε η τοποθέτησή τους στο τμήμα της κεφαλίδας (δηλαδή στο < head> στοιχείο) του εγγράφου χωρίς κανένα έγκυρο λόγο θα επηρεάσει σημαντικά την απόκριση μιας ιστοσελίδας.

Υπάρχουν διαφορές μεταξύ της γλώσσας προγραμματισμού σεναρίων από την πλευρά του πελάτη και από την πλευρά του διακομιστή. Οι γλώσσες σεναρίου από την πλευρά του πελάτη (π.χ. JavaScript ή VBScript) κατανοούνται και εκτελούνται από το πρόγραμμα περιήγησης ιστού, σε αντίθεση με τις γλώσσες σεναρίου από την πλευρά του διακομιστή (π.χ. PHP, ASP, Java, Python, Ruby κ.λπ.), οι οποίες εκτελούνται στον διακομιστή ιστού και η εξαγωγή τους επιστρέφεται πίσω στο πρόγραμμα περιήγησης ιστού σε μορφή HTML.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η γλώσσα σεναρίου από την πλευρά του πελάτη έχει πολλά πλεονεκτήματα σε σύγκριση με την κλασική γλώσσα σεναρίου από την πλευρά του διακομιστή. Για παράδειγμα, η JavaScript μπορεί να χρησιμοποιηθεί για να ελέγξει αν ο χρήστης έχει εισάγει μη έγκυρα δεδομένα σε πεδία φόρμας και να εμφανίσει συνεπώς ειδοποιήσεις για σφάλματα εισόδου σε πραγματικό χρόνο πριν από την υποβολή της φόρμας στο web-server για την τελική επικύρωση και περαιτέρω επεξεργασία δεδομένων, προκειμένου να αποφευχθούν περιττές χρήσεις εύρους ζώνης δικτύου και η κακή χρήση των πόρων του συστήματος διακομιστή.

Παρομοίως, η απόκριση της γλώσσας σεναρίου από την πλευρά του διακομιστή είναι πιο αργή σε σύγκριση με τη γλώσσα σεναρίου από την πλευρά του πελάτη, καθώς τα σενάρια από την πλευρά του διακομιστή υποβάλλονται σε επεξεργασία σε έναν απομακρυσμένο υπολογιστή (όχι σε έναν τοπικό).

Σύνταξη (κανόνων) JavaScript

Είναι καιρός να μάθουμε πώς να γράφουμε τους κώδικες της JS.

Πρώτον, είναι σημαντικό να κατανοήσουμε τη Σύνταξη της JavaScript.

Η σύνταξη της JS είναι το **σύνολο των κανόνων** που περιλαμβάνουν ένα καλά δομημένο πρόγραμμα JavaScript. Το JS περιλαμβάνει δηλώσεις που τοποθετούνται εντός των ετικετών <script> </script> HTML σε μια ιστοσελίδα ή εντός του εξωτερικού αρχείου JavaScript με επέκταση .js.

Το Σχήμα 5 παρακάτω παρουσιάζει ένα παράδειγμα μιας δήλωσης JS:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Example of JavaScript Statements</title>
</head>
<body>
    <script>
    var x = 5;
    var y = 10;
    var sum = x + y;
    document.write(sum); // Prints variable value
    </script>
</body>
</html>
```

15

Εικόνα 5 *– Παράδειγμα δήλωσης JavaScript*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Οι εκπαιδευόμενοι θα πρέπει να δηλώσουν ότι η JavaScript έχει **διάκριση πεζών/κεφαλαίων**. Επομένως, οι μεταβλητές, οι λέξεις-κλειδιά, τα ονόματα των λειτουργιών και άλλα αναγνωριστικά πρέπει να δακτυλογραφούνται με συνέπεια όσον αφορά την κεφαλαιοποίηση των γραμμάτων. Για παράδειγμα, η μεταβλητή myVar πρέπει να πληκτρολογείται με αυτόν τον τρόπο (όχι "MYVAR", "myvar" κ.λπ.). Αυτό ισχύει για **όλες τις περιπτώσεις**.

Επίσης, η JavaScript παρέχει επίσης τη δυνατότητα να γράψετε σχόλια σε όλες τις γραμμές κωδικοποίησης. Τα σχόλια εισάγονται κυρίως επειδή παρέχουν πρόσθετες πληροφορίες για τον πηγαίο κώδικα, αλλά και επειδή μπορούν να βοηθήσουν τους προγραμματιστές να κατανοήσουν τους κώδικες τους μετά από κάποιο χρονικό διάστημα, ομαδική εργασία κ.λπ.

Είναι δυνατή η προσθήκη σχολίων σε μία και πολλαπλές γραμμές στο JavaScript. Τα σχόλια μιας γραμμής ξεκινούν με μια διπλή κάθετο προς τα εμπρός (//), ακολουθούμενη από το κείμενο του σχολίου:

```
// This is my first JavaScript program
document.write("Hello World!");
```

Σχήμα 6 *– Σχόλιο μιας γραμμής στο JavaScript*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για ένα σχόλιο πολλαπλών γραμμών, μια κάθετος και ένας αστερίσκος (/*) είναι το σημείο εκκίνησης, που τελειώνει με έναν αστερίσκο και μια κάθετο (*/):

```
1  /* This is my first program
2  in JavaScript */
3  document.write("Hello World!");
```

***Εικόνα 7** – Σχόλιο πολλαπλών γραμμών στο JavaScript*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Μεταβλητές JavaScript

Για την αποθήκευση δεδομένων σε JavaScript, οι προγραμματιστές δημιουργούν μεταβλητές. Είναι το κλειδί για όλες τις γλώσσες προγραμματισμού και χρησιμοποιούνται για την αποθήκευση δεδομένων, για παράδειγμα με συμβολοσειρά κειμένου, αριθμούς ή άλλο/α στοιχείο. Όταν χρειαστεί, ο προγραμματιστής μπορεί να ρυθμίσει, να ενημερώσει και να ανακτήσει δεδομένα ή τιμές που είναι αποθηκευμένες στις μεταβλητές. Οι μεταβλητές μπορούν να γίνουν κατανοητές ως συμβολικά ονόματα για τις τιμές.

Μια μεταβλητή μπορεί να δημιουργηθεί χρησιμοποιώντας τη λέξη-κλειδί var, στην οποία ο τελεστής εκχώρησης (“=”) χρησιμοποιείται για την κατανομή τιμής σε μια μεταβλητή, ως εξής: `var varName = value`

```
1  var name = "Peter Parker";
2  var age = 21;
3  var isMarried = false;
```

***Εικόνα 8** – Δημιουργία μεταβλητής*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Σε παράδειγμα αυτό, έχουν δημιουργηθεί τρεις μεταβλητές:
Αριθμός 1 – τιμή συμβολοσειράς
Αριθμός 2 – Αριθμός
Αριθμός 3 – Δυαδική τιμή

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στην JS, δημιουργούνται ορισμένες μεταβλητές με σκοπό τη διατήρηση τιμών όπως η εισαγωγή στοιχείων από τον χρήστη. Τούτου λεχθέντος, μπορούν να δηλωθούν χωρίς να έχουν συσχετισθεί αρχικές τιμές, ως εξής:

```
1  // Declaring Variable
2  var userName;
3
4  // Assigning value
5  userName = "Clark Kent";
```

Σχήμα 9 *– Δημιουργία μιας μεταβλητής χωρίς να τους έχουν ανατεθεί αρχικές τιμές*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Επιπλέον, ο προγραμματιστής μπορεί επίσης να δηλώσει πολλαπλές μεταβλητές και, σε μια επιμέρους δήλωση, να ορίσει τις αρχικές τους τιμές. Κάθε μεταβλητή αποσπάται με κόμματα, ως εξής:

```
1  // Declaring multiple Variables
2  var name = "Peter Parker", age = 21, isMarried = false;
3
4  /* Longer declarations can be written to span
5  multiple lines to improve the readability */
6  var name = "Peter Parker",
7  age = 21,
8  isMarried = false;
```

Εικόνα 10 *– Δήλωση πολλαπλών μεταβλητών*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η τελευταία αναθεωρημένη έκδοση της JavaScript (ECMAScript 2015 ή ES6) εισάγει δύο νέες λέξεις-κλειδιά για τη δήλωση των μεταβλητών: let και const.
Η λέξη κλειδί const λειτουργεί με τον ίδιο τρόπο όπως και το let. Ωστόσο, οι μεταβλητές που δηλώνονται με τη χρήση της const δεν μπορούν να ανατεθούν εκ νέου αργότερα στον κωδικό, ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
// Declaring variables
let name = "Harry Potter";
let age = 11;
let isStudent = true;

// Declaring constant
const PI = 3.14;
console.log(PI); // 3.14

// Trying to reassign
PI = 10; // error
```

Εικόνα 11 *– Δήλωση πολλαπλών μεταβλητών*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Σε αντίθεση με τη λέξη-κλειδί var, η οποία δηλώνει μεταβλητές με εμβέλεια στο σώμα μιας συνάρτησης (function-scoped variables), τόσο η λέξη-κλειδί let όσο και η λέξη-κλειδί const δηλώνουν μεταβλητές με εμβέλεια σε επίπεδο μπλοκ ({}). Η οριοθέτηση του πεδίου εφαρμογής κατά μπλοκ σημαίνει ότι δημιουργείται ένα νέο πεδίο εφαρμογής μεταξύ ενός ζεύγους αγκύλων.

Οι μεταβλητές JavaScript έχουν **συγκεκριμένους κανόνες** για την ονομασία τους:

- Ένα όνομα μεταβλητής πρέπει να ξεκινά με ένα γράμμα, με κάτω παύλα (_), ή το σύμβολο δολαρίου ($).
- Ένα όνομα μεταβλητής δεν μπορεί να ξεκινήσει με έναν αριθμό.
- Ένα μεταβλητό όνομα μπορεί να περιλαμβάνει μόνο αλφαριθμητικούς χαρακτήρες (A-Z, 0-9) και κάτω παύλες.
- Ένα όνομα μεταβλητής δεν μπορεί να κατανοήσει κενά.
- Ένα όνομα μεταβλητής δεν μπορεί να είναι μια λέξη-κλειδί JavaScript ή μια δεσμευμένη λέξη JavaScript (reserved words).

Παραγωγή εξόδου JavaScript (Output)

Υπάρχουν ορισμένες περιπτώσεις στις οποίες οι προγραμματιστές μπορεί να χρειαστεί να δημιουργήσουν εξόδους από τον κώδικα JS, π.χ., δείτε την τιμή της μεταβλητής, γράψτε ένα μήνυμα στην κονσόλα του προγράμματος περιήγησης, κλπ. Στην

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

JavaScript, υπάρχουν μερικοί διαφορετικοί τρόποι παραγωγής εξόδου, συμπεριλαμβανομένης της εγγραφής εξόδου στο παράθυρο του προγράμματος περιήγησης ή στην κονσόλα του προγράμματος περιήγησης, της εμφάνισης εξόδου σε πλαίσια διαλόγου, της εγγραφής εξόδου σε στοιχείο HTML κ.λπ.

Δεν είναι δύσκολη η εξαγωγή ενός μηνύματος ή η εγγραφή δεδομένων στην κονσόλα του προγράμματος περιήγησης (μπορείτε να έχετε πρόσβαση σε αυτό κάνοντας κλικ στο F12). Για τον σκοπό αυτό, θα πρέπει να εφαρμόζεται η console.log (), μια πολύ δυνατή αλλά και απλή μέθοδος.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Writing into the Browser's Console with JavaScript</title>
</head>
<body>
    <script>
    // Printing a simple text message
    console.log("Hello World!"); // Prints: Hello World!

    // Printing a variable value
    var x = 10;
    var y = 20;
    var sum = x + y;
    console.log(sum); // Prints: 30
    </script>
    <p><strong>Note:</strong> Please check out the browser console by pressing the f12 key on the keyboard.</p>
</body>
</html>
```

Note: Please check out the browser console by pressing the f12 key on the keyboard.

Εικόνα 12 *– Παραγωγή εξόδων χρησιμοποιώντας το console.log*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Για να γράψετε το περιεχόμενο σε ένα τρέχον έγγραφο μόνο κατά την αποδόμησή του, μπορεί να χρησιμοποιηθεί η μέθοδος document.write().

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Writing into an Browser Window with JavaScript</title>
</head>
<body>
    <script>
    // Printing a simple text message
    document.write("Hello World!"); // Prints: Hello World!

    // Printing a variable value
    var x = 10;
    var y = 20;
    var sum = x + y;
    document.write(sum); // Prints: 30
    </script>
</body>
</html>
```

Hello World!30

Εικόνα 13 *– Παραγωγή εξόδων χρησιμοποιώντας το console.log (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εάν η μέθοδος document.write() χρησιμοποιηθεί μετά τη φόρτωση της σελίδας, θα αντικαταστήσει όλο το υφιστάμενο περιεχόμενο στο έγγραφο, όπως ακολουθεί στο σύνδεσμο αυτό.

Τα πλαίσια διαλόγου ειδοποίησης (alert dialog boxes) μπορούν επίσης να προστεθούν για την εμφάνιση του μηνύματος ή των δεδομένων εξόδου στο χρήστη. Για τη δημιουργία ενός διαλόγου ειδοποιήσεων, χρησιμοποιείται η μέθοδος alert(), που έχει ως εξής:

```
1  // Displaying a simple text message
2  alert("Hello World!"); // Outputs: Hello World!
3
4  // Displaying a variable value
5  var x = 10;
6  var y = 20;
7  var sum = x + y;
8  alert(sum); // Outputs: 30
```

***Εικόνα 14** – Παραγωγή πλαισίων διαλόγου ειδοποίησης (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Οι έξοδοι μπορούν να εισαχθούν ή να γραφτούν μέσα σε ένα στοιχείο HTML χρησιμοποιώντας την ιδιότητα innerHTML. Παρ 'όλα αυτά, ο προγραμματιστής θα πρέπει να επιλέξει το στοιχείο πριν από την εγγραφή της εξόδου, χρησιμοποιώντας τη μέθοδο getElementById().

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Writing into an HTML Element with JavaScript</title>
</head>
<body>
    <p id="greet"></p>
    <p id="result"></p>

    <script>
    // Writing text string inside an element
    document.getElementById("greet").innerHTML = "Hello World!";

    // Writing a variable value inside an element
    var x = 10;
    var y = 20;
    var sum = x + y;
    document.getElementById("result").innerHTML = sum;
    </script>
</body>
</html>
```

Hello World!

30

***Εικόνα 15** – Χρησιμοποιώντας τη μέθοδο getElementById()*

(***Πηγή***: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Τύποι Δεδομένων JavaScript

Οι τύποι δεδομένων ουσιαστικά ορίζουν τι είδους δεδομένα μπορούν να αποθηκευτούν και να επεξεργασθούν μέσα σε ένα πρόγραμμα. Υπάρχουν έξι βασικοί τύποι δεδομένων στην JS, οι οποίοι μπορούν να διακριθούν σε τρεις κύριες κατηγορίες:

- Πρωτόγονοι τύποι δεδομένων (ή πρωτογενείς) – Οι συμβολοσειρές, οι αριθμοί και τα δυαδικά είναι παραδείγματα πρωτόγονων τύπων δεδομένων, τα οποία μπορούν να περιέχουν μόνο μία τιμή κάθε φορά.
- Σύνθετοι τύποι δεδομένων (ή τύποι δεδομένων αναφοράς/ reference) – Το αντικείμενο, οι συστοιχίες και οι συναρτήσεις (που είναι όλοι οι τύποι αντικειμένων) είναι σύνθετοι τύποι δεδομένων. Αυτά μπορούν να περιέχουν συλλογές τιμών και πιο σύνθετων οντοτήτων · και
- Ειδικοί τύποι δεδομένων – οι ειδικοί τύποι δεδομένων είναι απροσδιόριστοι και μηδενικοί.

Ο τύπος δεδομένων συμβολοσειράς (ακολουθία χαρακτήρων)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Χρησιμοποιείται για την ενσωμάτωση κειμενικών δεδομένων (για παράδειγμα, ακολουθίες χαρακτήρων). Οι συμβολοσειρές δημιουργούνται χρησιμοποιώντας μονά ή διπλά εισαγωγικά που περιβάλλουν έναν ή περισσότερους χαρακτήρες:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript String Data Type</title>
</head>
<body>
    <script>
    // Creating variables
    var a = 'Hi there!';  // using single quotes
    var b = "Hi there!";  // using double quotes

    // Printing variable values
    document.write(a + "<br>");
    document.write(b);
    </script>
</body>
</html>
```

Hi there!
Hi there!

Εικόνα 16 *– Ο τύπος δεδομένων συμβολοσειράς*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Θα πρέπει να σημειωθεί ότι τα εισαγωγικά μπορούν να συμπεριληφθούν και μέσα σε μια συμβολοσειρά, αλλά δεν θα πρέπει να ταιριάζουν με εκείνα που τα περικλείουν, όπως συνεπάγεται στο παράδειγμα αυτό.

Ο τύπος αριθμητικών δεδομένων

Ο τύπος αριθμητικών δεδομένων είναι χρήσιμος για την εμφάνιση θετικών ή αρνητικών αριθμών με ή χωρίς δεκαδικό ψηφίο, ή αριθμών που γράφονται με εκθετική σημειογραφία, για παράδειγμα: 1.5e-4 (που ισοδυναμεί με $1.5x10^{-4}$).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Number Data Type</title>
</head>
<body>
    <script>
    // Creating variables
    var a = 25;
    var b = 80.5;
    var c = 4.25e+6;
    var d = 4.25e-6;

    // Printing variable values
    document.write(a + "<br>");
    document.write(b + "<br>");
    document.write(c + "<br>");
    document.write(d);
    </script>
</body>
</html>
```

```
25
80.5
4250000
0.00000425
```

Εικόνα 17 *– Ο τύπος αριθμητικών δεδομένων*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ο τύπος αριθμητικών δεδομένων περιλαμβάνει επίσης ορισμένες ειδικές τιμές που είναι: Infinity, -Infinity και NaN. Το άπειρο αντιπροσωπεύει το μαθηματικό άπειρο (∞), το οποίο είναι μεγαλύτερο από οποιονδήποτε αριθμό. Το άπειρο είναι το αποτέλεσμα της διαίρεσης ενός μη μηδενικού αριθμού με το 0, όπως θα μπορούσε να ελεγχθεί στην *Εικόνα 18*:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Infinity</title>
</head>
<body>
    <script>
    document.write(16 / 0);
    document.write("<br>");
    document.write(-16 / 0);
    document.write("<br>");
    document.write(16 / -0);
    </script>
</body>
</html>
```

```
Infinity
-Infinity
-Infinity
```

Εικόνα 18 *– Ο τύπος αριθμητικών δεδομένων*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ωστόσο, το NaN (Not a Number) αντιπροσωπεύει μια ειδική τιμή μη-αριθμού. Είναι αποτέλεσμα μιας μη έγκυρης ή απροσδιόριστης μαθηματικής πράξης, για παράδειγμα, παίρνοντας την τετραγωνική ρίζα του -1 ή διαιρώντας το 0 με το 0, κ.λπ.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο Τύπος Δεδομένων Αλήθειας ή Boolean

Δύο τιμές μπορούν να διατηρηθούν σε αυτόν τον τύπο δεδομένων: σωστό (true) ή λάθος (false). Χρησιμοποιείται κλασικά σε τιμές αποθέματος όπως "yes" (σωστό) ή "no" (λάθος), on (σωστό) ή off (λάθος) κ.λπ. ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Boolean Data Type</title>
</head>
<body>
    <script>
    // Creating variables
    var isReading = true;   // yes, I'm reading
    var isSleeping = false; // no, I'm not sleeping

    // Printing variable values
    document.write(isReading + "<br>");
    document.write(isSleeping);
    </script>
</body>
</html>
```

```
true
false
```

Εικόνα 19 *– Ο τύπος δεδομένων Boolean*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Οι δυαδικές τιμές Boolean προκύπτουν επίσης ως αποτέλεσμα συγκρίσεων σε ένα πρόγραμμα. Αυτό το παράδειγμα συσχετίζει δύο μεταβλητές και εμφανίζει το αποτέλεσμα σε ένα πλαίσιο διαλόγου ειδοποίησης.

Ο Αφηρημένος Τύπος Δεδομένων (Undefined Data Type)

Ο Αφηρημένος Τύπος Δεδομένων μπορεί να λάβει μόνο μία τιμή – η ειδική τιμή είναι απροσδιόριστη. Εάν μια μεταβλητή έχει δηλωθεί, αλλά δεν έχει οριστεί τιμή, η τιμή δηλώνεται ως απροσδιόριστη.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Undefined Data Type</title>
</head>
<body>
    <script>
    // Creating variables
    var a;
    var b = "Hello World!"

    // Printing variable values
    document.write(a + "<br>");
    document.write(b);
    </script>
</body>
</html>
```

```
undefined
Hello World!
```

Εικόνα 20 *– Ο αφηρημένος τύπος δεδομένων*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ο Τύπος Μηδενικών Δεδομένων

Αυτός είναι ένας ακόμα ειδικός τύπος δεδομένων που μπορεί να έχει μόνο μία τιμή – την μηδενική τιμή. Μια μηδενική τιμή σημαίνει ότι απλά δεν υπάρχει τιμή. Δεν είναι ισοδύναμη με μια κενή συμβολοσειρά ("") ή με το μηδέν, αλλά αποτελεί αμιγώς τίποτα. Μια μεταβλητή μπορεί σαφώς να αδειάσει από το τρέχον περιεχόμενό της μέσω της εκχώρησης σ' αυτήν της **μηδενικής τιμής**.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Null Data Type</title>
</head>
<body>
    <script>
    var a = null;
    document.write(a + "<br>"); // Print: null

    var b = "Hello World!"
    document.write(b + "<br>"); // Print: Hello World!

    b = null;
    document.write(b) // Print: null
    </script>
</body>
</html>
```

```
null
Hello World!
null
```

Εικόνα 21 *– Ο τύπος δεδομένων μηδενικής τιμής*

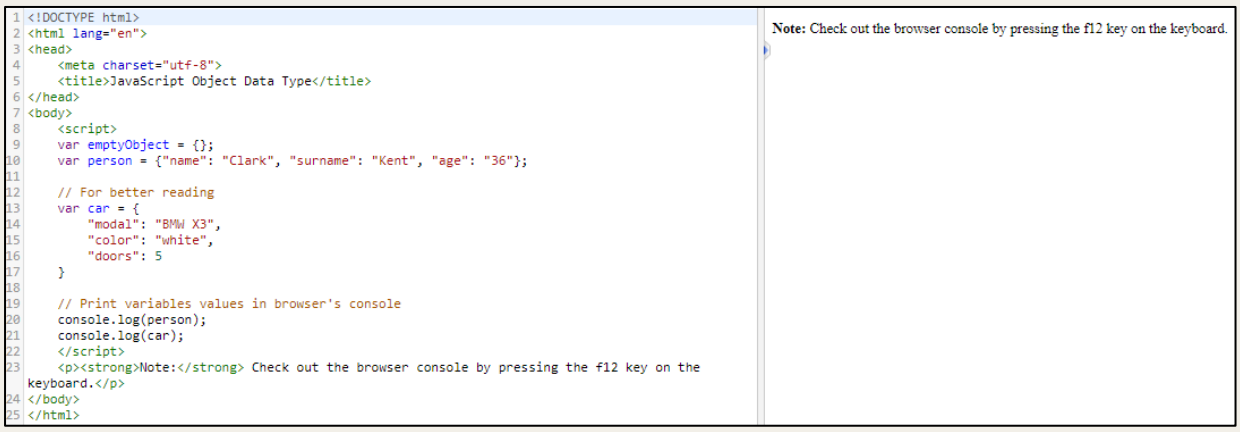
*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο Τύπος Δεδομένων Αντικειμένου

Το αντικείμενο είναι ένας πολύπλευρος τύπος δεδομένων που επιτρέπει την αποθήκευση συλλογών δεδομένων.

Ένα αντικείμενο περιέχει ιδιότητες, που ορίζονται ως ζεύγος τιμής-κλειδιού. Μια ιδιότητα key (όνομα) είναι πάντα μια συμβολοσειρά, αλλά η τιμή μπορεί να είναι οποιοσδήποτε τύπος δεδομένων (συμβολοσειρές, αριθμοί, δυαδικές τιμές, ή σύνθετοι τύποι δεδομένων όπως πίνακες, συναρτήσεις και άλλα αντικείμενα). Πιο κάτω παρατίθεται ο απλούστερος τρόπος για να δημιουργήσετε ένα αντικείμενο στο JavaScript:



***Εικόνα 22** – Ο απλούστερος τρόπος δημιουργίας ενός αντικειμένου στο JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Τα εισαγωγικά γύρω από την ιδιότητα ονόματος μπορούν να παραλειφθούν εάν το όνομα είναι έγκυρο όνομα JS. Αυτό σημαίνει ότι τα εισαγωγικά χρειάζεται να τοποθετούνται γύρω από το "first-name", αλλά δεν είναι υποχρεωτικό να τοποθετούνται γύρω από το firstname. Έτσι, το αντικείμενο αυτοκινήτου στο *Σχήμα 22* μπορεί επίσης να γραφτεί ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Object Properties Names without Quotes</title>
</head>
<body>
    <script>
    var car = {
        modal: "BMW X3",
        color: "white",
        doors: 5
    }

    // Print variable value in browser's console
    console.log(car);
    </script>
    <p><strong>Note:</strong> Check out the browser console by pressing the f12 key on the keyboard.</p>
</body>
</html>
```

Note: Check out the browser console by pressing the f12 key on the keyboard.

Εικόνα 23 *– Επανεγγραφή του αντικειμένου αυτοκινήτου από την Εικ. 22 (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ο τύπος δεδομένων συστοιχιών

Μια συστοιχία (array) είναι ένας τύπος αντικειμένου που χρησιμοποιείται για τη εκχώρηση πολλαπλών τιμών σε μία μεταβλητή. Κάθε τιμή (επίσης γνωστή και ως «στοιχείο») σε μια συστοιχία έχει μια αριθμητική θέση, γνωστή και ως ο δείκτης της (index), και μπορεί να περιλαμβάνει δεδομένα οποιουδήποτε τύπου δεδομένων- αριθμών, συμβολοσειρές, δυαδικές τιμές, συναρτήσεις, αντικείμενα, ακόμη και άλλες συστοιχίες. Ο δείκτης συστοιχίας ξεκινά από το 0 (μηδέν), έτσι ώστε το πρώτο στοιχείο συστοιχίας να είναι η συστοιχία arr[0] και όχι η συστοιχία arr[1].

Ο απλούστερος τρόπος για να φτιάξετε μια συστοιχία είναι να ορίσετε τα στοιχεία της συστοιχίας ως μια λίστα χωρισμένη με κόμματα που περικλείεται από αγκύλες, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Array Data Type</title>
</head>
<body>
    <script>
    // Creating arrays
    var colors = ["Red", "Yellow", "Green", "Orange"];
    var cities = ["London", "Paris", "New York"];

    // Printing array values
    document.write(colors[0] + "<br>");    // Output: Red
    document.write(cities[2]);    // Output: New York
    </script>
</body>
</html>
```

Red
New York

Εικόνα 24 *– Επανεγγραφή του αντικειμένου αυτοκινήτου από την Εικ. 22 (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ο Τύπος Δεδομένων Συνάρτησης

Μια συνάρτηση είναι ένα αντικείμενο που μπορεί να κληθεί για να παράξει ένα κώδικα μπλοκ. Οι συναρτήσεις είναι αντικείμενα, έτσι είναι δυνατόν να τις αντιστοιχίσουμε σε μεταβλητές, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Function Data Type</title>
</head>
<body>
    <script>
    var greeting = function(){
        return "Hello World!";
    }

    // Check the type of greeting variable
    document.write(typeof greeting) // Output: function
    document.write("<br>");
    document.write(greeting());     // Output: Hello World!
    </script>
</body>
</html>
```

```
function
Hello World!
```

Εικόνα 25 *– Ο τύπος δεδομένων συνάρτησης (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Στην πραγματικότητα, οι συναρτήσεις μπορούν να χρησιμοποιηθούν οπουδήποτε, και άλλες τιμές μπορούν επίσης να χρησιμοποιηθούν. Μπορούν να αποθηκευτούν σε μεταβλητές, αντικείμενα και συστοιχίες. Οι συναρτήσεις μπορούν να εγκριθούν/ να περάσουν ως ορίσματα/παράμετροι σε άλλες συναρτήσεις, και μπορούν επίσης να επιστραφούν ως αποτέλεσμα από τις συναρτήσεις.

Ο Τελεστής typeof

Ο Τελεστής typeof μπορεί να χρησιμοποιηθεί για να αναγνωρίσει τι είδους δεδομένα καλύπτει μια μεταβλητή. Μπορεί να χρησιμοποιηθεί με ή χωρίς παρενθέσεις [typeof(x) ή typeof x].

Ο Τελεστής typeof είναι ως επί το πλείστων επωφελής για την επεξεργασία των τιμών διαφόρων τύπων με διαφορετικό τρόπο. Ωστόσο, ο προγραμματιστής θα πρέπει να είναι

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

προσεκτικός, καθώς μπορεί να παράγει απρόβλεπτα αποτελέσματα σε ορισμένες περιπτώσεις:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript typeof Operator</title>
</head>
<body>
    <script>
    // Numbers
    document.write(typeof 15 + "<br>");  // Prints: "number"
    document.write(typeof 42.7 + "<br>");  // Prints: "number"
    document.write(typeof 2.5e-4 + "<br>");  // Prints: "number"
    document.write(typeof Infinity + "<br>");  // Prints: "number"
    document.write(typeof NaN + "<br>");  // Prints: "number". Despite being "Not-A-Number"

    // Strings
    document.write(typeof '' + "<br>");  // Prints: "string"
    document.write(typeof 'hello' + "<br>");  // Prints: "string"
    document.write(typeof '12' + "<br>");  // Prints: "string". Number within quotes is document.write(typeof string

    // Booleans
    document.write(typeof true + "<br>");  // Prints: "boolean"
    document.write(typeof false + "<br>");  // Prints: "boolean"

    // Undefined
    document.write(typeof undefined + "<br>");  // Prints: "undefined"
    document.write(typeof undeclaredVariable + "<br>"); // Prints: "undefined"

    // Null
    document.write(typeof Null + "<br>");  // Prints: "object"

    // Objects
    document.write(typeof {name: "John", age: 18} + "<br>");  // Prints: "object"

    // Arrays
    document.write(typeof [1, 2, 4] + "<br>");  // Prints: "object"

    // Functions
    document.write(typeof function(){});  // Prints: "function"
    </script>
</body>
</html>
```

```
number
number
number
number
number
string
string
string
boolean
boolean
undefined
undefined
undefined
object
object
function
```

Εικόνα 26 *– Ο τελεστής typeof (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Όπως είναι αρκετά εμφανές στο παράδειγμα από την *Εικόνα 26*, όταν γίνεται η δοκιμαστική εφαρμογή της μηδενική τιμή χρησιμοποιώντας τον τύπο τελεστή (γραμμή 22), το «αντικείμενο» (object) επιστρέφεται αντί για το «μηδενικό» (null).

Αυτό είναι ένα επαναλαμβανόμενο σφάλμα στην JavaScript, αλλά δεδομένου ότι πολλοί κώδικες στο διαδίκτυο είναι γραμμένοι γύρω από αυτή τη συμπεριφορά, και δεδομένου ότι η επιδιόρθωσή του θα δημιουργήσει πολλά περισσότερα προβλήματα, αυτό το ζήτημα αποκλείστηκε από την επιτροπή που σχεδιάζει και διατηρεί την JavaScript.

Τελεστές JavaScript

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Οι Τελεστές JavaScript είναι σύμβολα ή λέξεις-κλειδιά που ενημερώνουν την μηχανή της JavaScript για να εκτελέσει μια δεδομένη ενέργεια. Για παράδειγμα, το σύμβολο add (+) είναι ένας τελεστής που λέει στον κινητήρα JavaScript να προσθέσει δύο μεταβλητές ή τιμές, ενώ τα σύμβολα equal-to (=), greater-than (>) ή less-than (<) είναι οι τελεστές που λένε στην μηχανή της JavaScript να συγκρίνει δύο μεταβλητές ή τιμές κ.λπ.

Μεταξύ των διαφόρων τελεστών που χρησιμοποιούνται στην JavaScript, οι πρώτοι που θα περιγραφούν είναι οι **Αριθμητικοί Τελεστές** της JavaScript. Αυτοί οι αριθμητικοί τελεστές τίθενται σε λειτουργία προκειμένου να εκτελεστούν κοινές αριθμητικές πράξεις (πρόσθεση, αφαίρεση, πολλαπλασιασμός και ούτω καθεξής). Ακολουθεί ο πλήρης κατάλογος:

Τελεστής	Περιγραφή	Παράδειγμα	Αποτέλεσμα
+	Πρόσθεση	x + y	Άθροισμα των x και y
-	Αφαίρεση	x - y	Διαφορά x και y.
*	Πολλαπλασιασμός	x * y	Γινόμενο των x και y.
/	Διαίρεση	x / y	Πηλίκο των x και y
%	Όρισμα	x % y	Αποτέλεσμα του x διαιρούμενο με το y

Πίνακας 1 *– Αριθμητικοί Τελεστές (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ακολουθεί ένα πρακτικό παράδειγμα σχετικά με τον τρόπο με τον οποίο μπορούν να χρησιμοποιηθούν οι προαναφερθέντες τελεστές:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Arithmetic Operators</title>
</head>
<body>
    <script>
    var x = 10;
    var y = 4;
    document.write(x + y); // Prints: 14
    document.write("<br>");

    document.write(x - y); // Prints: 6
    document.write("<br>");

    document.write(x * y); // Prints: 40
    document.write("<br>");

    document.write(x / y); // Prints: 2.5
    document.write("<br>");

    document.write(x % y); // Prints: 2
    </script>
</body>
</html>
```

```
14
6
40
2.5
2
```

Εικόνα 27 *– Οι αριθμητικοί τελεστές JS*

(***Πηγή***: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Στη συνέχεια, υπάρχουν επίσης οι Τελεστές **Στοιχειοσειρών JS/ String Operators** (δύο από αυτούς):

Τελεστής	Περιγραφή	Παράδειγμα	Αποτέλεσμα
+	Συνένωση	str1 + str2	Συνένωση των str1 και str2
+=	Τελεστής εκχώρησης συνένωσης (Concatenation assignment operator)	str1 + str2	Προσθέτει το str2 στο str1

Πίνακας 2 *– Οι Τελεστές Στοιχειοσειρών JS*

(***Πηγή***: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Ένα πρακτικό παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript String Operators</title>
</head>
<body>
    <script>
    var str1 = "Hello";
    var str2 = " World!";

    document.write(str1 + str2 + "<br>"); // Outputs: Hello World!

    str1 += str2;
    document.write(str1); // Outputs: Hello World!
    </script>
</body>
</html>
```

```
Hello World!
Hello World!
```

Εικόνα 27 *– Οι Τελεστές Στοιχειοσειρών JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Όσον αφορά τους **Τελεστές Αύξησης και Μείωσης της JS (Incrementing and Decrementing Operators),** χρησιμοποιούνται για την αύξηση/μείωση της τιμής μιας μεταβλητής.

Τελεστής	Όνομα	Αποτέλεσμα
++x	Προ-αύξηση	Προσαυξήσεις του x κατά ένα, έπειτα επιστροφή του x (??)
++x	Μετά την προσαύξηση	Επιστροφή του x, έπειτα προσαύξηση του x κατά ένα
++x	Πριν τη μείωση	Μείωση του x κατά ένα, στη συνέχεια επιστροφή του x
++x	Μετά την μείωση	Επιστροφή του x, στη συνέχεια μείωση του x κατά ένα

Πίνακας 3 *– Τελεστές αύξησης και μείωσης JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ένα πραγματικό παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Incrementing and Decrementing Operators</title>
</head>
<body>
    <script>
    var x; // Declaring Variable

    x = 10;
    document.write(++x); // Prints: 11
    document.write("<p>" + x + "</p>");   // Prints: 11

    x = 10;
    document.write(x++); // Prints: 10
    document.write("<p>" + x + "</p>");   // Prints: 11

    x = 10;
    document.write(--x); // Prints: 9
    document.write("<p>" + x + "</p>");   // Prints: 9

    x = 10;
    document.write(x--); // Prints: 10
    document.write("<p>" + x + "</p>");   // Prints: 9
    </script>
</body>
</html>
```

```
11
11
10
11
9
9
10
9
```

Εικόνα 28 *– Τελεστές Μείωσης και Αύξησης JS*

*(**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Για τον συνδυασμό δηλώσεων υπό όρους, χρησιμοποιούνται οι **Λογικοί Τελεστές της JS (Logical Operators**).

Τελεστής	Όνομα	Παράδειγμα	Αποτέλεσμα
&&	Και	x && y	Αληθές εάν και το x και το y είναι αληθή
||	Ή	x || y	Αληθές εάν είτε το x είτε το y είναι αληθή
!	Όχι	!x	Αληθές αν το x δεν είναι ψευδές

Πίνακας 4 *– Λογικοί Τελεστές (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ιδού ένα πρακτικό παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Logical Operators</title>
</head>
<body>
    <script>
    var year = 2018;

    // Leap years are divisible by 400 or by 4 but not 100
    if((year % 400 == 0) || ((year % 100 != 0) && (year % 4 == 0))){
        document.write(year + " is a leap year.");
    } else{
        document.write(year + " is not a leap year.");
    }
    </script>
</body>
</html>
```

```
2018 is not a leap year.
```

Εικόνα 29 *– Τελεστές Μείωσης και Αύξησης JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Όσον αφορά τους **Τελεστές Σύγκρισης JS**, οι προγραμματιστές τους χρησιμοποιούν για να συγκρίνουν δύο τιμές σύμφωνα με ένα δυαδικό ή μπουλειακό συστήματος.

Τελεστής	Όνομα	Παράδειγμα	Αποτέλεσμα
==	Ίσο	x == y	Αληθές αν το x είναι ίσο με το y
===	πανομοιότυπο	x == y	Σωστό αν το x είναι ίσο με το y, και είναι του ίδιου τύπου
!=	Άνισο	x == y	Σωστό αν το x είναι ίσο με το y
!==	Δεν είναι πανομοιότυπο	x == y	Σωστό εάν το x δεν είναι ίσο με το y, ή δεν είναι του ίδιου τύπου
<	Λιγότερο από	x < y	Σωστό εάν το x είναι μικρότερο από το y
>	Μεγαλύτερο από	x > y	Σωστό εάν το x είναι μεγαλύτερο από το y
>=	Μεγαλύτερο από ή ίσο με	x >= y	Σωστό εάν το x είναι μεγαλύτερο ή ίσο με το y
<=	μικρότερο από ή ίσο με	x < y	Σωστό εάν το x είναι μεγαλύτερο ή ίσο με το y

Πίνακας 5 *– Λογικοί Τελεστές (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πώς λειτουργούν:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Comparison Operators</title>
</head>
<body>
    <script>
    var x = 25;
    var y = 35;
    var z = "25";

    document.write(x == z);  // Prints: true
    document.write("<br>");

    document.write(x === z); // Prints: false
    document.write("<br>");

    document.write(x != y);  // Prints: true
    document.write("<br>");

    document.write(x !== z); // Prints: true
    document.write("<br>");

    document.write(x < y);   // Prints: true
    document.write("<br>");

    document.write(x > y);   // Prints: false
    document.write("<br>");

    document.write(x <= y);  // Prints: true
    document.write("<br>");

    document.write(x >= y);  // Prints: false
    </script>
</body>
</html>
```

```
true
false
true
true
true
false
true
false
```

Εικόνα 30 *– Τελεστές Μείωσης και Αύξησης JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

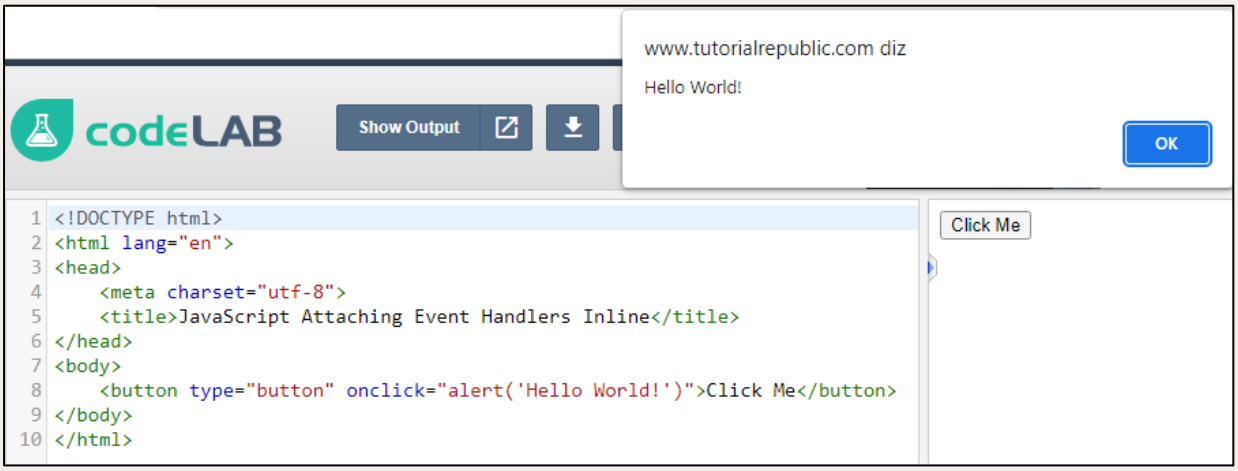
Συμβάντα Javascript (events)

Πριν μπούμε στα βαθιά του σημείου αυτού, είναι σημαντικό να αναγνωρίσουμε τι είναι ένα γεγονός στο πλαίσιο αυτό. Ένα συμβάν είναι κάτι που συμβαίνει κάθε φορά που οι χρήστες αλληλεπιδρούν με την ιστοσελίδα, όπως όταν ένας σύνδεσμος ή ένα κουμπί πατιέται, το κείμενο εισάγεται σε ένα πλαίσιο εισόδου ή περιοχή κειμένου, μια επιλογή γίνεται σε ένα πλαίσιο επιλογής, το πλήκτρο πατιέται στο πληκτρολόγιο, ο δείκτης του ποντικιού μετακινείται, μια φόρμα υποβάλλεται, και ούτω καθεξής. Κάθε τόσο, το πρόγραμμα περιήγησης είναι σε θέση να ενεργοποιήσει τα ίδια τα συμβάντα, για παράδειγμα κατά τη φόρτωση μιας σελίδας.

Όταν πραγματοποιείται ένα συμβάν, οι προγραμματιστές μπορούν να χρησιμοποιήσουν έναν διαχειριστή συμβάντων JavaScript (ή ακροατή) για να τους εντοπίσουν και να εκτελέσουν συγκεκριμένες εργασίες. Κατά σύμβαση, τα ονόματα για τους χειριστές συμβάντων ξεκινούν πάντα με τη λέξη "on", έτσι ένας χειριστής συμβάντων για το συμβάν του κλικαρίσματος του ποντικίου ονομάζεται onclick, ενώ ένας χειριστής συμβάντων για το συμβάν της φόρτωσης ονομάζεται onload, καθώς επίσης ένας χειριστής συμβάντων για το συμβάν blur (θόλωμα) ονομάζεται onblur, κ.λπ.

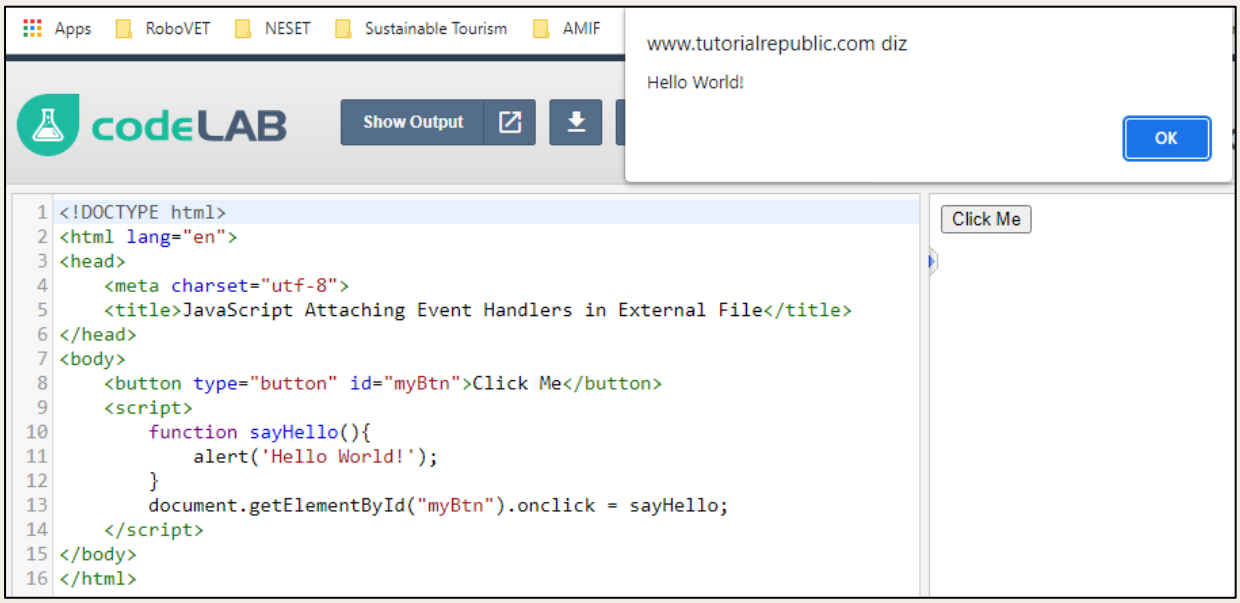
Υπάρχουν πολλοί τρόποι ανάθεσης ενός χειριστή συμβάντων. Ο απλούστερος τρόπος είναι να τα προσθέσετε απευθείας στην ετικέτα έναρξης των στοιχείων HTML, μέσω των ειδικών χαρακτηριστικών χειριστή συμβάντων. Π.χ., για να αναθέσετε ένα χειριστή κλικαρίσματος για ένα στοιχείο κουμπιού, μπορεί να χρησιμοποιηθεί το χαρακτηριστικό onclick, ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 31 *– Τελεστές Μείωσης και Αύξησης JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Παρ 'όλα αυτά, για να κρατήσουν την JavaScript αποκομμένη από την HTML, οι προγραμματιστές μπορούν να ρυθμίσουν τον χειριστή συμβάντων σε ένα εξωτερικό αρχείο JavaScript ή εντός των ετικετών < script > και</script >, ως εξής:



Εικόνα 32 *– Ανάθεση χειριστή συμβάντος (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Γενικά, τα συμβάντα μπορούν να κατηγοριοποιηθούν σε τέσσερις κύριες ομάδες — **συμβάντα ποντικιού, συμβάντα πληκτρολογίου, συμβάντα φόρμας και συμβάντα εγγράφου/παραθύρου**. Υπάρχουν πολλοί άλλοι τύποι συμβάντων, οι οποίες θα εξερευνηθούν περαιτέρω. Η επισκόπηση των πιο χρήσιμων συμβάντων θα επιχειρηθεί παρακάτω:

- **Συμβάντα ποντικιού**

Ένα συμβάν ποντικιού ενεργοποιείται όταν ο χρήστης κάνει κλικ σε κάποιο στοιχείο, μετακινεί το δείκτη του ποντικιού πάνω σε ένα στοιχείο και ούτω καθεξής. Μερικά σημαντικά γεγονότα του ποντικιού και των χειριστών των συμβάντων τους είναι οι εξής:

- **Το συμβάν κλικ (onclick):** Το συμβάν κλικ συμβαίνει όταν ένας χρήστης κάνει κλικ σε ένα στοιχείο σε μια ιστοσελίδα. Συνήθως, πρόκειται για στοιχεία φόρμας και συνδέσμων. Μπορείτε να χειριστείτε ένα συμβάν κλικ με έναν χειριστή συμβάντων onclick.

  Το ακόλουθο παράδειγμα παρουσιάζει την περίπτωση ενός μηνύματος ειδοποίησης που εμφανίζεται όταν ο χρήστης κάνει κλικ στα στοιχεία.
- **Το Contextmenu Event (oncontextmenu):** συμβαίνει όταν οι χρήστες κάνουν κλικ στο δεξί κουμπί του ποντικιού σε ένα στοιχείο, ανοίγοντας ένα μενού περιεχομένων. Ο χειριστής συμβάντων Oncontextmenu χειρίζεται ένα συμβάν μενού συμφραζομένων. Αυτό το παράδειγμα εμφανίζει ένα μήνυμα ειδοποίησης όταν οι χρήστες κάνουν δεξί κλικ στα στοιχεία.
- **Το συμβάν του ποντικιού (onmouseover):** συμβαίνει όταν οι χρήστες μετακινούν το δείκτη του ποντικιού πάνω από ένα στοιχείο. Μπορεί να αντιμετωπιστεί με τον χειριστή συμβάντων onmouseover. Το ακόλουθο παράδειγμα εμφανίζει ένα μήνυμα ειδοποίησης όταν το ποντίκι τοποθετείται πάνω από τα στοιχεία.
- **Το συμβάν Mouseout (onmouseout):** λαμβάνει χώρα όταν οι χρήστες μετακινούν τον δείκτη του ποντικιού έξω από ένα στοιχείο. Μπορείτε να το χειριστείτε χρησιμοποιώντας τον χειριστή συμβάντων onmouseout. Το ακόλουθο παράδειγμα θα σας εμφανίσει ένα μήνυμα ειδοποίησης όταν πραγματοποιηθεί το συμβάν του mouseout.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- **Συμβάντα Πληκτρολογίου**

Ένα συμβάν πληκτρολογίου λαμβάνει χώρα όταν ο χρήστης πατήσει ή απελευθερώσει ένα πλήκτρο στο πληκτρολόγιο. Μερικά από τα πιο σημαντικά συμβάντα πληκτρολογίου και οι χειριστές των συμβάντων τους είναι τα εξής:

- **Το Συμβάν Keydown (onkeydown):** συμβαίνει όταν οι χρήστες πατούν ένα πλήκτρο στο πληκτρολόγιο. Μπορείτε να το χειριστείτε χρησιμοποιώντας τον χειριστή συμβάντων onmouseout. Αυτό το παράδειγμα εμφανίζει ένα μήνυμα ειδοποίησης κάθε φορά που πραγματοποιείται ένα συμβάν κλειδιού.
- **The Συμβάν Keyup (onkeyup):** συμβαίνει όταν οι χρήστες απελευθερώνουν ένα πλήκτρο στο πληκτρολόγιο. ,το οποίο χειρίζεται ένας χειριστής συμβάντων onkeyup. Αυτό το παράδειγμα εμφανίζει ένα μήνυμα ειδοποίησης όταν συμβαίνει.
- **The Συμβάν Keypress (onkeypress):** συμβαίνει όταν ένας χρήστης πατήσει ένα πλήκτρο στο πληκτρολόγιο που έχει μια τιμή χαρακτήρα που συνδέεται με αυτό. Π.χ., Ctrl, Shift, Alt, Esc, Arrow keys κ.λπ. δεν θα δημιουργήσει ένα συμβάν keypress, αλλά θα δημιουργήσουν ένα συμβάν keydown και keyup. Ο χειριστής συμβάντων onkeypress χειρίζεται ένα συμβάν keypress. Μπορεί να ελεγχθεί εδώ μέσω ενός μηνύματος συναγερμού.

- **Συμβάντα Φόρμας (Form Event)**

Ένα συμβάν φόρμας ενεργοποιείται μόλις ένας έλεγχος φόρμας λάβει ή χάσει εστίαση ή όταν ο χρήστης τροποποιήσει μια τιμή ελέγχου φόρμας (π.χ. πληκτρολογώντας κείμενο σε μια είσοδο κειμένου), ή πατήσει μια επιλογή σε ένα πλαίσιο επιλογών κ.λπ.

- **Το Focus Event (onfocus):** συμβαίνει όταν οι χρήστες δίνουν έμφαση σε ένα στοιχείο σε μια ιστοσελίδα. και τον διαχειρίζεται ένας χειριστής συμβάντων εστίασης. Αυτό το παράδειγμα θα επισημάνει το φόντο της

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

εισαγωγής κειμένου σε κίτρινο χρώμα όταν ένας χρήστης εστιάσει πάνω σ' αυτό.

- **Το συμβάν θόλωσης (onblur):** συμβαίνει όταν ο χρήστης απομακρύνει την εστίαση από ένα παράθυρο ή στοιχείο φόρμας. Αυτό το διαχειρίζεται ένας χειριστής συμβάντων onblur. Το ακόλουθο παράδειγμα θα εμφανίσει ένα μήνυμα ειδοποίησης μόλις το στοιχείο εισαγωγής κειμένου χάσει εστίαση.
- **Το συμβάν αλλαγής (onchange):** συμβαίνει μόλις ένας χρήστης τροποποιήσει την τιμή ενός στοιχείου φόρμας. Χειριστής συμβάντων: onchange. Αυτό το παράδειγμα εμφανίζει ένα μήνυμα ειδοποίησης όταν αλλάζει η επιλογή στο πλαίσιο επιλογής.
- **Το συμβάν υποβολής (onsubmit):** συμβαίνει μόνο όταν ο χρήστης υποβάλλει μια φόρμα σε μια ιστοσελίδα. Χειριστής συμβάντων: onsubmit. Αυτό το παράδειγμα παρουσιάζει ένα μήνυμα ειδοποίησης όταν υποβάλλεται η φόρμα.

- **Γεγονότα εγγράφου/παραθύρου**

Καταστάσεις στις οποίες η σελίδα έχει φορτώσει ή αλλάξει το μέγεθος του παραθύρου του προγράμματος περιήγησης μπορούν επίσης να προκαλέσουν συμβάντα.

- **Το Γεγονός Φόρτωσης (onload):** συμβαίνει όταν μια ιστοσελίδα φορτώνει πλήρως σε ένα πρόγραμμα περιήγησης ιστού. Χειριστής συμβάντος: onload. Το ακόλουθο παράδειγμα θα εκπέμψει ένα μήνυμα ειδοποίησης μόλις φορτωθεί η σελίδα.
- **Το συμβάν μεταφόρτωσης (onunload):** όταν ένας χρήστης αποχωρήσει από την παρούσα ιστοσελίδα, αυτό το γεγονός λαμβάνει χώρα. Χειριστής συμβάντων: on unload. Αυτό το παράδειγμα εμφανίζει ένα αναδυόμενο μήνυμα ειδοποίησης όταν ο χρήστης προσπαθεί να αποχωρήσει από τη σελίδα.
- **Το γεγονός αλλαγής μεγέθους/ resize event (onresize):** συμβαίνει όταν ένας διαδικτυακός χρήστης αλλάζει μέγεθος, ελαχιστοποιεί ή μεγιστοποιεί το παράθυρο του προγράμματος περιήγησης. Χειριστής

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

συμβάντων: onresize. Αυτό το παράδειγμα εμφανίζει ένα μήνυμα ειδοποίησης αμέσως μόλις ο χρήστης αλλάξει το μέγεθος του παραθύρου του προγράμματος περιήγησης.

Συμβολοσειρές JavaScript

Στην JavaScript, οι συμβολοσειρές διαδραματίζουν βασικό ρόλο στη συνολική δομή μιας ιστοσελίδας, καθώς είναι μια ακολουθία γραμμάτων, αριθμών, ειδικών χαρακτήρων και αριθμητικών τιμών ή ακόμα και ένας συνδυασμός όλων. Μπορούν να δημιουργηθούν με την περιτύλιξη της συμβολοσειράς κειμένου (δηλαδή χαρακτήρες συμβολοσειράς) είτε μέσα σε μονά εισαγωγικά (') ή διπλά εισαγωγικά ("), ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Creating Strings in JavaScript</title>
</head>
<body>
    <script>
    // Creating variables
    var myString = 'Hello World!'; // Single quoted string
    var myString = "Hello World!"; // Double quoted string

    // Printing variable values
    document.write(myString + "<br>");
    document.write(myString);
    </script>
</body>
</html>
```

Hello World!
Hello World!

Εικόνα 33 *– Παράδειγμα συμβολοσειρών JavaScript (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Τα εισαγωγικά μπορούν να χρησιμοποιηθούν μέσα σε μια συμβολοσειρά, αλλά δεν πρέπει να ταιριάζουν με τα εισαγωγικά που περιβάλλουν τη συμβολοσειρά:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Using Quotes inside JavaScript Strings</title>
</head>
<body>
    <script>
    // Creating variables
    var str1 = "it's okay";
    var str2 = 'He said "Goodbye"';
    var str3 = "She replied 'Calm down, please'";

    // Printing variable values
    document.write(str1 + "<br>");
    document.write(str2 + "<br>");
    document.write(str3);
    </script>
</body>
</html>
```

```
it's okay
He said "Goodbye"
She replied 'Calm down, please'
```

Εικόνα 34 *– Παράδειγμα συμβολοσειρών JavaScript II (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Παρ 'όλα αυτά, τα μονά εισαγωγικά μπορούν ακόμα να τοποθετηθούν μέσα σε συμβολοσειρές με μονά εισαγωγικά ενώ τα διπλά εισαγωγικά σε συμβολοσειρές με διπλά εισαγωγικά, διαχωρίζοντας τα εισαγωγικά με ένα χαρακτήρα backlash (\), όπως φαίνεται στο *Σχήμα 36*. Το backlash ορίζεται ως ένας **χαρακτήρας διαφυγής** και οι ακολουθίες \' and \" είναι **ακολουθίες διαφυγής**.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Escaping Quotes inside JavaScript Strings</title>
</head>
<body>
    <script>
    // Creating variables
    var str1 = 'it\'s okay';
    var str2 = "He said \"Goodbye\"";
    var str3 = 'She replied \'Calm down, please\'';

    // Printing variable values
    document.write(str1 + "<br>");
    document.write(str2 + "<br>");
    document.write(str3);
    </script>
</body>
</html>
```

```
it's okay
He said "Goodbye"
She replied 'Calm down, please'
```

Εικόνα 35 *– Ο χαρακτήρας διαφυγής backlash (\) (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Οι ακολουθίες διαφυγής είναι επίσης χρήσιμες για την προσθήκη χαρακτήρων που δεν μπορούν να εισαχθούν μέσω ενός πληκτρολογίου. Μερικές άλλες ακολουθίες διαφυγής που χρησιμοποιούνται συχνότερα είναι:

- Το \n αντικαθίσταται από το χαρακτήρα νέας γραμμής (newline charater)
- \t αντικαθίσταται από τον χαρακτήρα καρτέλας (tab character)
- το\r αντικαθίσταται από το χαρακτήρα επιστροφής φορέα (carriage-return character)
- Το \b αντικαθίσταται από τον χαρακτήρα backspace
- Το \\ αντικαθίσταται από ένα χαρακτήρα backlash (\)

Η Εικόνα 36 διευκρινίζει πώς λειτουργούν οι ακολουθίες διαφυγής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Escape Sequences</title>
</head>
<body>
    <script>
    // Creating variables
    var str1 = "The quick brown fox \n jumps over the lazy dog.";
    document.write("<pre>" + str1 + "</pre>"); // Create line break

    var str2 = "C:\Users\Downloads";
    document.write(str2 + "<br>"); // Prints C:UsersDownloads

    var str3 = "C:\\Users\\Downloads";
    document.write(str3); // Prints C:\Users\Downloads
    </script>
</body>
</html>
```

```
The quick brown fox
 jumps over the lazy dog.

C:UsersDownloads
C:\Users\Downloads
```

Εικόνα 36 *– Πώς λειτουργούν οι ακολουθίες διαφυγής (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Εκτέλεση λειτουργιών σε συμβολοσειρές

Η JavaScript καθιστά διαθέσιμες διάφορες ιδιότητες και μεθόδους για να εκτελεί λειτουργίες στις τιμές των συβμολοσειρών. Πιο συγκεκριμένα, μόνο τα αντικείμενα μπορούν να έχουν ιδιότητες και μεθόδους. Ωστόσο, στην JavaScript, οι πρωτόγονοι τύποι δεδομένων μπορούν να εκτελέσουν λειτουργίες όπως και τα αντικείμενα, όταν ο προγραμματιστής αναφέρεται σε αυτά με την ιδιότητα σημειογραφίας πρόσβασης (property access notation). Η JavaScript προσφέρει αυτή τη δυνατότητα μέσω της

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

δημιουργίας ενός provisional wrapper object για πρωτόγονους τύπους δεδομένων. Αυτή η διαδικασία γίνεται αυτόματα από τον διερμηνέα JS στο παρασκήνιο.

Λήψη του μήκους μιας συμβολοσειράς

Η ιδιότητα μήκους επιστρέφει το μήκος μιας συμβολοσειράς, το οποίο είναι ο αριθμός των χαρακτήρων που οριοθετούνται σε μια συμβολοσειρά, συμπεριλαμβανομένου του αριθμού των ειδικών χαρακτήρων επίσης, όπως \t ή \n. Οι προγραμματιστές πρέπει να είναι προσεκτικοί στη χρήση παρενθέσεων μετά το μήκος (π.χ. str.length ()), καθώς ο σωστός τρόπος είναι str.length (διαφορετικά, θα δημιουργήσει ένα σφάλμα).

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Get String Length in JavaScript</title>
</head>
<body>
    <script>
    var str1 = "This is a paragraph of text.";
    document.write(str1.length + "<br>"); // Prints 28

    var str2 = "This is a \n paragraph of text.";
    document.write(str2.length); // Prints 30, because \n is only one
character
    </script>
</body>
</html>
```

Εικόνα 37 *– Πώς να λάβετε το μήκος μιας συμβολοσειράς (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εύρεση συμβολοσειράς μέσα σε άλλη συμβολοσειρά

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Find the Position of Substring within a String</title>
</head>
<body>
    <script>
    var str = "If the facts don't fit the theory, change the facts.";
    var pos = str.indexOf("facts");
    document.write(pos); // 0utputs: 7
    </script>
</body>
</html>
```

7

Εικόνα 38 *– Εύρεση συμβολοσειράς μέσα σε άλλη συμβολοσειρά (**Πηγή**:*
https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Παρομοίως, η τεχνική lastIndexOf () μπορεί να χρησιμοποιηθεί για να λάβει το δείκτη ή τη θέση της τελευταίας εμφάνισης της καθορισμένης συμβολοσειράς μέσα σε μια συμβολοσειρά, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Find the Position of Substring within a String</title>
</head>
<body>
    <script>
    var str = "If the facts don't fit the theory, change the facts.";
    var pos = str.lastIndexOf("facts");
    document.write(pos); // 0utputs: 46
    </script>
</body>
</html>
```

46

Εικόνα 39 *– Εύρεση συμβολοσειράς μέσα σε άλλη συμβολοσειρά χρησιμοποιώντας τη μέθοδολastIndexOf () (**Πηγή**:*
https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Τόσο οι προσεγγίσεις του indexOf(), όσο και του lastIndexOf () μπορούν να επιστραφούν -1 εάν η υποσυμβολοσειρά δεν βρεθεί. Και οι δύο μέθοδοι επίσης δέχονται μια προαιρετική ακέραια παράμετρο η οποία ορίζει τη θέση εντός της συμβολοσειράς στην οποία θα ξεκινήσει η αναζήτηση.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αναζήτηση ενός μοτίβου μέσα σε μια συμβολοσειρά

Η μέθοδος αναζήτησης() μπορεί να χρησιμοποιηθεί για την αναζήτηση ενός συγκεκριμένου κομματιού κειμένου ή μοτίβου μέσα σε μια συμβολοσειρά. Καθώς η προσέγγιση indexOf (), η search() επιστρέφει επίσης το ευρετήριο της πρώτης αντιστοίχισης, και επιστρέφει το -1 no matches were found, αλλά σε αντίθεση με το indexOf (), το search () μπορεί επίσης να λάβει μια *κανονική έκφραση* ως όρισμα/παράμετρο για την παροχή προηγμένων δεξιοτήτων αναζήτησης. Θα πρέπει να αναφερθεί ότι η προσέγγιση αναζήτησης() δεν υποστηρίζει παγκόσμιες αναζητήσεις, καθώς αγνοεί τη σημαία g (flag) ή τον τροποποιητή του ορίσματος της κανονικής έκφρασης.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Search Text or Pattern inside a String</title>
</head>
<body>
    <script>
    var str = "Color red looks brighter than color blue.";

    // Case sensitive search
    var pos1 = str.search("color");
    document.write(pos1 + "<br>"); // 0utputs: 30

    // Case insensitive search using regexp
    var pos2 = str.search(/color/i);
    document.write(pos2); // 0utputs: 0
    </script>
</body>
</html>
```

```
30
0
```

Εικόνα 40 *– Αναζήτηση μοτίβου μέσα σε μια συμβολοσειρά (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Εξαγωγή μιας υποσυμβολοσειράς από μια συμβολοσειρά

Για να αφαιρέσετε ένα μέρος ή υποσυμβολοσειρά από μια συμβολοσειρά, μπορείτε να χρησιμοποιήσετε την μέθοδο slice (). Αυτή λαμβάνει δύο παραμέτρους: *αρχικός δείκτης*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

(δείκτης όπου εκκινεί την εξαγωγή), και ένας προαιρετικός *τελικός δείκτης (δείκτης* πριν από τον οποίο τελειώνει η εξαγωγή), όπως το str.slice(startIndex, endIndex).

Το παρακάτω παράδειγμα χωρίζει ένα τμήμα μιας συμβολοσειράς από τη θέση 4 στη θέση 15:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Slice Out a Portion of a String</title>
</head>
<body>
    <script>
    var str = "The quick brown fox jumps over the lazy dog.";
    var subStr = str.slice(4, 15);
    document.write(subStr); // Prints: quick brown
    </script>
</body>
</html>
```

quick brown

Εικόνα 41 *– Εξαγωγή μιας υποσυμβολοσειράς από μια συμβολοσειρά χρησιμοποιώντας τη μέθοδο slice() (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Μπορούν επίσης να καθοριστούν αρνητικές τιμές. Αυτές οι τιμές επεξεργάζονται ως strLength + startIndex, όπου το strLength είναι το μήκος της συμβολοσειράς (str.length), για παράδειγμα, αν το startIndex είναι -5 θα επεξεργασθεί ως strLength - 5. Αν το startIndex είναι μεγαλύτερο ή ίσο με το μήκος της συμβολοσειράς, η μέθοδος slice() επιστρέφει μια κενή συμβολοσειρά. Αντίστοιχα, εάν το προαιρετικό endIndex δεν έχει καθοριστεί ή παραλειφθεί, η μέθοδος slice() εξάγεται στο τέλος της συμβολοσειράς.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Slice Strings Using Negative Indexes</title>
</head>
<body>
    <script>
    var str = "The quick brown fox jumps over the lazy dog.";
    document.write(str.slice(-28, -19) + "<br>"); // Prints: fox jumps
    document.write(str.slice(31)); // Prints: the lazy dog.
    </script>
</body>
</html>
```

fox jumps
the lazy dog.

Εικόνα 42 *– Καθορισμός αρνητικών τιμών (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η μέθοδος subring () για την εξαγωγή ενός τμήματος της δοσμένης συμβολοσειράς με βάση τους δείκτες έναρξης και τέλους, ως str.subring(startIndex, endIndex). Η μέθοδος υποσυμβολοσειράς() είναι κατά πολύ συγκρίσιμη με την τομή() (slice), εκτός από ορισμένες διαφορές:

- Εάν οποιοδήποτε από τα δύο ορίσματα είναι μικρότερο από 0 ή είναι NaN, τότε αντιμετωπίζεται ως 0.
- Εάν οποιοδήποτε από τα δύο ορίσματα είναι μεγαλύτερο από το str.length, αντιμετωπίζεται σαν να ήταν str.length.
- Αν το startIndex είναι μεγαλύτερο από toendIndex, τότε το subring () θα αλλάξει αυτά τα δύο ορίσματα, δηλαδή το str.substring (5, 0) ==str.substring (0, 5).

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <title>JavaScript Extract substring from a String</title>
6  </head>
7  <body>
8      <script>
9      var str = "The quick brown fox jumps over the lazy dog.";
10     document.write(str.substring(4, 15) + "<br>"); // Prints: quick brown
11     document.write(str.substring(9, 0) + "<br>"); // Prints: The quick
12     document.write(str.substring(-28, -19) + "<br>"); // Prints nothing
13     document.write(str.substring(31)); // Prints: the lazy dog.
14     </script>
15 </body>
16 </html>
```

quick brown
The quick

the lazy dog.

Εικόνα 43 *– Η μέθοδος υποσυμβολοσειράς() για την εξαγωγή ενός τμήματος της δοσμένης συμβολοσειράς με βάση τους δείκτες έναρξης και τέλους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Εξαγωγή συγκεκριμένου αριθμού χαρακτήρων από μια συμβολοσειρά

Η JavaScript παρέχει επίσης την τεχνική subr (), η οποία είναι παρόμοια με την slice() με μια μικρή διαφορά: η δεύτερη παράμετρος ορίζει τον αριθμό των χαρακτήρων που θα εξαχθούν αντί για τον τελικό δείκτη, ως str.substr (startIndex, length). Αν το μήκος είναι 0 ή αρνητικός αριθμός, επιστρέφεται μια κενή συμβολοσειρά.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Extract Fixed Number of Characters from a String</title>
</head>
<body>
    <script>
    var str = "The quick brown fox jumps over the lazy dog.";
    document.write(str.substr(4, 15) + "<br>"); // Prints: quick brown fox
    document.write(str.substr(-28, -19) + "<br>"); // Prints nothing
    document.write(str.substr(-28, 9) + "<br>"); // Prints: fox jumps
    document.write(str.substr(31)); // Prints: the lazy dog.
    </script>
</body>
</html>
```

quick brown fox

fox jumps
the lazy dog.

Εικόνα 44 *– Εξαγωγή σταθερού αριθμού χαρακτήρων από μια συμβολοσειρά (**Πηγή**:* https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Αντικατάσταση των περιεχομένων μιας συμβολοσειράς

Η μέθοδος replace() χρησιμοποιείται για την αντικατάσταση μέρους μιας συμβολοσειράς με άλλη συμβολοσειρά. Αυτή η προσέγγιση παίρνει μια κανονική έκφραση για να ταιριάξει η υποσυμβολοσειρά με δύο παραμέτρους και μια συμβολοσειρά αντικατάστασης, δηλαδή str.replace (regexp|subr, newSubstr). Η μέθοδος replace() επιστρέφει μια νέα συμβολοσειρά και δεν ταράζει την αρχική συμβολοσειρά, η οποία θα παραμείνει ανεπηρέαστη.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Replace Part of a String with another String</title>
</head>
<body>
    <script>
    var str = "Color red looks brighter than color blue.";
    var result = str.replace("color", "paint");
    document.write(result); // 0utputs: Color red looks brighter than paint blue.
    </script>
</body>
</html>
```

Color red looks brighter than paint blue.

Εικόνα 45 *– Αντικατάσταση των περιεχομένων μιας συμβολοσειράς χρησιμοποιώντας την τεχνική replace() (**Πηγή**:* https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Από προεπιλογή, η μέθοδος replace() υποκαθιστά μόνο την πρώτη αντιστοίχιση και ;έχει διάκριση πεζών/ κεφαλαίων. Για να αντικαταστήσετε την υποσυμβολοσειρά μέσα σε μια συμβολοσειρά με έναν τρόπο που δεν επηρεάζεται από πεζά/κεφαλαία γράμματα, μπορεί να χρησιμοποιηθεί μια *κανονική έκφραση (regexp)* με έναν τροποποιητή i, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Replace Part of a String with another String</title>
</head>
<body>
    <script>
    var str = "Color red looks brighter than color blue.";
    var result = str.replace(/color/i, "paint");
    document.write(result); // 0utputs: paint red looks brighter than color
blue.
    </script>
</body>
</html>
```

```
paint red looks brighter than color blue.
```

Εικόνα 46 *– Αντικατάσταση των περιεχομένων μιας συμβολοσειράς χρησιμοποιώντας την τεχνική replace() (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Επίσης, για να αντικατασταθούν όλες οι περιπτώσεις μιας υποσυμβολοσειράς μέσα σε μια συμβολοσειρά με έναν τρόπο που δεν επηρεάζεται από πεζά/κεφαλαία γράμματα, μπορεί να χρησιμοποιηθεί ο τροποποιητής g μαζί με τον τροποποιητή i, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Replace All Occurrences of a Substring in a
String</title>
</head>
<body>
    <script>
    var str = "Color red looks brighter than color blue.";
    var result = str.replace(/color/ig, "paint");
    document.write(result); // 0utputs: paint red looks brighter than paint
blue.
    </script>
</body>
</html>
```

```
paint red looks brighter than paint blue.
```

Εικόνα 47 *– Αντικατάσταση όλων των περιστατικών μιας υποσυμβολοσειράς μέσα σε μια συμβολοσειρά με τρόπο που δεν επηρεάζεται από πεζά/κεφαλαία γράμματα (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μετατροπή συμβολοσειράς σε κεφαλαία ή πεζά γράμματα

Η μέθοδος toUpperCase() χρησιμοποιείται για τη μετατροπή μιας συμβολοσειράς σε κεφαλαία, ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Convert a String to Uppercase Characters</title>
</head>
<body>
    <script>
    var str = "Hello World!";
    var result = str.toUpperCase();
    document.write(result); // Prints: HELLO WORLD!
    </script>
</body>
</html>
```

HELLO WORLD!

Εικόνα 48 *– Μετατροπή συμβολοσειράς σε κεφαλαία χρησιμοποιώντας τη μέθοδο UpperCase () (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Παρομοίως, η μέθοδος toLowerCase() χρησιμοποιείται για τη μετατροπή μιας συμβολοσειράς σε πεζά:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Convert a String to Lowercase Characters</title>
</head>
<body>
    <script>
    var str = "Hello World!";
    var result = str.toLowerCase();
    document.write(result); // Prints: hello world!
    </script>
</body>
</html>
```

hello world!

Εικόνα 49 *– Μετατροπή συμβολοσειράς σε πεζά χρησιμοποιώντας τη μέθοδο toLowerCase() (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Συνδυασμός δύο ή περισσότερων συμβολοσειρών

Δύο ή περισσότερες συμβολοσειρές μπορούν να συνενωθούν ή να συνδυαστούν χρησιμοποιώντας τους τελεστές ανάθεσης + και +=.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Join Two or More Strings</title>
</head>
<body>
    <script>
    var hello = "Hello";
    var world = "World";
    var greet = hello + " " + world;
    document.write(greet + "<br>"); // Prints: Hello World

    var wish  = "Happy";
        wish += " New Year";
    document.write(wish); // Prints: Happy New Year
    </script>
</body>
</html>
```

Hello World
Happy New Year

Εικόνα 50 *– Σύνδεση δύο ή περισσότερων συμβολοσειρών (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Πρόσβαση σε μεμονωμένους χαρακτήρες από μια συμβολοσειρά

Η μέθοδος CharAt () μπορεί να χρησιμοποιηθεί για την πρόσβαση μεμονωμένου χαρακτήρα από μια συμβολοσειρά, ως str.charAt(index). Ο καθορισμένος δείκτης θα πρέπει να είναι ένας αριθμός μεταξύ 0 και str.length - 1. Αν δεν δοθεί ευρετήριο, επιστρέφεται ο πρώτος χαρακτήρας στη συμβολοσειρά, αφού η προεπιλογή είναι 0.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Extract a Single Character from a String</title>
</head>
<body>
    <script>
    var str = "Hello World!";
    document.write(str.charAt() + "<br>");  // Prints: H
    document.write(str.charAt(6) + "<br>"); // Prints: W
    document.write(str.charAt(30) + "<br>"); // Prints nothing
    document.write(str.charAt(str.length - 1)); // Prints: !
    </script>
</body>
</html>
```

H
W
!

Εικόνα 51 *– Πρόσβαση σε μεμονωμένους χαρακτήρες από μια συμβολοσειρά (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Ωστόσο, υπάρχει μια καλή εναλλακτική λύση σε αυτή τη διαδικασία. Δεδομένου ότι το ECMAScript 5, οι συμβολοσειρές μπορούν να αντιμετωπιστούν ως πίνακες μόνο για

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ανάγνωση και οι μεμονωμένοι χαρακτήρες μπορούν να εμφανιστούν από μια συμβολοσειρά χρησιμοποιώντας αγκύλες ([]) αντί της προσέγγισης CharAt (), ως εξής:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Extract a Single Character from a String</title>
</head>
<body>
    <script>
    var str = "Hello World!";
    document.write(str[0] + "<br>"); // Prints: H
    document.write(str[6] + "<br>"); // Prints: W
    document.write(str[str.length - 1] + "<br>"); // Prints: !
    document.write(str[30]); // Prints: undefined
    </script>
</body>
</html>
```

```
H
W
!
undefined
```

***Εικόνα 52** – Πρόσβαση σε μεμονωμένους χαρακτήρες από μια συμβολοσειρά χρησιμοποιώντας αγκύλες ([]) αντί της προσέγγισηςCharAt () (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Διαχωρισμός μιας συμβολοσειράς σε μια συστοιχία

Η μέθοδος split() μπορεί να χρησιμοποιηθεί για να κατακερματίσει μια συμβολοσειρά σε έναν πίνακα συμβολοσειρών, χρησιμοποιώντας την σύνταξη str.split(διαχωριστικό, όρισμα). Το όρισμα διαχωριστή ορίζει τη συμβολοσειρά στην οποία θα πρέπει να συμβεί κάθε διαίρεση, ενώ τα ορίσματα ορίου καθορίζουν το μέγιστο μήκος του πίνακα. Εάν το διαχωριστικό όρισμα παραλειφθεί ή δεν βρεθεί στην καθορισμένη συμβολοσειρά, ολόκληρη η συμβολοσειρά κατανέμεται στο πρώτο στοιχείο του πίνακα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Split a String into an Array</title>
</head>
<body>
    <script>
    var fruitsStr = "Apple, Banana, Mango, Orange, Papaya";
    var fruitsArr = fruitsStr.split(", ");
    document.write(fruitsArr[0] + "<br>"); // Prints: Apple
    document.write(fruitsArr[2] + "<br>"); // Prints: Mango
    document.write(fruitsArr[fruitsArr.length - 1]); // Prints: Papaya
    document.write("<hr>");

    // Loop through all the elements of the fruits array
    for(var i in fruitsArr) {
        document.write("<p>" + fruitsArr[i] + "</p>");
    }
    </script>
</body>
</html>
```

Apple
Mango
Papaya

Apple
Banana
Mango
Orange
Papaya

Εικόνα 53 *– Διαχωρισμός μιας συμβολοσειράς σε μια συστοιχία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-get-started.php)*

Για να χωρίσετε μια συμβολοσειρά σε έναν πίνακα χαρακτήρων, μια κενή συμβολοσειρά ("") πρέπει να οριστεί ως διαχωριστικό, όπως παρουσιάζεται σε αυτό το παράδειγμα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αριθμοί JavaScript

Υπάρχουν δύο τύποι αριθμών στο JavaScript:

- **Οι κανονικοί αριθμοί** σε JavaScript αποθηκεύονται σε μορφή 64-bit IEEE-754, αλλιώς είναι γνωστοί ως «*αριθμοί κινητής υποδιαστολής διπλής ακρίβειας*». Αυτοί είναι οι πιο συνηθισμένοι τύποι αριθμών.

- **Αριθμοί BigInt**, που αντιπροσωπεύουν ακέραιους αριθμούς τυχαίου μήκους (random lenghtlength). Μερικές φορές χρειάζονται, επειδή ένας κανονικός αριθμός δεν μπορεί να υπερβαίνει με ασφάλεια τα 253 ή να είναι μικρότερος από -253.

Υπάρχουν περισσότεροι τρόποι να γράψεις έναν αριθμό. Για παράδειγμα, ο προφανής τρόπος εγγραφής 1 δισεκατομμυρίου θα ήταν "1000000000" ή "1_000_000_000", χρησιμοποιώντας την υπογράμμιση ως διαχωριστικό. Σε αυτή την περίπτωση, η υπογράμμιση είναι η «συντακτική ζάχαρη» (syntactic sugar), που σημαίνει ότι καθιστά τον αριθμό πιο ευανάγνωστο. Ο μηχανή της JS αγνοεί την υπογράμμιση μεταξύ των ψηφίων, οπότε είναι ακριβώς το ίδιο, δηλαδή το ένα δισεκατομμύριο της πρώτης περίπτωσης.
Ωστόσο, στην πραγματικότητα, όλοι σίγουρα θα προσπαθήσουν να αποφύγουν να γράψουν μεγάλες ακολουθίες μηδενικών. Κάτι σαν «1 δις» για ένα δισεκατομμύριο ή «2,5 δις» για 2 δισεκατομμύρια 500 εκατομμύρια φαίνεται πιο λογικό. Η ίδια αρχή ισχύει για τους περισσότερους μεγάλους αριθμούς. Τούτου λεχθέντος, είναι δυνατόν να συντομευθεί ένας αριθμός σε JS, προσθέτοντας το γράμμα "e" σε αυτό και καθορίζοντας την ποσότητα των μηδενικών:

```
1  let billion = 1e9;  // 1 billion, literally: 1 and 9 zeroes
2
3  alert( 7.3e9 );  // 7.3 billions (same as 7300000000 or 7_300_000_000)
```

Εικόνα 54 *– Καθορισμός της ποσότητας των μηδενικών (**Πηγή**: https://javascript.info/number)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Έτσι, το e πολλαπλασιάζει τον αριθμό με το 1 με το δεδομένο αριθμό μηδενικών.

```
1e3 === 1 * 1000; // e3 σημαίνει *1000
1.23e6 == 1.23 * 1000000; // e6 σημαίνει *1000000
```

Η ίδια αρχή ισχύει και για τους μικρούς αριθμούς. Για παράδειγμα, τι πρέπει να γραφτεί για 1 μικροδευτερόλεπτο (ένα εκατομμυριοστό του δευτερολέπτου);

```
let mcs = 0,000001 ·
```

Ακριβώς όπως η προαναφερθείσα περίπτωση με τους μεγάλους αριθμούς, χρησιμοποιώντας το "e" μπορεί να είναι χρήσιμο. Για να αποφύγετε να γράψετε τα μηδενικά ρητά, μπορούν να γραφτούν τα ακόλουθα:

```
mcs = 1e-6; // έξι μηδενικά προς τα αριστερά από 1
```

Υπάρχουν 6 μηδενικά στο 0.000001. Στη συνέχεια, δεν είναι δύσκολο να συναχθεί το συμπέρασμα 1e-6.

Επομένως, ένας αρνητικός αριθμός μετά το "e" υποδηλώνει μια διαίρεση με το 1 με το δοσμένο αριθμό μηδενικών, ως εξής:

```
1  // -3 divides by 1 with 3 zeroes
2  1e-3 === 1 / 1000; // 0.001
3
4  // -6 divides by 1 with 6 zeroes
5  1.23e-6 === 1.23 / 1000000; // 0.00000123
```

Εικόνα 55 *– Η διαίρεση του 1 με το δοσμένο αριθμό μηδενικών (**Πηγή**: https://javascript.info/number)*

Δεκαεξαδικοί, δυαδικοί και οκταδικοί αριθμοί

Οι δεκαεξαδικοί αριθμοί χρησιμοποιούνται συνήθως στην JavaScript για να ενσωματώσουν χρώματα, να κωδικοποιήσουν χαρακτήρες και για πολλούς άλλους σκοπούς. Έτσι, προφανώς, υπάρχει ένας γρηγορότερος τρόπος για να τους γράψετε,

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

πιο συγκεκριμένα προσθέτοντας το πρόθεμα (prefix) 0x (0x) ακολουθούμενο από μια ακολουθία αριθμών, ως εξής:

```
1  alert( 0xff ); // 255
2  alert( 0xFF ); // 255 (the same, case doesn't matter)
```

Εικόνα 56 *– Η προσθήκη του προθέματος 0x για την επιτάχυνση των διαδικασιών της JS (**Πηγή**: https://javascript.info/number)*

Τα δυαδικά και οκταδικά αριθμητικά συστήματα δεν χρησιμοποιούνται συχνά, αλλά και αυτά μπορούν να υποστηριχθούν με την χρήση άλλων προθεμάτων όπως το 0b και το 0o (*μηδέν*, *ο*):

```
1  let a = 0b11111111; // binary form of 255
2  let b = 0o377; // octal form of 255
3
4  alert( a == b ); // true, the same number 255 at both sides
```

Εικόνα 57 *– Τα προθέματα 0b και 0o για τα δυαδικά και οκταδικά αριθμητικά συστήματα (**Πηγή**: https://javascript.info/number)*

Μόνο τα προαναφερθέντα 3 αριθμητικά συστήματα μπορούν να υποστηριχθούν μέσω της προσθήκης προθεμάτων. Σε άλλα αριθμητικά συστήματα, μπορεί να χρησιμοποιηθεί η συνάρτηση parseInt.

Η μέθοδος toString(base)

Η μέθοδος num.toString (base) επιστρέφει μια αναπαράσταση της συμβολοσειράς του num στο αριθμητικό σύστημα με την παρεχόμενη βάση (base), ως εξής:

```
1  let num = 255;
2
3  alert( num.toString(16) );  // ff
4  alert( num.toString(2) );   // 11111111
```

Εικόνα 58 *– Η μέθοδος num.toString (base) (**Πηγή**: https://javascript.info/number)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η βάση μπορεί να κυμαίνεται από το 2 μέχρι το 36. Από προεπιλογή, είναι 10.

Υπάρχουν περιπτώσεις που η χρήση του μπορεί να γενικοποιηθεί:

- Η **Base16** εφαρμόζεται για δεκαεξαδικά χρώματα, κωδικοποιήσεις χαρακτήρων κ.λπ., και αποτελείται από ακολουθίες από αριθμούς από το 0..9 ή γράμματα από το A..F
- Το **base=2** εφαρμόζεται κυρίως για να διορθώσει δυαδικούς τελεστές, τα ψηφία μπορούν να είναι 0 ή 1.
- Το **αριθμητικό σύστημα με βάση τα 36 ψηφία** χρησιμοποιεί το μέγιστο σύνολο αριθμών ή γραμμάτων, και μπορεί να αποτελείται από τα ψηφία, από το 0..9 ή από το A..Z. Για την αναπαράσταση ενός αριθμού μπορεί να χρησιμοποιηθούν όλα τα γράμματα του λατινικού αλφαβήτου. Μια αξιοπερίεργη, αλλά χρήσιμη περίπτωση της χρήσης της base =36 είναι όταν χρειάζεται να μετατρέψετε ένα μακρύ αριθμητικό αναγνωριστικό σε κάτι πιο σύντομο, όπως για παράδειγμα, για να δημιουργήσετε μια σύντομη διεύθυνση URL.

Υπάρχει μια συγκεκριμένη αξιοσημείωτη περίπτωση. Όταν εντοπίζονται δύο τελείες στο 123456.. toString(36), αυτό δεν είναι τυπογραφικό λάθος. Όταν ο προγραμματιστής θέλει να καλέσει μια μέθοδο απευθείας σε έναν αριθμό, είναι απαραίτητο να τοποθετήσετε δύο τελείες (‘..‘) μετά από αυτό.

Εάν, όμως, τοποθετηθεί μία τελεία ως εξής [‘123456.toString(36)’], αυτό θα αποτελούσε σφάλμα, καθώς η σύνταξη JavaScript πρέπει να περιέχει το δεκαδικό τμήμα μετά την πρώτη τελεία. Αν ο προγραμματιστής τοποθετήσει μια ακόμα τελεία, τότε η JavaScript ξέρει ότι το δεκαδικό τμήμα είναι κενό και ότι τώρα δεν μπορεί να ολοκληρωθεί η μέθοδος.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στρογγυλοποίηση (Rounding)

Μία από τις πιο εφαρμοσμένες λειτουργίες κατά τη λειτουργία με αριθμούς είναι η **στρογγυλοποίηση**.

Υπάρχουν ορισμένες ενσωματωμένες λειτουργίες για τη στρογγυλοποίηση:

- **Math.floor**

Η λειτουργεία Math.floor στρογγυλοποιεί προς τα κάτω το 3.1 μετατρέποντάς το σε 3, ενώ το -1.1 το μετατρέπει σε -2.

- **Math.ceil**

Η λειτουργία Math.ceil στρογγυλοποιεί το 3.1 μετατρέποντάς το σε 4, ενώ το -1.1 το μετατρέπει σε -1.

- **Math.round**

Η λειτουργία Math.round στρογγυλοποιεί στον πλησιέστερο ακέραιο αριθμό. Για παράδειγμα το 3.1 μετατρέπεται σε 3, το 3.6 σε 4, ενώ το 3.5 (δηλαδή οι μεσαίες τιμές 0,5) στρογγυλοποιείται επίσης σε 4.

- **Math.trunc (δεν υποστηρίζεται από τον Internet Explorer)**

Εξαλείφει οτιδήποτε μετά το δεκαδικό οριοθέτη (την τελεία) χωρίς στρογγυλοποίηση. Για παράδειγμα, το 3.1 μετατρέπεται σε 3 και το -1.1 γίνεται -1.

Αυτές οι συναρτήσεις καλύπτουν όλους τους δυνητικούς τρόπους διαχείρισης του δεκαδικού μέρους ενός αριθμού. Αλλά τι συμβαίνει όταν θέλουμε να στρογγυλοποιήσουμε έναν αριθμό στη νιοστή του δύναμη μετά από τον δεκαδικό οριοθέτη;

Για παράδειγμα, το δεκαδικό μέρος του 1,2345 στρογγυλοποιείται σε δύο ψηφία (1,23).

Υπάρχουν δύο τρόποι για να γίνει αυτό:

1. **Πολλαπλασιασμός και διαίρεση.**

Π.χ., για να στρογγυλοποιηθεί το δεκαδικό μέρος ενός αριθμού μέχρι το 2ο του ψηφίο, μπορεί να πολλαπλασιαστεί με το 100 (ή με μια μεγαλύτερη δύναμη από το 10), να καλέσει τη συνάρτηση στρογγυλοποίησης και στη συνέχεια να τη διαιρέσει.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

2. Η μέθοδος **toFixed (n)** στρογγυλοποιεί τον αριθμό στη νιοστή του δύναμη (δηλαδή σε n= εκθέτης) μετά τον δεκαδικό οριοθέτη (τελεία) και επιστρέφει μια συμβολοσειρά που αναπαριστά το γινόμενο του πολλαπλασιασμού τον n παραγόντων.

```
1  let num = 12.34;
2  alert( num.toFixed(1) ); // "12.3"
```

Εικόνα 59 *– Η μέθοδος toFixed (n) (**Πηγή**: https://javascript.info/number)*

Αυτή η μέθοδος στρογγυλοποιεί τον αριθμό προς τα πάνω ή προς τα κάτω στην πλησιέστερη του τιμή, όπως στη μέθοδο Math.round:

```
1  let num = 12.36;
2  alert( num.toFixed(1) ); // "12.4"
```

Εικόνα 60 *– Η μέθοδος toFixed (n) – συνέχεια (**Πηγή**: https://javascript.info/number)*

Θα πρέπει να αναφερθεί ότι το αποτέλεσμα του toFixed είναι μια συμβολοσειρά. Εάν το δεκαδικό μέρος είναι μικρότερο απ' ό,τι απαιτείται για την αναπαράσταση μιας συμβολοσειράς, τότε προσαρτώνται στο τέλος μηδενικά:

```
1  let num = 12.34;
2  alert( num.toFixed(5) ); // "12.34000", added zeroes to make exactly 5 digits
```

Εικόνα 61 *– Η μέθοδος toFixed (n) – τελική παρουσίαση (**Πηγή**: https://javascript.info/number)*

Ανακριβείς υπολογισμοί

Σε ένα αριθμητικό σύστημα, ένας αριθμός αντιπροσωπεύεται υπό την μορφή 64-bit (**IEEE-754**). Έτσι λοιπόν ένας αριθμός αποθηκεύεται χρησιμοποιώντας 64 bits: 52 εκ των οποίων χρησιμοποιούνται για την αποθήκευση των ψηφίων του ακέραιου μέρους ενός αριθμού, 11 για την αποθήκευση του δεκαδικού μέρους ενός αριθμού (10 μηδενικά για τις ακέραιες τιμές στο δεκαδικό μέρος, π.χ. 3.10000000000), και 1 bit είναι για τον δεκαδικό οριοθέτη (την τελεία).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αν ένας αριθμός είναι πολύ μεγάλος, τότε δεν θα χωρέσει στην αποθήκευση των 64-bit, δίνοντας ενδεχομένως μια άπειρη τιμή.

Αυτή η απώλεια ακρίβειας είναι αρκετά συνήθης.

Λαμβάνοντας υπόψη τα εξής:

```
1  alert( 0.1 + 0.2 == 0.3 ); // false
```

Εικόνα 62 *– Πρώτη δοκιμή για ανακριβείς υπολογισμούς (**Πηγή**: https://javascript.info/number)*

Όπως φαίνεται παραδόξως στο πιο κάτω παράδειγμα, το αποτέλεσμα από την πρόσθεση 0,1 +0,2= 0,3 επιστρέφεται εσφαλμένα ως "Ψευδές". Τι θα μπορούσε να είναι, τότε;

```
1  alert( 0.1 + 0.2 ); // 0.30000000000000004
```

Εικόνα 63 *– Δεύτερη δοκιμή για ανακριβείς υπολογισμούς (**Πηγή**: https://javascript.info/number)*

Όπως θα μπορούσε να φανταστεί κανείς, οι συνέπειες από μια εσφαλμένη σύγκριση θα ήταν πολύ περισσότερες στην περίπτωση, για παράδειγμα, που ένας αγοραστής κατά τη διάρκεια μιας ηλεκτρονικής αγοράς στο διαδίκτυο προσθέσει αγαθά αξίας $ 0.10 και $ 0.20 στο καλάθι του/της και το συνολικό ποσό της παραγγελίας ανερχόταν στο $ 0.30000000000000004. Αυτό θα ήταν προς εκπλήξεως μας πολύ απρόσμενο.

Γιατί όμως παρουσιάζονται τέτοιες καταστάσεις;

Ένας αριθμός αποθηκεύεται στη μνήμη, στη δυαδική μορφή του, δηλαδή σε μια ακολουθία bits αποτελούμενη από τιμές του 1 και του 0. Ωστόσο, κλάσματα όπως το 0,1, 0,2 τα οποία φαίνονται απλά στο δεκαδικό αριθμητικό σύστημα, είναι στην πραγματικότητα ατελείωτα κλάσματα στη δυαδική τους μορφή.

Δηλαδή, τι είναι το 0,1; Είναι ο αριθμός 1 διαιρούμενος με το δέκα, δηλαδή η κλασματική μορφή 1/10, ένα δέκατο. Στο δεκαδικό αριθμητικό σύστημα οι αριθμοί αυτοί μπορούν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

εύκολα να μεταφραστούν και να αναπαρασταθούν. Ωστόσο, στην περίπτωση της διαίρεσης το με το ένα τρίτο: 1/3, το αποτέλεσμα που έχουμε είναι ένας ατελείωτος δεκαδικός αριθμός 0,3333(3).

Έτσι λοιπόν, η διαίρεση με τις δυνάμεις του 10 είναι εφαρμόσιμη στο δεκαδικό σύστημα, αλλά η διαίρεση με το 3 όχι. Για τον ίδιο λόγο λοιπόν, στο δυαδικό αριθμητικό σύστημα, η διαίρεση με τις δυνάμεις του 2 είναι επίσης εφαρμόσιμη, αλλά η μετατροπή ενός δεκαδικού κλάσματος σε δυαδικό είναι μια επ' αόριστο διαδικασία.

Απλούστατα, δεν υπάρχει τρόπος να αποθηκεύσετε ακριβώς τους δεκαδικούς αριθμούς 0,1 ή 0,2 χρησιμοποιώντας το δυαδικό σύστημα, ακριβώς όπως δεν υπάρχει τρόπος να αποθηκεύσετε το ένα τρίτο ως δεκαδικό κλάσμα.

Η αριθμητική μορφή IEEE-754 το επιλύει αυτό στρογγυλοποιώντας το στον πλησιέστερο δυνατό αριθμό. Οι κανόνες στρογγυλοποίησης δεν μας επιτρέπουν να δούμε κανονικά εκείνη την "μικρή απώλεια ακρίβειας", έστω και αν αυτή υφίσταται στην πραγματικότητα.

```
1  alert( 0.1.toFixed(20) ); // 0.10000000000000000555
```

Εικόνα 64 *– Τελική δοκιμή για ανακριβείς υπολογισμούς (**Πηγή**: https://javascript.info/number)*

Όταν αθροίζονται δύο αριθμοί, οι ‘’ απώλειες ακρίβειας ’’ τους ενώνονται. Ως εκ τούτου, 0,1 + 0,2 δεν είναι 0,3 ακριβώς.

Η πιο αξιόπιστη μέθοδος για την αντιμετώπιση αυτής της κατάστασης είναι η χρήση της συνάρτησης **toFixed(n)**:

```
1  let sum = 0.1 + 0.2;
2  alert( sum.toFixed(2) ); // 0.30
```

Εικόνα 65 *– Η μέθοδος toFixed (n) (**Πηγή**: https://javascript.info/number)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Θα πρέπει να αναφερθεί ότι το toFixed (n) επιστρέφει μια συμβολοσειρά κάθε φορά. Εξασφαλίζει ότι ένας αριθμός έχει μόνο 2 ψηφία στο δεκαδικό του μέρος. Αυτή η μέθοδος είναι, ωστόσο, αρκετά χρήσιμη στην περίπτωση μιας ηλεκρονικής αγοράς στο διαδίκτυο όπου θα πρέπει να εμφανιστεί στο καλάθι η ακριβής τιμή αγοράς,

Δοκιμές: isFinite και isNaN

Όπως ειπώθηκε στην προηγούμενη περίπτωση με τα $ 0,30, μια τιμή **Infinity** (και **-infinity**) είναι μια ειδική αριθμητική τιμή που είναι μεγαλύτερη (μικρότερη) από οποιαδήποτε άλλη. Το **NaN, συντομογραφία του not a number,** αντιπροσωπεύει ένα σφάλμα, καθώς είναι μια τιμή που μπορεί να πάρει κάποια μεταβλητή που λαμβάνει είτε μια αδιευκρίνιστη τιμή, είτε μια τιμή που δεν μπορεί να αναπαρασταθεί. Και οι δύο προαναφερθέντες τιμές (του απείρου και του NaN) ενώ ανήκουν σε **έναν** τύπο αριθμών, δεν είναι εντούτοις «κανονικοί» αριθμοί, και γι' αυτό υπάρχουν συγκεκριμένες λειτουργίες για τον έλεγχό τους:

- το **isNaN (τιμή)** μετατρέπει το όρισμά του σε έναν αριθμό και στη συνέχεια τον δοκιμάζει ως **NaN**:

```
1  alert( isNaN(NaN) ); // true
2  alert( isNaN("str") ); // true
```

***Εικόνα 66** – Η μέθοδος isNaN (τιμή) (**Πηγή**: https://javascript.info/number)*

- το **isFinite (τιμή)** μετατρέπει το όρισμά του σε έναν αριθμό και το επιστρέφει ως ένα αποτέλεσμα που είναι "**Αληθές**" αν είναι κανονικός αριθμός, και όχι αν είναι **NaN/Infinity/-Infinity;**

```
1  alert( isFinite("15") ); // true
2  alert( isFinite("str") ); // false, because a special value: NaN
3  alert( isFinite(Infinity) ); // false, because a special value: Infinity
```

***Εικόνα 67** – Η μέθοδος isFinite (τιμή) (**Πηγή**: https://javascript.info/number)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Το parseInt και parseFloat

Η αριθμητική μετατροπή με τη χρήση του συν **+** ή **Αριθμού()** είναι εξονυχιστική. Εάν μια τιμή δεν αναπαριστάνει ακριβώς έναν αριθμό, τότε η αριθμητική μετατροπή είναι αποτυχής. Η μόνη εξαίρεση είναι τα κενά στην αρχή ή στο τέλος μιας συμβολοσειράς, καθώς αυτά αγνοούνται.

Ωστόσο, στην πραγματική ζωή υπάρχουν συχνά τιμές σε μονάδες, όπως για παράδειγμα "100px" ή "12pt" στο CSS. Παρομοίως, σε πολλές χώρες τα σύμβολα νομισμάτων τοποθετούνται μετά το αριθμητικό ποσό, όπως για παράδειγμα «19€», από το οποίο ένας προγραμματιστής θα ήθελε να εξάγει μια αριθμητική τιμή.

Γι 'αυτό λοιπόν υπάρχουν οι μέθοδοι **parseInt** και **parseFloat**.

«*Διαβάζουν*» έναν αριθμό από μια συμβολοσειρά μέχρι που δεν μπορούν να τον "διαβάσουν". Σε περίπτωση σφάλματος, επιστρέφεται ο αριθμός που συλλέχθηκε. Η μέθοδος parseInt **επιστρέφει** έναν ακέραιο αριθμό, ενώ η **συνάρτηση** parseFloat επιστρέφει έναν αριθμό κινητής υποδιαστολής:

```
1  alert( parseInt('100px') ); // 100
2  alert( parseFloat('12.5em') ); // 12.5
3
4  alert( parseInt('12.3') ); // 12, only the integer part is returned
5  alert( parseFloat('12.3.4') ); // 12.3, the second point stops the reading
```

Εικόνα 68 *– Οι μέθοδοι parseInt και parseFloat (**Πηγή**: https://javascript.info/number)*

Άλλες μαθηματικές λειτουργίες

Η JavaScript κατέχει ένα ενσωματωμένο αντικείμενο μαθηματικών που περιλαμβάνει μια μικρή βιβλιοθήκη μαθηματικών συναρτήσεων και σταθερών:

- **Math.random()**

  Επιστρέφει έναν τυχαίο αριθμό από το 0 μέχρι 1 (μη συμπεριλαμβανομένου του 1).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- **Math.max(a, b, c...) / Math.min(a, b, c...)**

Επιστρέφει το μεγαλύτερο/μικρότερο από τον αυθαίρετο αριθμό ορισμάτων.

- **Math.pow(n, ανύψωση ενός αριθμού στη δύναμη ενός εκθέτη)**

Επιστρέφει το αποτέλεσμα, δηλαδή το γινόμενο στην προκειμένη περίπτωση, της ύψωσης ενός αριθμού στη νιοστή δύναμη, δηλαδή στη δύναμη του n εκθέτη του.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

JavaScript Δηλώσεις If...Else

Όπως πολλές άλλες γλώσσες προγραμματισμού, η JavaScript επιτρέπει επίσης την εγγραφή κωδίκων που εκτελούν διαφορετικές ενέργειες με βάση τα αποτελέσματα τύπων υπολογισμών, όπως τους λογικούς τελεστές ή τους συγκριτικούς τελεστές και σε χρόνο εκτέλεσης. Έτσι, οι συνθήκες δοκιμής μπορούν να δημιουργηθούν ως εκφράσεις που αξιολογούν είτε το "Αληθές" είτε το "Ψευδές" και, με βάση αυτά τα αποτελέσματα, μπορούν να εκτελεστούν ορισμένες ενέργειες.

Υπάρχουν αρκετές υπό όρους δηλώσεις στο JavaScript που μπορούν να χρησιμοποιηθούν για τη λήψη αποφάσεων:

- Η δήλωση **if** ·
- Η δήλωση **if...else**;
- Η δήλωση **if...else if....else**;
- Η δήλωση **switch...case**

Η δήλωση if

Η δήλωση if χρησιμοποιείται για την εκτέλεση ενός μπλοκ κώδικα μόνο αν η καθορισμένη συνθήκη αξιολογείται ως αληθής. Αυτές είναι οι απλούστερες υπό όρους δηλώσεις της JS και μπορούν να γραφτούν ως:

```
if(condition) {
    // Code to be executed
}
```

Εικόνα 69 *– Κωδικός για τη δημιουργία μιας συνθήκης if (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-if-else-statements.php)*

Ένα πρακτικό παράδειγμα θα μπορούσε να αποδείξει αποτελεσματικότερα τη χρησιμότητα αυτού του χαρακτηριστικού. Εάν π.χ. σήμερα είναι Παρασκευή, η κωδικοποίηση που φαίνεται στην *Εικόνα 68* θα εμφανίσει ένα μήνυμα που θα λέει "Καλό Σαββατοκύριακο":

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript If Statement</title>
</head>
<body>
    <script>
    var now = new Date();
    var dayOfWeek = now.getDay(); // Sunday - Saturday : 0 - 6

    if(dayOfWeek == 5) {
        document.write("Have a nice weekend!");
    }
    </script>
    <p><strong>Note:</strong> This example will print "Have a nice weekend!" if the current day is Friday.</p>
</body>
</html>
```

Note: This example will print "Have a nice weekend!" if the current day is Friday.

Εικόνα 70 *– Η δήλωση if...else για την εμφάνιση ενός μηνύματος "Καλό Σαββατοκύριακο" σε περίπτωση που είναι Παρασκευή (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-if-else-statements.php)*

Η δήλωση if…else statement

Η διαδικασία λήψης αποφάσεων της JS μπορεί να ενισχυθεί με την παροχή μιας εναλλακτικής επιλογής μέσω της προσθήκης μιας **άλλης δήλωσης** στη **δήλωση if**. Η δήλωση if…else επιτρέπει την εκτέλεση ενός κώδικα μπλοκ εάν η καθορισμένη συνθήκη αξιολογείται ως *αληθής* και εάν ένας άλλος κώδικας μπλοκ αξιολογείται ως *ψευδής*.

```
if(condition) {
    // Code to be executed if condition is true
} else {
    // Code to be executed if condition is false
}
```

Εικόνα 71 *– Κώδικας για τη δημιουργία μιας συνθήκης if (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-if-else-statements.php)*

Το παράδειγμα από το *Σχήμα 68* θα εφαρμοστεί για τη δοκιμή της δήλωσης if…else. Αυτή τη φορά, το μήνυμα "Καλό Σαββατοκύριακο!" θα εμφανιστεί σε περίπτωση που η τρέχουσα ημέρα είναι Παρασκευή (όπως και στην προηγούμενη περίπτωση), ωστόσο, θα εκπέμψει το μήνυμα "Καλή σας μέρα" εάν δεν είναι Παρασκευή.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript If-Else Statement</title>
</head>
<body>
    <script>
    var now = new Date();
    var dayOfWeek = now.getDay(); // Sunday - Saturday : 0 - 6

    if(dayOfWeek == 5) {
        document.write("Have a nice weekend!");
    } else {
        document.write("Have a nice day!");
    }
    </script>
</body>
</html>
```

Have a nice day!

Εικόνα 72 *– Κώδικας για τη δημιουργία μιας συνθήκης if…else, με την εμφάνιση ενός μηνύματος "Καλή σας μέρα" ή "Καλό σας Σαββατοκύριακο" ανάλογα με την περίπτωση (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-if-else-statements.php)*

Ο Τριμερής Φορέας Εκμετάλλευσης (The Ternary Operator)

Ο τριμερής φορέας εκμετάλλευσης μας επιτρέπει **να γράφουμε τις συντομογραφίες των δηλώσεων if...else**, χρησιμοποιώντας τον χαρακτήρα του ερωτηματικού ('?') και χρειάζονται τρεις τελεστέοι: ένας που ελέγχει και συγκρίνει τις τιμές, ένας που επιστρέφει ένα αποτέλεσμα που είναι "Αληθές" και ένας που επιστρέφει το αποτέλεσμα που είναι "Ψευδές". Η βασική σύνταξη έχει ως εξής:

var result = (condition) ? value1 : value2

Αν η συνθήκη αξιολογηθεί ως *αληθής,* η τιμή 1 θα επιστραφεί, αν όχι τότε θα επιστραφεί η τιμή 2.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Typical Conditional Statement</title>
</head>
<body>
    <script>
    var userType;
    var age = 21;
    if(age < 18) {
        userType = 'Child';
    } else {
        userType = 'Adult';
    }
    document.write(userType); // Prints Adult
    </script>
</body>
</html>
```

Adult

***Εικόνα 73** – Τρόπος εφαρμογής του τριμερή φορέα εκμετάλλευσης (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-if-else-statements.php)*

Δηλώσεις Switch…Case της JavaScript

Μια δήλωση **switch..case** είναι ένα εναλλακτικό σενάριο της δήλωσης **if...else if...else**, το οποίο κάνει σχεδόν το ίδιο πράγμα. Η δήλωση **switch...case** αναλύει μια μεταβλητή ή μια έκφραση έναντι μιας σειράς τιμών μέχρι να βρει μια αντιστοιχία, και στη συνέχεια εκτελεί το μπλοκ του κώδικα που αντιστοιχεί σε αυτή την αντιστοιχία. Η βασική σύνταξη έχει ως εξής:

```
switch(x){
    case value1:
        // Code to be executed if x === value1
        break;
    case value2:
        // Code to be executed if x === value2
        break;
    ...
    default:
        // Code to be executed if x is different from all values
}
```

***Εικόνα 74 –** Σύνταξη για δηλώσεις switch…case (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-if-else-statements.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αυτό το παράδειγμα εμφανίζει το όνομα της ημέρας της εβδομάδας στην οποία βρίσκεται ο αναγνώστης.

Η δήλωση **switch...case** διαφέρει από την δήλωση **if...else** σε μια σημαντική πτυχή. Η δήλωση switch...case εκτελεί γραμμή προς γραμμή και όταν η JavaScript βρίσκει μια πρόταση (case clause) που αξιολογεί ως *αληθή*, δεν εκτελεί μόνο τον κώδικα που αντιστοιχεί για την πρόταση αυτή, αλλά εκτελεί αυτόματα επίσης όλες τις διαδοχικές προτάσεις μέχρι το τέλος του μπλοκ switch.

Για να αποφευχθεί αυτό, πρέπει να περιλαμβάνεται μια δήλωση switch μετά από κάθε περίπτωση (όπως μπορούμε να συμπεράνουμε στην *Εικόνα 73*). Η δήλωση **switch...case** ενημερώνει τον διερμηνέα της JS να σπάσει το μπλοκ switch όταν τελειώσει με την εκτέλεση του κώδικα στην περίπτωση που ένα αποτέλεσμα αξιολογείται ως *αληθές*.

Η δήλωση break δεν είναι απαραίτητη σε προεπιλεγμένες περιπτώσεις ή προτάσεις (default clause). Ωστόσο, η απόρριψη της τελευταίας περίπτωσης ή προεπιλεγμένης πρότασης σε μια δήλωση switch με διακοπή/λέξη-κλειδί break είναι μια καλή πρακτική στον προγραμματισμό. Αυτή η επιλογή επιτρέπει την αποφυγή ενός πιθανούς σφάλματος στην πορεία εάν μια άλλη δήλωση περίπτωσης συμπεριλαμβάνεται ήδη στην δήλωση switch.

Η προεπιλεγμένη πρόταση είναι προαιρετική, η οποία ορίζει τις ενέργειες που πρέπει να γίνουν εάν καμία περίπτωση δεν ταιριάζει με την έκφραση του διακόπτη. Η προεπιλεγμένη πρόταση δεν χρειάζεται να είναι η τελευταία πρόταση που βρίσκουμε σε μια δήλωση break.

Η τιμή κάθε περίπτωσης (case) θα πρέπει να είναι αποκλειστική μόνο μέσα σε μια δήλωση switch. Παρόλα αυτά, διάφοροι τύποι περιπτώσεων δεν χρειάζεται να έχουν μια μοναδική δράση. Αρκετές υποθέσεις μπορούν να έχουν την ίδια δράση.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Συστοιχίες JavaScript

Οι συστοιχίες είναι πολύπλοκες μεταβλητές που επιτρέπουν την αποθήκευση περισσότερων από μία τιμών ή μιας ομάδας τιμών κάτω από ένα όνομα μεταβλητής. Οι συστοιχίες JavaScript μπορούν να αποθηκεύσουν οποιαδήποτε έγκυρη τιμή, συμπεριλαμβανομένων συμβολοσειρών, αριθμών, αντικειμένων, λειτουργιών και ταυτόχρονα άλλων συστοιχιών, επιτρέποντας έτσι τη δημιουργία πιο σύνθετων δομών δεδομένων, όπως μια συστοιχία αντικειμένων ή μια συστοιχία συστοιχιών.

Στο παρακάτω παράδειγμα, το όνομα των χρωμάτων θα αποθηκευτεί σε έναν κώδικα JavaScript:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Storing Single Values</title>
</head>
<body>
    <script>
    // Creating variables
    var color1 = "Red";
    var color2 = "Green";
    var color3 = "Blue";

    // Printing variable values
    document.write(color1 + "<br>");
    document.write(color2 + "<br>");
    document.write(color3);
    </script>
</body>
</html>
```

```
Red
Green
Blue
```

Εικόνα 75 – *Αποθήκευση του ονόματος των χρωμάτων σε έναν κώδικα JS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Ωστόσο, μπορεί να είναι αρκετά δύσκολο και κουραστικό να αποθηκεύσετε πολλαπλά στοιχεία σε μεταβλητές, σε αντίθεση με μόνο τρεις στην παραπάνω περίπτωση. Επιπλέον, η χρήση τόσων πολλών μεταβλητών ταυτόχρονα και η παρακολούθηση όλων αυτών θα είναι ένα πολύ δύσκολο έργο. Και εδώ ο πίνακας μπαίνει στο παιχνίδι. Οι πίνακες επιλύουν αυτό το πρόβλημα παρέχοντας μια ταξινομημένη δομή για την αποθήκευση πολλαπλών τιμών ή μιας ομάδας τιμών.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Δημιουργία μιας συστοιχίας

Ο ευκολότερος τρόπος για να δημιουργήσετε μια συστοιχία στο JavaScript είναι να επισυνάψετε μια λίστα τιμών διαχωρισμένων με κόμματα σε αγκύλες ([]), όπως φαίνεται στην ακόλουθη σύνταξη:

```
var myArray = [element0, element1, ..., elementN];
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Creating Arrays in JavaScript</title>
</head>
<body>
    <script>
    // Creating variables
    var colors = ["Red", "Green", "Blue"];
    var fruits = ["Apple", "Banana", "Mango", "Orange", "Papaya"];
    var cities = ["London", "Paris", "New York"];
    var person = ["John", "Wick", 32];

    // Printing variable values
    document.write(colors + "<br>");
    document.write(fruits + "<br>");
    document.write(cities + "<br>");
    document.write(person);
    </script>
</body>
</html>
```

```
Red,Green,Blue
Apple,Banana,Mango,Orange,Papaya
London,Paris,New York
John,Wick,32
```

Εικόνα 76 – *Δημιουργία ενός πίνακα (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Πρόσβαση στα στοιχεία μιας συστοιχίας

Τα στοιχεία πίνακα μπορούν να προσπελαστούν από τον δείκτη τους χρησιμοποιώντας τη σημειογραφία τετραγωνικής παρένθεσης. Ένας δείκτης είναι ένας αριθμός που αντιπροσωπεύει τη θέση ενός στοιχείου σε μια συστοιχία.

Οι δείκτες των συστοιχιών βασίζονται στο μηδέν. Αυτό σημαίνει ότι το πρώτο στοιχείο ενός πίνακα αποθηκεύεται στο δείκτη 0, όχι 1, το δεύτερο στοιχείο αποθηκεύεται στο δείκτη 1, και ούτω καθεξής. Οι δείκτες της συστοιχίας αρχίζουν από το 0 και φτάνουν μέχρι τον αριθμό των στοιχείων μείον 1. Έτσι, η σειρά των πέντε στοιχείων θα έχει δείκτες από το 0 έως το 4.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Access Individual Elements of an Array</title>
</head>
<body>
    <script>
    var fruits = ["Apple", "Banana", "Mango", "Orange", "Papaya"];

    document.write(fruits[0] + "<br>"); // Prints: Apple
    document.write(fruits[1] + "<br>"); // Prints: Banana
    document.write(fruits[2] + "<br>"); // Prints: Mango
    document.write(fruits[fruits.length - 1]); // Prints: Papaya
    </script>
</body>
</html>
```

Apple
Banana
Mango
Papaya

Εικόνα 77 – *Τρόπος λήψης μεμονωμένου στοιχείου πίνακα από το ευρετήριό τους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Λήψη του μήκους μιας συστοιχίας

Η ιδιότητα μήκος επιστρέφει το μήκος ενός πίνακα, που είναι ο συνολικός αριθμός των στοιχείων που περιλαμβάνονται στον πίνακα. Το μήκος του πίνακα είναι πάντα μεγαλύτερο από το δείκτη οποιουδήποτε στοιχείου του.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Get the Length of an Array</title>
</head>
<body>
    <script>
    var fruits = ["Apple", "Banana", "Mango", "Orange", "Papaya"];
    document.write(fruits.length); // Outputs: 5
    </script>
</body>
</html>
```

5

Εικόνα 78 – *Λήψη του μήκους μιας διάταξης (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Βρόχοι στοιχείων μιας συστοιχίας.

Για να αποκτήσετε πρόσβαση σε όλα τα στοιχεία μιας συστοιχίας με διαδοχική σειρά, μπορείτε να χρησιμοποιήσετε τους βρόχους, ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Loop Through an Array Using For Loop</title>
</head>
<body>
    <script>
    var fruits = ["Apple", "Banana", "Mango", "Orange", "Papaya"];

    // Iterates over array elements
    for(var i = 0; i < fruits.length; i++){
        document.write(fruits[i] + "<br>"); // Print array element
    }
    </script>
</body>
</html>
```

Apple
Banana
Mango
Orange
Papaya

Εικόνα 79 – *Ανακύκλωση στοιχείων πίνακα (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Το ECMAScript 6 εισήγαγε έναν απλούστερο τρόπο επανάληψης πάνω από το στοιχείο συστοιχίας, το οποίο είναι for-of loop. Σε αυτόν τον βρόχο δεν υπάρχει ανάγκη αρχικοποίησης και παρακολούθησης της μεταβλητής του μετρητή βρόχων (i).

Προσθήκη νέων στοιχείων σε μια συστοιχία

Για να προσθέσετε ένα νέο στοιχείο στο τέλος ενός πίνακα, απλά χρησιμοποιήστε τη μέθοδο push(), ως εξής:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Add a New Element at the End of an Array</title>
</head>
<body>
    <script>
    var colors = ["Red", "Green", "Blue"];
    colors.push("Yellow");

    document.write(colors + "<br>"); // Prints: Red,Green,Blue,Yellow
    document.write(colors.length); // Prints: 4
    </script>
</body>
</html>
```

Red,Green,Blue,Yellow
4

Εικόνα 80 – *Προσθήκη νέων στοιχείων σε μια συστοιχία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Με παρόμοιο τρόπο, για να προστεθεί ένα νέο στοιχείο στην αρχή μιας συστοιχίας, θα πρέπει να χρησιμοποιηθεί η μέθοδος unshift (), ως εξής:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Add a New Element at the Beginning of an Array</title>
</head>
<body>
    <script>
    var colors = ["Red", "Green", "Blue"];
    colors.unshift("Yellow");

    document.write(colors + "<br>"); // Prints: Yellow,Red,Green,Blue
    document.write(colors.length); // Prints: 4
    </script>
</body>
</html>
```

```
Yellow,Red,Green,Blue
4
```

Εικόνα 81 – *Προσθέστε ένα νέο στοιχείο στην αρχή ενός πίνακα χρησιμοποιώντας τη μέθοδο unshift() (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Αφαίρεση στοιχείων σε μια συστοιχία

Για να αφαιρέσετε το τελευταίο στοιχείο από έναν πίνακα, μπορείτε να χρησιμοποιήσετε τη μέθοδο Pop(). Αυτή η μέθοδος επιστρέφει την τιμή που αναδύθηκε.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Remove the Last Element from an Array</title>
</head>
<body>
    <script>
    var colors = ["Red", "Green", "Blue"];
    var last = colors.pop();

    document.write(last + "<br>"); // Prints: Blue
    document.write(colors.length); // Prints: 2
    </script>
</body>
</html>
```

```
Blue
2
```

Εικόνα 82 – *Αφαίρεση στοιχείων από μια συστοιχία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Προσθήκη ή αφαίρεση στοιχείων σε οποιαδήποτε θέση

Η μέθοδος splice() είναι μια πολύ ευέλικτη μέθοδος συστοιχίας που επιτρέπει την προσθήκη ή αφαίρεση στοιχείων από οποιοδήποτε ευρετήριο, χρησιμοποιώντας τη διάταξη σύνταξης `arr.splice(startIndex, deleteCount, elem1, ..., elemN)`. Αυτή η μέθοδος παίρνει τρεις παραμέτρους: η πρώτη είναι ο δείκτης στον οποίο θα ξεκινήσει η σύζευξη της συστοιχίας, απαιτείται. Η δεύτερη είναι ο αριθμός των στοιχείων που πρέπει να αφαιρεθούν (το 0 πρέπει να χρησιμοποιηθεί σε περίπτωση που ο

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

προγραμματιστής δεν θέλει να αφαιρέσει κανένα στοιχείο), είναι προαιρετική. Και η τρίτη παράμετρος είναι ένα σύνολο στοιχείων αντικατάστασης, είναι επίσης προαιρετική.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Add or Remove Array Elements at any Index</title>
</head>
<body>
    <script>
    var colors = ["Red", "Green", "Blue"];
    var removed = colors.splice(0,1); // Remove the first element

    document.write(colors + "<br>"); // Prints: Green,Blue
    document.write(removed + "<br>"); // Prints: Red (one item array)
    document.write(removed.length + "<br>"); // Prints: 1

    removed = colors.splice(1, 0, "Pink", "Yellow"); // Insert two items at position one
    document.write(colors + "<br>"); // Prints: Green,Pink,Yellow,Blue
    document.write(removed + "<br>"); // Empty array
    document.write(removed.length + "<br>"); // Prints: 0

    removed = colors.splice(1, 1, "Purple", "Voilet"); // Insert two values, remove one
    document.write(colors + "<br>"); //Prints: Green,Purple,Voilet,Yellow,Blue
    document.write(removed + "<br>"); // Prints: Pink (one item array)
    document.write(removed.length); // Prints: 1
    </script>
</body>
</html>
```

```
Green,Blue
Red
1
Green,Pink,Yellow,Blue

0
Green,Purple,Voilet,Yellow,Blue
Pink
1
```

Εικόνα 83 – *Προσθήκη ή αφαίρεση στοιχείων σε οποιαδήποτε θέση (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Η μέθοδος splice() επιστρέφει μια συστοιχία των διαγραμμένων στοιχείων, ή μια κενή συστοιχία αν δεν διαγράφηκαν στοιχεία, όπως φαίνεται στην *Εικόνα 81*. Εάν το δεύτερο όρισμα απουσιάζει, όλα τα στοιχεία από την αρχή έως το τέλος του πίνακα αφαιρούνται. Σε αντίθεση με τις μεθόδους slice() και concat(), η μέθοδος splice() τροποποιεί τη συστοιχία στην οποία γίνεται κλήση.

Δημιουργία συμβολοσειράς από μια συστοιχία

Μπορεί να υπάρχουν καταστάσεις στις οποίες ένας προγραμματιστής απλά σκοπεύει να δημιουργήσει μια συμβολοσειρά με τη σύνδεση των στοιχείων ενός πίνακα. Για να γίνει αυτό, μπορεί να χρησιμοποιήσει τη μέθοδο join(). Αυτή η μέθοδος παίρνει μια προαιρετική παράμετρο η οποία είναι μια διαχωριστική συμβολοσειρά που προστίθεται ανάμεσα σε κάθε στοιχείο. Αν παραλείψετε το διαχωριστικό, τότε η JavaScript θα χρησιμοποιήσει κόμμα (,) από προεπιλογή.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Join All Elements of an Array into a String</title>
</head>
<body>
    <script>
    var colors = ["Red", "Green", "Blue"];

    document.write(colors.join() + "<br>"); // Prints: Red,Green,Blue
    document.write(colors.join("") + "<br>"); // Prints: RedGreenBlue
    document.write(colors.join("-") + "<br>"); // Prints: Red-Green-Blue
    document.write(colors.join(", ")); // Prints: Red, Green, Blue
    </script>
</body>
</html>
```

```
Red,Green,Blue
RedGreenBlue
Red-Green-Blue
Red, Green, Blue
```

Εικόνα 84 – *Δημιουργία συμβολοσειράς από μια συστοιχία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Μια συστοιχία μπορεί επίσης να μετατραπεί σε μια συμβολοσειρά διαχωρισμένη με κόμμα χρησιμοποιώντας το toString(). Αυτή η μέθοδος δεν επιτρέπει την παράμετρο διαχωρισμού ως join().

Συγχώνευση δύο ή περισσότερων πινάκων

Η μέθοδος concat() μπορεί να χρησιμοποιηθεί για τον συνδυασμό δύο ή περισσότερων συστοιχιών. Αυτή η μέθοδος δεν αλλάζει τις επικρατούσες συστοιχίες, αντ 'αυτού επιστρέφει μια νέα συστοιχία.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Merge Two Arrays</title>
</head>
<body>
    <script>
    var pets = ["Cat", "Dog", "Parrot"];
    var wilds = ["Tiger", "Wolf", "Zebra"];

    // Creating new array by combining pets and wilds arrays
    var animals = pets.concat(wilds);
    document.write(animals); // Prints: Cat,Dog,Parrot,Tiger,Wolf,Zebra
    </script>
</body>
</html>
```

```
Cat,Dog,Parrot,Tiger,Wolf,Zebr
```

Εικόνα 85 – *Συγχώνευση δύο ή περισσότερων παρατάξεων (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Η μέθοδος concat() μπορεί να λάβει οποιονδήποτε αριθμό παραμέτρων πίνακα, έτσι ένας πίνακας μπορεί να δημιουργηθεί από οποιονδήποτε αριθμό άλλων πινάκων, όπως φαίνεται στο ακόλουθο παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Merge Multiple Arrays</title>
</head>
<body>
    <script>
    var pets = ["Cat", "Dog", "Parrot"];
    var wilds = ["Tiger", "Wolf", "Zebra"];
    var bugs = ["Ant", "Bee"];

    // Creating new array by combining pets, wilds and bugs arrays
    var animals = pets.concat(wilds, bugs);
    document.write(animals); // Prints: Cat,Dog,Parrot,Tiger,Wolf,Zebra,Ant,Bee
    </script>
</body>
</html>
```

Cat,Dog,Parrot,Tiger,Wolf,Zebra,Ant,Bee

Εικόνα 86 – *Η μέθοδοςconcat () (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Αναζήτηση μέσα σε μια συστοιχία

Οι μέθοδοι indexOf() και lastIndexOf() μπορούν να χρησιμοποιηθούν για την αναζήτηση μιας συστοιχίας για μια συγκεκριμένη τιμή. Εάν βρεθεί η τιμή, και οι δύο μέθοδοι επιστρέφουν ένα δείκτη που αντιπροσωπεύει το στοιχείο πίνακα. Αν δεν βρεθεί η τιμή, επιστρέφεται -1. Η μέθοδος indexOf() επιστρέφει την πρώτη που βρέθηκε, αν και η lastIndexOf() επιστρέφει την τελευταία που βρέθηκε. Και οι δύο μέθοδοι αναγνωρίζουν επίσης μια προαιρετική ακέραια παράμετρο από το ευρετήριο, η οποία καθορίζει το δείκτη εντός του πίνακα στον οποίο θα ξεκινήσει η αναζήτηση.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Search an Array for a Specific Value</title>
</head>
<body>
    <script>
    var fruits = ["Apple", "Banana", "Mango", "Orange", "Papaya"];

    document.write(fruits.indexOf("Apple") + "<br>"); // Prints: 0
    document.write(fruits.indexOf("Banana") + "<br>"); // Prints: 1
    document.write(fruits.indexOf("Pineapple")); // Prints: -1
    </script>
</body>
</html>
```

0
1
-1

Εικόνα 87 – *Αναζήτηση σε πίνακα (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Η μέθοδος include() μπορεί επίσης να χρησιμοποιηθεί για να μάθετε αν ένας πίνακας περιλαμβάνει ένα συγκεκριμένο στοιχείο ή όχι. Αυτή η μέθοδος παίρνει τις ίδιες παραμέτρους με τις μεθόδους indexOf() και lastIndexOf(), αλλά επιστρέφει τις τιμές αληθές ή ψευδές αντί για αριθμητικούς δείκτες.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Find Whether an Array Includes a Certain Value</title>
</head>
<body>
    <script>
    var arr = [1, 0, 3, 1, false, 5, 1, 4, 7];

    document.write(arr.includes(1) + "<br>"); // Prints: true
    document.write(arr.includes(6) + "<br>"); // Prints: false
    document.write(arr.includes(1, 2) + "<br>"); // Prints: true
    document.write(arr.includes(3, 4)); // Prints: false
    </script>
</body>
</html>
```

```
true
false
true
false
```

Εικόνα 88 – *Αναζήτηση σε μια συστοιχία χρησιμοποιώντας τη μέθοδο includes () (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Για να αναζητήσετε μια συστοιχία με βάση μια συγκεκριμένη κατάσταση μπορεί να χρησιμοποιηθεί η μέθοδος JavaScript find(), η οποία έχει εισαχθεί πρόσφατα στο ES6. Αυτή η μέθοδος επιστρέφει την τιμή του πρώτου στοιχείου στη συστοιχία που πληροί την παρεχόμενη λειτουργία δοκιμής. Αν όχι, επιστρέφει την τιμή undefined.

Η μέθοδος find() αναζητά απλά το πρώτο στοιχείο που πληροί την παρεχόμενη λειτουργία δοκιμής. Ακόμα, εάν ο προγραμματιστής σκοπεύει να ανακαλύψει όλα τα αντιστοιχισμένα στοιχεία, μπορεί να χρησιμοποιήσει τη μέθοδο filter(). Η μέθοδος αυτή δημιουργεί μια νέα συστοιχία με όλα τα στοιχεία που περνούν επιτυχώς τη δοσμένη δοκιμή, όπως μπορεί να εξετασθεί σε αυτό το παράδειγμα.

Ταξινόμηση συστοιχιών αντικειμένων JavaScript

Η ταξινόμηση είναι μια δημοφιλής εργασία όταν εργάζεστε με συστοιχίες. Μπορεί να χρησιμοποιηθεί, για παράδειγμα, για την εμφάνιση των ονομάτων των πόλεων ή των χωρών με αλφαβητική σειρά. Το αντικείμενο JavaScript Array έχει μια ενσωματωμένη μέθοδο sorting() για την ταξινόμηση στοιχείων μιας συστοιχίας με αλφαβητική σειρά.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JavaScript Sort an Array Alphabetically</title>
</head>
<body>
    <script>
    var fruits = ["Banana", "Orange", "Apple", "Papaya", "Mango"];
    var sorted = fruits.sort();

    document.write(fruits + "<br>"); // Outputs: Apple,Banana,Mango,Orange,Papaya
    document.write(sorted); // Outputs: Apple,Banana,Mango,Orange,Papaya
    </script>
</body>
</html>
```

```
Apple,Banana,Mango,Orange,Papaya
Apple,Banana,Mango,Orange,Papaya
```

Εικόνα 89 – *Η ενσωματωμένη μέθοδος ταξινόμησης() για τη ταξινόμηση στοιχείων πίνακα με αλφαβητική σειρά (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-arrays.php)*

Για να **αντιστραφεί η σειρά των στοιχείων** ενός πίνακα, μπορεί να χρησιμοποιηθεί η reverse() μέθοδος. Αντιστρέφει μια συστοιχία με τέτοιο τρόπο ώστε το πρώτο στοιχείο συστοιχίας να γίνεται το τελευταίο, και αντίστροφα. Η μέθοδος sort() και reverse() αλλάζει την αρχική συστοιχία και επιστρέφει μια αναφορά (reference) στην ίδια συστοιχία, όπως μπορεί να εξετασθεί στο παράδειγμα εδώ.

Για **τη ταξινόμηση αριθμητικών συστοιχιών**, δεν συνιστάται η χρήση της μεθόδου sort(), καθώς μπορεί να παράγει απροσδόκητα αποτελέσματα. Για το σκοπό αυτό, οι προγραμματιστές θα πρέπει να περνούν μια συνάρτηση σύγκρισης (compare function), καθώς όταν καθορίζεται μια συνάρτηση σύγκρισης, τα στοιχεία πίνακα ταξινομούνται σύμφωνα με την τιμή επιστροφής της συνάρτησης σύγκρισης.
Για παράδειγμα, κατά τη σύγκριση των a και b:

- Αν η συνάρτηση Compare επιστρέφει μια τιμή μικρότερη από 0, τότε η *a* έρχεται πρώτη.
- Εάν η συνάρτηση Compare επιστρέφει μια τιμή μεγαλύτερη από 0, τότε το *b* έρχεται πρώτο.
- Αν η συνάρτηση Compare επιστρέφει 0, το *a* και το *b* παραμένουν αμετάβλητα μεταξύ τους, αλλά ομαδοποιημένα σε σχέση με όλα τα άλλα στοιχεία.

Ως εκ τούτου, δεδομένου ότι 5 - 20 = - 15 που είναι μικρότερο από 0, επομένως το 5 έρχεται πρώτο, ομοίως 20 - 10 = 10 που είναι μεγαλύτερο από 0, επομένως το 10 έρχεται πριν από 20, ομοίως 20 - 75 = -55 που είναι μικρότερο από 0, έτσι το 20 έρχεται

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

πριν από 75, ομοίως 50 έρχεται πριν από 75, και ούτω καθεξής. Αυτό μπορεί να ελεγχθεί σε αυτό το παράδειγμα.

Για **την εύρεση της μέγιστης και ελάχιστης τιμής σε μι συστοιχία**, μπορεί να χρησιμοποιηθεί η μέθοδος apply() σε συνδυασμό με τα Math.max() και Math.min (), όπως παρουσιάζεται στο παράδειγμα αυτό. Η μέθοδος apply() προσφέρει έναν προσβάσιμο τρόπο για να περάσετε τις τιμές πίνακα ως ορίσματα σε μια συνάρτηση που δέχεται πολλαπλά ορίσματα με έναν τρόπο που μοιάζει με πίνακα, αλλά όχι έναν πίνακα. Στη συνέχεια, η δήλωση που προκύπτει Math.max.apply(null, numbers) στο παραπάνω παράδειγμα είναι ισοδύναμη με το Math.max(3, -7, 10, 8, 15, 2).

Τέλος, για την **ταξινόμηση μιας σειράς αντικειμένων** μπορεί να χρησιμοποιηθεί η μέθοδος sort(). Σε αυτό το παράδειγμα, μπορείτε να δείτε πώς να ταξινομείτε μια σειρά αντικειμένων με τιμές ιδιοτήτων.

Βρόχοι JavaScript

Οι βρόχοι εφαρμόζονται για την εκτέλεση του ίδιου μπλοκ κώδικα ξανά και ξανά εάν συναντάται μια συγκεκριμένη προϋπόθεση. Η βασική ιδέα πίσω από ένα βρόχο είναι να μηχανοποιήσει τις επαναλαμβανόμενες εργασίες μέσα σε ένα πρόγραμμα για να εξοικονομήσει χρόνο και προσπάθεια. Η JavaScript υποστηρίζει πλέον πέντε διαφορετικούς τύπους βρόχων:

- **while** — βρόχος μέσω ενός μπλοκ κώδικα, αν μια συγκεκριμένη προϋπόθεση *είναι αληθής*
- **do...while** — βρόχος μέσω ενός μπλοκ κώδικα μια φορά; τότε η κατάσταση αξιολογείται. Εάν η προϋπόθεση είναι *αληθής* τότε η δήλωση επαναλαμβάνεται.
- **for** — βρόχος μέσω ενός μπλοκ κώδικα για έναν αριθμό επαναλήψεων.
- **for…in** — βρόχος μέσα από τις ιδιότητες ενός αντικειμένου.
- **for… — of** βρόχος μέσω των τιμών ενός επαναλαμβανόμενου αντικειμένου όπως πίνακες, συστοιχίες κ.λπ.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο βρόχος while

Αυτή είναι η ευκολότερη δήλωση βρόχου που παρέχεται από την JS, δηλαδή ένας βρόχος μέσω ενός μπλοκ κώδικα αν μια καθορισμένη προϋπόθεση *είναι αληθής*. Εάν μια προϋπόθεση είναι ψευδής, τότε ο βρόχος διακόπτεται. Η γενική σύνταξη του βρόχου while είναι:

```
while(condition) {
   // ο κώδικας προς εκτέλεση
}
```

Στο παράδειγμα αυτό, ένας βρόχος συνεχίζει να λειτουργεί για όσο διάστημα διαρκεί η μεταβλητή *i ≤ 5.* Το i θα προσαυξάνεται κατά 1 κάθε φορά που τρέχει ο βρόχος. Οι προγραμματιστές πρέπει να διασφαλίζουν ότι μια προϋπόθεση που καθορίζεται στο βρόχο μπορεί τελικά να επιστραφεί ως ψευδής. Αν όχι, ο βρόχος δεν θα σταματήσει ποτέ να επαναλαμβάνεται (άπειρος βρόχος/infinite loop).

Ο βρόχος do…while

Ο βρόχος do-while είναι μια παραλλαγή του βρόχου while, η οποία αξιολογεί την κατάσταση στο τέλος κάθε επανάληψης βρόχου. Με ένα βρόχο do…while, το μπλοκ του κώδικα εκτελείται μία φορά, και στη συνέχεια η κατάσταση αξιολογείται. Εάν η προϋπόθεση είναι *αληθής*, η δήλωση τότε επαναλαμβάνεται εάν πληρούται η καθορισμένη προϋπόθεση. Η κοινή του σύνταξη είναι:

```
do {
   // Code to be executed
}
while(condition);
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο κώδικας JavaScript στο παράδειγμα αυτό ορίζει έναν βρόχο που ξεκινά με *i=1*. Στη συνέχεια θα παράγει την έξοδο και θα αυξήσει την τιμή της μεταβλητής i κατά 1. Στη συνέχεια αξιολογείται η κατάσταση και ο βρόχος θα συνεχίσει να λειτουργεί εάν το *i ≤ 5*.

Ο βρόχος for

Επαναλαμβάνει ένα μπλοκ κώδικα εάν πληρούται μια συγκεκριμένη προϋπόθεση. Χρησιμοποιείται συνήθως για την εφαρμογή ενός μπλοκ κώδικα για ορισμένες φορές. Η σύνταξή του είναι:

```
for(initialization; condition; increment) {
   // Code to be executed
}
```

Οι παράμετροι της δήλωσης **βρόχου** έχουν τις ακόλουθες επιπτώσεις:

- **εκκίνηση** (initialisation) — χρησιμοποιείται για να ρυθμίσετε τις μεταβλητές μετρητή, και αξιολογείται μία φορά εξ ολοκλήρου πριν από την πρώτη εκτέλεση του σώματος του βρόχου.
- **προϋπόθεση** — αξιολογείται στην αρχή κάθε επανάληψης. Εάν αξιολογηθεί ως *αληθής*, οι δηλώσεις βρόχου επιτυγχάνουν. Αν αξιολογηθεί ως *ψευδής*, τότε η εκτέλεση του βρόχου θα τελειώσει.
- **προσαύξηση** — ενημερώνει τον μετρητή βρόχων με μια νέα τιμή κάθε φορά που τρέχει ο βρόχος.

Αυτό το παράδειγμα ορίζει έναν βρόχο που ξεκινά με i=1. Ο βρόχος θα συνεχιστεί μέχρι την τιμή *i ≤ 5*. Η μεταβλητή i θα προσαυξάνεται κατά 1 κάθε φορά που εκτελείται ο βρόχος.

Ο βρόχος for...in

Είναι ένας ειδικός τύπος βρόχου μέσα από τις ιδιότητες ενός αντικειμένου, ή τα στοιχεία μιας συστοιχίας. Η σύνταξή του είναι:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
for (η μεταβλητή στο αντικείμενο) {
   // Ο κωδικός προς εκτέλεση
}
```

Ο μετρητής βρόχων, δηλαδή η μεταβλητή στον βρόχο for-in, είναι μια συμβολοσειρά και όχι ένας αριθμός. Περιλαμβάνει το όνομα της παρούσας ιδιότητας ή το δείκτη του τρέχοντος στοιχείου συστοιχίας.

Αυτό το παράδειγμα δείχνει πώς να περιηγηθείτε σε όλες τις ιδιότητες ενός αντικειμένου JS.

Ο βρόχος for...of

Το ES6 παρουσιάζει μια νέα δήλωση for-of που επιτρέπει τη δημιουργία βρόχων πάνω σε δομές δεδομένων που είναι επαναληπτικές όπως οι συστοιχίες ή άλλα επαναλαμβανόμενα αντικείμενα (π.χ. συμβολοσειρές) με πολύ απλό τρόπο. Επιπλέον, ο κώδικας μέσα στον βρόχο εκτελείται για κάθε στοιχείο του επαναλαμβανόμενου αντικειμένου. Αυτό το παράδειγμα δείχνει πώς να δημιουργήσετε βρόχους πάνω σε δομές δεδομένων όπως οι συστοιχίες και οι συμβολοσειρές.

Συναρτήσεις JavaScript

Μια συνάρτηση είναι ένα σύνολο δηλώσεων που εκτελούν συγκεκριμένα ενέργειες και μπορούν να διατηρηθούν ανεξάρτητα. Οι συναρτήσεις παρέχουν τη δυνατότητα δημιουργίας επαναχρησιμοποιήσιμων πακέτων κώδικα που είναι πιο διαχειρίσιμα και πιο εύκολο να αποσφαλματωθούν. Μερικά πλεονεκτήματα της χρήσης συναρτήσεων είναι τα ακόλουθα:

- **Οι συναρτήσεις μειώνουν την επανάληψη του κώδικα μέσα σε ένα πρόγραμμα** — Μια συνάρτηση επιτρέπει την εξαγωγή ενός συχνά χρησιμοποιούμενο μπλοκ κώδικα σε ένα ενιαίο στοιχείο/αντικείμενο. Με αυτόν τον τρόπο είναι δυνατό να εκτελεστεί η ίδια εργασία καλώντας μια συνάρτηση

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

όπου χρειάζεται μέσα σε ένα σενάριο χωρίς να χρειάζεται να αντιγράψετε και να επικολλήσετε το ίδιο μπλοκ κώδικα ξανά και ξανά.

- **Οι συναρτήσεις καθιστούν πολύ ευκολότερη τη διατήρηση του κώδικα τον κώδικα** — Δεδομένου ότι μια συνάρτηση που δημιουργείται μία φορά μπορεί να εφαρμοστεί πολλές φορές, ενώ οποιεσδήποτε αλλαγές γίνονται μέσα σε μια συνάρτηση εφαρμόζονται αυτόματα σε όλα τα μέρη χωρίς να επηρεάζουν τα αντίστοιχα αρχεία.
- **Οι συναρτήσεις καθιστούν ευκολότερη την απαλλαγή από την εμφάνιση σφαλμάτων** — Όταν το πρόγραμμα χωρίζεται σε συναρτήσεις, εάν προκύψει οποιοδήποτε σφάλμα, ο προγραμματιστής ξέρει ακριβώς ποια συνάρτηση προκαλεί το σφάλμα και πού να το εντοπίσει. Κατά συνέπεια, η διόρθωση σφαλμάτων γίνεται πολύ απλούστερη.

Ο ορισμός και η κλήση μιας συνάρτησης

Η δήλωση μιας συνάρτησης ξεκινά με τη λέξη-κλειδί της συνάρτησης, ακολουθούμενη από το όνομα της συνάρτησης που θα δημιουργηθεί, έπειτα από παρενθέσεις και τέλος από τον κώδικα της συνάρτησης μεταξύ αγκύλων {}. Για να δηλώσετε μια συνάρτηση, ισχύει η ακόλουθη σύνταξη:

```
function functionName() {
   // Code to be executed
}
```

Αυτό το απλό παράδειγμα εμφανίζει ένα μήνυμα "Hello".

Μόλις οριστεί μια συνάρτηση, μπορεί να κληθεί από οπουδήποτε στο έγγραφο, πληκτρολογώντας το όνομά της ακολουθούμενο από ένα σύνολο παρενθέσεων, όπως το sayHello() στο προαναφερθέν παράδειγμα.

Προσθήκη παραμέτρων στις συναρτήσεις

Οι παράμετροι μπορούν να καθοριστούν κατά τον ορισμό μιας συνάρτησης, για την αποδοχή τιμών εισόδου κατά το χρόνο εκτέλεσης. Οι παράμετροι ενεργούν ως

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

μεταβλητές κράτησης θέσης μέσα σε μια συνάρτηση. Αντικαθίστανται στο χρόνο εκτέλεσης από τις τιμές (γνωστές ως *όρισματα*) που προσφέρονται στη συνάρτηση κατά τη στιγμή ενός αιτήματος.

Οι παράμετροι ορίζονται στην πρώτη γραμμή της συνάρτησης μέσα στο σύνολο των παρενθέσεων, ως εξής:

```
function functionName(parameter1, parameter2, parameter3) {
   // Code to be executed
}
```

Η συνάρτηση displaySum() σε αυτό το παράδειγμα λαμβάνει δύο αριθμούς ως ορίσματα, απλά προσθέστε τους ως ένα και μετά εμφανίστε το αποτέλεσμα στο πρόγραμμα περιήγησης. Δεν υπάρχει όριο για τον καθορισμό των παραμέτρων. Ωστόσο, για κάθε καθορισμένη παράμετρο, ένα αντίστοιχο όρισμα χρειάζεται να περάσει στη συνάρτηση όταν καλείται, αν όχι η τιμή της γίνεται απροσδιόριστη.

Προεπιλεγμένες τιμές για παραμέτρους συνάρτησης

Με το ES6, είναι δυνατόν να καθοριστούν προεπιλεγμένες τιμές για τις παραμέτρους λειτουργίας. Αυτό σημαίνει ότι εάν δεν παρέχονται ορίσματα για να λειτουργήσουν όταν η ES6 καλείται, τότε θα χρησιμοποιηθούν οι προεπιλεγμένες τιμές παραμέτρων. Αυτό το παράδειγμα είναι αρκετά επεξηγηματικό σχετικά με το πόσο χρήσιμη είναι αυτή η δυνατότητα, δεδομένου ότι για να επιτευχθεί το ίδιο αποτέλεσμα, η προηγούμενη διαδικασία ήταν η ίδια.

Επιστροφή τιμών από μια συνάρτηση

Μια συνάρτηση μπορεί να επιστρέψει μια τιμή πίσω στο σενάριο που κάλεσε τη συνάρτηση ως αποτέλεσμα, χρησιμοποιώντας τη δήλωση επιστροφής. Η τιμή μπορεί να είναι οποιουδήποτε τύπου (π.χ. συστοιχίες και αντικείμενα). Η δήλωση επιστροφής (return statement) τυπικά τοποθετείται στην τελευταία γραμμή της συνάρτησης πριν από

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

το κλείσιμο της αγκύλης και τελειώνει με ένα ερωτηματικό, όπως φαίνεται στο παράδειγμα που παρατίθεται εδώ.

Μια συνάρτηση δεν μπορεί να επιστρέψει πολλαπλές τιμές. Παρόλα αυτά, παρόμοια αποτελέσματα μπορούν να ληφθούν επιστρέφοντας μια σειρά τιμών, όπως παρουσιάζεται εδώ.

Δουλεύοντας με τις εκφράσεις συνάρτησεων

Η σύνταξη που χρησιμοποιήθηκε πριν για τη δημιουργία συναρτήσεων ονομάζεται **δήλωση συναρτήσεων (function declaration)**. Υπάρχει μια άλλη σύνταξη για την οικοδόμηση μιας συνάρτησης – έκφρασης συνάρτησης. Μόλις αποθηκευτεί σε μια μεταβλητή, η μεταβλητή μπορεί να χρησιμοποιηθεί ως συνάρτηση. Η σύνταξη της δήλωσης συνάρτησης (function declaration) και της έκφρασης λειτουργίας (function expression) φαίνεται πολύ παρόμοια, αλλά ποικίλλουν στον τρόπο με τον οποίο αξιολογούνται. Όπως παρατηρήθηκε στο προηγούμενο παράδειγμα, η έκφραση συνάρτησης έδωσε μια εξαίρεση όταν έγινε επίκληση πριν οριστεί, αλλά η δήλωση συνάρτησης εκτελέστηκε αποτελεσματικά. Η JS αναλύει τη δήλωση συνάρτησης πριν από την εκτέλεση του προγράμματος. Ως εκ τούτου, δεν κάνει καμία διαφορά εάν το πρόγραμμα επικαλείται τη λειτουργία πριν από τον ορισμό της, καθώς η JavaScript έχει ανυψώσει τη λειτουργία στην κορυφή του τρέχοντος πεδίου στο παρασκήνιο. Μια έκφραση συνάρτησης δεν αξιολογείται μέχρι να ανατεθεί σε μια μεταβλητή. Κατά συνέπεια, εξακολουθεί να είναι απροσδιόριστη όταν γίνεται επίκληση.

Κατανόηση του εύρους μιας μεταβλητής

Οι μεταβλητές μπορούν να δηλωθούν οπουδήποτε στο JS. Ωστόσο, η τοποθεσία της δήλωσης θα καθορίσει την έκταση της διαθεσιμότητας μιας μεταβλητής εντός του προγράμματος JavaScript – αυτή η διαδικασία ονομάζεται επίσης **μεταβλητό πεδίο εφαρμογής**. Από προεπιλογή, οι μεταβλητές που δηλώνονται εντός μιας συνάρτησης έχουν τοπική εμβέλεια, πράγμα που σημαίνει ότι δεν μπορούν να προβληθούν ή να ελεγχθούν εκτός αυτής της συνάρτησης, όπως φαίνεται εδώ. Αν και, οποιεσδήποτε μεταβλητές που δηλώνονται σε ένα πρόγραμμα εκτός μιας συνάρτησης έχουν καθολική

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

εμβέλεια, όπου αυτό το σενάριο βρίσκεται σχετικά με τη συνάρτηση, όπως μπορεί να ελεγχθεί εδώ.

Αντικείμενα JavaScript

Η JavaScript είναι μια γλώσσα που βασίζεται σε αντικείμενα και σχεδόν τα πάντα είναι αντικείμενα ή ενεργούν σαν αντικείμενα. Έτσι, για να συνεργαστούν με τον JS με επιτυχία και αποτελεσματικότητα, οι προγραμματιστές πρέπει να κατανοήσουν πώς λειτουργούν τα αντικείμενα, καθώς και πώς να δημιουργήσουν αντικείμενα και να τα χρησιμοποιήσουν.

Ένα αντικείμενο JavaScript είναι απλά μια συλλογή από ονομαστικές τιμές. Συνήθως αναφέρονται ως ιδιότητες του αντικειμένου. Όντας ένας πίνακας με μια συλλογή τιμών, στην οποία κάθε τιμή έχει ένα δείκτη (ένα αριθμητικό κλειδί) ξεκινώντας από το μηδέν και αυξάνοντας κατά ένα για κάθε τιμή. Ένα αντικείμενο είναι σαν μια συστοιχία, αλλά η διαφορά είναι ότι ο προγραμματιστής ορίζει τα κλειδιά, (όνομα, ηλικία, φύλο, κ.λπ.).

Δημιουργία ενός αντικειμένου

Οι προγραμματιστές μπορούν να δημιουργήσουν αντικείμενα με αγκύλες, συμπεριλαμβανομένης μιας προαιρετικής λίστας ιδιοτήτων επίσης. Μια ιδιότητα μπορεί να είναι ζεύγος “key:value”, στο οποίο το κλειδί (ή το *όνομα της ιδιότητας*) είναι πάντα μια συμβολοσειρά, και η τιμή (ή η *τιμή της ιδιότητας*) μπορεί να είναι οποιοσδήποτε τύπος δεδομένων (συμβολοσειρές, αριθμοί, Booleans, πίνακες, λειτουργίες κ.λπ.). Επιπλέον, οι ιδιότητες με συναρτήσεις όπως οι τιμές τους συχνά ονομάζονται μέθοδοι, για μπορούμε να τις διακρίνουμε εύκολα από άλλα χαρακτηριστικά. Ένα αντικείμενο JS μπορεί να έχει ως εξής. Αυτό το παράδειγμα δημιουργεί ένα αντικείμενο που ονομάζεται person, το οποίο έχει 3 ιδιότητες (όνομα, ηλικία και φύλο) και τη μέθοδο displayName(). Αυτή η μέθοδος εμφανίζει την τιμή του this.name, το οποίο συμφωνεί με το person.name. Αυτός είναι ο ευκολότερος και ιδανικότερος τρόπος για να δημιουργήσετε ένα νέο αντικείμενο στο JF, το οποίο είναι γνωστό ως **object literals syntax**. Τα

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ονόματα των ιδιοτήτων γενικά δεν χρειάζεται να αναφέρονται εκτός αν είναι δεσμευμένες λέξεις, ή αν περιέχουν κενά ή ειδικούς χαρακτήρες (οτιδήποτε άλλο εκτός από γράμματα, αριθμούς και τους χαρακτήρες _ και $), ή αν ξεκινούν με έναν αριθμό, όπως φαίνεται εδώ.

Πρόσβαση στις ιδιότητες του αντικειμένου

Για την πρόσβαση ή τη λήψη της τιμής μιας ιδιότητας, μπορεί να εφαρμοστεί η τελεία (.), καθώς και η σημειογραφία αγκύλων ([]), όπως φαίνεται στο παράδειγμα αυτό. Η εγγραφή και ανάγνωση της σημειογραφίας της τελείας είναι απλούστερη, αλλά δεν μπορεί να χρησιμοποιηθεί. Εάν το όνομα της ιδιότητας δεν είναι έγκυρο (για παράδειγμα, εάν περιέχει ειδικούς χαρακτήρες ή κενά), δεν μπορεί να χρησιμοποιηθεί ο σημειογραφία της τελείας, αλλά η σημειογραφία της αγκύλης. Παρουσιάζει πολύ μεγαλύτερη ευελιξία από τη σημειογραφία της τελείας και επιπλέον επιτρέπει τον εντοπισμό των ονομάτων των ιδιοτήτων (property names) ως μεταβλητές για την αντικατάσταση των γραμμάτων συμβολοσειράς (string literals).

Δημιουργία Βρόχων μέσα από τις Ιδιότητες ενός Αντικειμένου

Οι προγραμματιστές μπορούν να επαναλάβουν ενέργειες μέσα από τα ζεύγη κλειδιών-τιμών ενός αντικειμένου χρησιμοποιώντας το είδος βρόχου for...in. Ο βρόχος for...in υποστηρίζει την επανάληψη εργασιών μέσα από τις ιδιότητες ενός αντικειμένου, όπως φαίνεται στο παράδειγμα εδώ.

Ορισμός ιδιοτήτων αντικειμένου

Παρομοίως, μπορούν να οριστούν νέες ιδιότητες ή να ενημερωθεί η υπάρχουσα χρησιμοποιώντας τη σημειογραφία dot (.) ή bracket ([]), όπως φαίνεται στο παράδειγμα εδώ.

Διαγραφή Ιδιοτήτων Αντικειμένου

Ο τελεστής διαγραφής (delete operator) μπορεί να εφαρμοστεί για την πλήρη διαγραφή ιδιοτήτων από ένα αντικείμενο. Η αφαίρεση είναι ο μόνος τρόπος για να ξεφορτωθείς

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

μια ιδιότητα. Η ρύθμιση μιας ιδιότητας ως απροσδιόριστης ή μηδενικής απλώς αλλάζει την τιμή της, όμως δεν αφαιρεί την ιδιότητα από το αντικείμενο. Έτσι, δεν έχει καμία επίδραση στις μεταβλητές ή στις δηλωμένες συναρτήσεις. Ωστόσο, οι προγραμματιστές πρέπει να αποφεύγουν τον τελεστή διαγραφής για σκοπούς διαγραφής στοιχείων ενός πίνακα, καθώς δεν τροποποιεί το μήκος της συστοιχίας, απλά ρίχνει μια τρύπα σε αυτήν.

Μέθοδοι Κλήσης Αντικειμένου (Calling Object's Methods)

Η μέθοδος κλήσης ενός αντικειμένου μπορεί να προσπελαστεί με τον ίδιο τρόπο που γίνονται προσβάσιμες οι ιδιότητες - χρησιμοποιώντας δηλαδή τη σημειογραφία της τελείας ή εφαρμόζοντας τη σημειογραφία των αγκύλων, όπως μπορεί να εξετασθεί στο παράδειγμα που παρατίθεται εδώ .

Χειρισμός βάσει Τιμής (Value) έναντι Αναφοράς (Reference)

Τα αντικείμενα JS είναι τύποι αναφοράς, πράγμα που σημαίνει ότι όταν ένας προγραμματιστής δημιουργεί τα αντίγραφά τους, αντιγράφουν ουσιαστικά τις αναφορές αυτών των αντικειμένων, ενώ οι πρωτόγονες τιμές (primitive values) όπως οι συμβολοσειρές και οι αριθμοί εκχωρούνται ή αντιγράφονται ως ακέραιες τιμές. Αυτό το παράδειγμα είναι αρκετά ενδεικτικό της έννοιας αυτής. Όπως μπορούμε να παρατηρήσουμε στο παράδειγμα αυτό, δημιουργήθηκε ένα αντίγραφο ενός μεταβλητού μηνύματος το οποίο άλλαξε την τιμή του ίδιου του αντίγραφού του. Και οι δύο μεταβλητές παραμένουν διακριτές και ξεχωριστές. Ωστόσο, εάν η ίδια αρχή εφαρμοστεί σε ένα αντικείμενο, τότε το αποτέλεσμα θα είναι διαφορετικό. Έτσι, οποιεσδήποτε αλλαγές γίνονται στην μεταβλητή χρήστη (variable user) παρεμβαίνουν επίσης στη μεταβλητή ατόμου (object variable), καθώς και οι δύο μεταβλητές αναφέρονται στο ίδιο αντικείμενο. Επομένως, μια απλή αντιγραφή του αντικειμένου δεν το κλωνοποιεί πραγματικά, αλλά αντιγράφει ουσιαστικά την αναφορά σε αυτό το αντικείμενο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

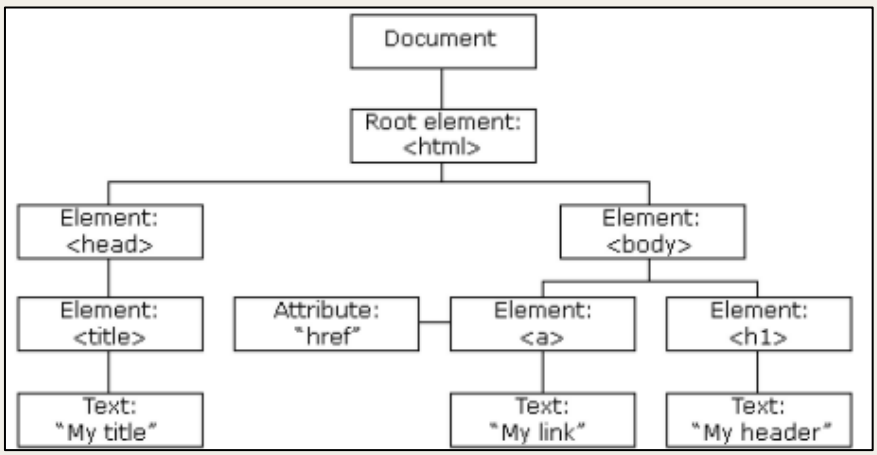
5.2. JavaScript & DOM

Τι είναι το Μοντέλο Αντικειμένων Εγγράφου (DOM);

Το DOM δημιουργείται από το πρόγραμμα περιήγησης όταν μια ιστοσελίδα φορτώνεται σε έγγραφα HTML ή XML. Χρησιμοποιείται για τον καθορισμό της λογικής δομής αυτών των εγγράφων και για την πρόσβαση και την τροποποίηση των στοιχείων τους.

Το HTML DOM αναφέρεται στο Μοντέλο Αντικειμένων Εγγράφου των εγγράφων HTML, ενώ το XML DOM αναφέρεται στο Μοντέλο Αντικειμένων Εγγράφου των εγγράφων XML. Στο υποκεφάλαιο αυτό, θα εστιάσουμε στο HTML DOM που μπορεί να χρησιμοποιηθεί για την πρόσβαση και την επεξεργασία εγγράφων HTML μέσω της JavaScript.

Το DOM κατασκευάζεται ως μια δομή δέντρου όπου οι καταχωρήσεις των αντικειμένων σε αυτό οργανώνονται ιεραρχικά. Τα αντικείμενα αυτά περιλαμβάνουν όλα τα μέρη ενός εγγράφου HTML όπως στοιχεία, ιδιότητες, κείμενο κ.λπ. Αυτά τα επιμέρους αντικείμενα του δέντρου είναι επίσης γνωστά ως κόμβοι (nodes) και αποτελούνται από γονικούς και θυγατρικούς κόμβους (parent and child nodes). Σε γραφική μορφή, η ιεραρχική δομή του δέντρου DOM μοιάζει με το διάγραμμα που παρατίθεται παρακάτω.



Εικόνα 90 – *Δέντρο αντικειμένων DOM HTML (**Πηγή**: https://www.w3schools.com/js/js_htmldom.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εντός του HTML DOM, η JavaScript μπορεί να κάνει τα εξής:

- Αλλαγή όλων των στοιχείων HTML στη σελίδα
- Αλλαγή όλων των ιδιοτήτων HTML στη σελίδα
- Αλλαγή όλων των στυλ CSS στη σελίδα
- Κατάργηση υπαρχόντων στοιχείων και ιδιοτήτων HTML
- Προσθήκη νέων στοιχείων και ιδιοτήτων HTML
- Αντίδραση σε όλα τα υπάρχοντα συμβάντα HTML στη σελίδα
- Δημιουργία νέων συμβάντων HTML στη σελίδα

Επιλογείς DOM της JavaScript

Επιλογή στοιχείων DOM στο JavaScript

Η JavaScript χρησιμοποιείται για να λάβει ή να τροποποιήσει το περιεχόμενο ή τις τιμές των στοιχείων HTML της ιστοσελίδας και να εφαρμόσει κάποια ειδικά εφέ, όπως κινούμενα σχέδια ή απόκρυψη (hide). Για να μπορέσετε να εκτελέσετε οποιαδήποτε ενέργεια, πρέπει να βρείτε ή να επιλέξετε το HTML στοιχείο-στόχος (target element).

Θα εξετάσουμε μερικούς από τους πιο συνηθισμένους τρόπους επιλογής στοιχείων σε μια σελίδα και της επεξεργασίας τους στην JavaScript.

Επιλογή των στοιχείων που βρίσκονται στην κορυφή του δέντρου DOM (Topmost Elements)

Τα στοιχεία που βρίσκονται στην κορυφή μπορούν να γίνουν άμεσα προσβάσιμα ως ιδιότητες εγγράφου.

Για παράδειγμα, για να αποκτήσετε πρόσβαση σε <html> ένα στοιχείο, χρησιμοποιήστε την ιδιότητα document.documentElement. Για <head> ένα στοιχείο, μπορείτε να χρησιμοποιήσετε την ιδιότητα document.head <body> ή την ιδιότητα document.body.

Ας δούμε ένα παράδειγμα για να κατανοήσουμε καλύτερα αυτή τη λειτουργία.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Select Topmost Elements</title>
</head>
<body>
    <script>
    // Display lang attribute value of html element
    alert(document.documentElement.getAttribute("lang")); // Outputs: en

    // Set background color of body element
    document.body.style.background = "yellow";

    // Display tag name of the head element's first child
    alert(document.head.firstElementChild.nodeName); // Outputs: meta
    </script>
</body>
</html>
```

Εικόνα 91 – *Παράδειγμα στοιχείων που βρίσκονται στην κορυφή του δέντρου DOM (**Πηγή**:* *https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

* Είναι σημαντικό να σημειωθεί ότι το έγγραφο.body δεν πρέπει να χρησιμοποιείται πριν από το <body> στοιχείο, δεδομένου ότι θα επιστρέψει την τιμή null. Το πρόγραμμα πρέπει να περάσει πρώτα μέσα από το <body> στοιχείο για να αποκτήσει πρόσβαση στην ιδιότητα document.body.

Ας δούμε το παράδειγμα πιο κάτω για να καταλάβουμε γιατί η τίμη <body> θα είναι μηδενική (null):

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Document.body Demo</title>
    <script>
    alert("From HEAD: " + document.body); // Outputs: null (since <body> is not parsed yet)
    </script>
</head>
<body>
    <script>
    alert("From BODY: " + document.body); // Outputs: HTMLBodyElement
    </script>
</body>
</html>
```

Εικόνα 92 – *Παράδειγμα στοιχείων που βρίσκονται στην κορυφή του δέντρου DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Αυτό το παράδειγμα δείχνει αυτό που είδαμε στην αρχή για τις ιεραρχικές σχέσεις που υπάρχουν μεταξύ των κόμβων. Θα πρέπει να έχετε κατά νου ότι για να αποκτήσετε πρόσβαση στην ιδιότητα document.body, θα πρέπει να ξεκινήσετε από το <body> στοιχείο για να αποφύγετε μηδενικές τιμές.

Επιλογή στοιχείων με αναγνωριστικό (ID)

Αν θέλετε να βρείτε ή να επιλέξετε ένα στοιχείο HTML, ο ευκολότερος τρόπος είναι να το επιλέξετε με βάση το μοναδικό αναγνωριστικό του. Μπορείτε να το κάνετε αυτό με τη μέθοδο getElementById ().

Στο ακόλουθο παράδειγμα, επιλέγεται και επισημαίνεται το στοιχείο που έχει αναγνωριστικό ιδιότητας id ="Mark":

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Select Element by ID</title>
</head>
<body>
    <p id="mark">This is a paragraph of text.</p>
    <p>This is another paragraph of text.</p>

    <script>
    // Selecting element with id mark
    var match = document.getElementById("mark");

    // Highlighting element's background
    match.style.background = "yellow";
    </script>
</body>
</html>
```

Εικόνα 93 – *Επιλογή στοιχείων με αναγνωριστικό (ID) (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Η μέθοδος getElementById () χρησιμοποιείται για την επιστροφή του στοιχείου ως αντικειμένου αν βρεθεί ένα αντίστοιχο στοιχείο. Διαφορετικά, θα επιστρέψει μηδενική τιμή (null).

* Λάβετε υπόψη ότι οποιοδήποτε στοιχείο HTML μπορεί να έχει ένα αναγνωριστικό ιδιότητας, το οποίο πρέπει να είναι μια **μοναδική τιμή μέσα σε μια σελίδα**. Αυτό ουσιαστικά σημαίνει ότι κανένα στοιχείο δεν έχει το ίδιο αναγνωριστικό.

Επιλογή στοιχείων ανά όνομα κλάσης

Αν θέλετε να επιλέξετε όλα τα στοιχεία με συγκεκριμένα ονόματα κλάσης, χρησιμοποιήστε τη μέθοδο getElementsByClassName (). Η επιλογή αυτή θα επιστρέψει ένα αντικείμενο που μοιάζει με πίνακα όλων των θυγατρικών στοιχείων που έχουν όλα τα καθορισμένα ονόματα κλάσης.

Ας δούμε ένα παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Select Elements by Class Name</title>
</head>
<body>
    <p class="test">This is a paragraph of text.</p>
    <div class="block test">This is another paragraph of text.</div>
    <p>This is one more paragraph of text.</p>

    <script>
    // Selecting elements with class test
    var matches = document.getElementsByClassName("test");

    // Displaying the selected elements count
    document.write("Number of selected elements: " + matches.length);

    // Applying bold style to first element in selection
    matches[0].style.fontWeight = "bold";

    // Applying italic style to last element in selection
    matches[matches.length - 1].style.fontStyle = "italic";

    // Highlighting each element's background through loop
    for(var elem in matches) {
        matches[elem].style.background = "yellow";
    }
    </script>
</body>
</html>
```

Εικόνα 94 – *Επιλογή στοιχείων ανά όνομα κλάσης (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Επιλογή στοιχείων ανά όνομα ετικέτας

Αν θέλετε να επιλέξετε στοιχεία με το όνομα της ετικέτας τους (tag), χρησιμοποιήστε τη μέθοδο getElementsByTagName (). Αυτή η μέθοδος θα επιστρέψει επίσης ένα αντικείμενο που μοιάζει με πίνακα όλων των θυγατρικών στοιχείων που έχουν το καθορισμένο όνομα ετικέτας.

Ας δούμε ένα παράδειγμα για να το κατανοήσουμε λίγο καλύτερα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Select Elements by Tag Name</title>
</head>
<body>
    <p>This is a paragraph of text.</p>
    <div class="test">This is another paragraph of text.</div>
    <p>This is one more paragraph of text.</p>

    <script>
    // Selecting all paragraph elements
    var matches = document.getElementsByTagName("p");

    // Printing the number of selected paragraphs
    document.write("Number of selected elements: " + matches.length);

    // Highlighting each paragraph's background through loop
    for(var elem in matches) {
        matches[elem].style.background = "yellow";
    }
    </script>
</body>
</html>
```

Εικόνα 95 – *Επιλογή στοιχείων ανά όνομα ετικέτας (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Επιλογή στοιχείων με επιλογείς CSS

Οι Επιλογείς CSS προσφέρουν έναν πολύ δυνατό και αποτελεσματικό τρόπο επιλογής στοιχείων HTML σε ένα έγγραφο. Για να επιλέξετε στοιχεία που ταιριάζουν με τον καθορισμένο επιλογέα CSS, μπορείτε να χρησιμοποιήσετε τη μέθοδο QUERYSelectorAll ().

Αυτή η μέθοδος θα επιστρέψει μια λίστα όλων των στοιχείων που ταιριάζουν με τους καθορισμένους επιλογείς.

Ακολουθήστε το παρακάτω παράδειγμα για να δείτε πώς μπορείτε να εξετάσετε αυτήν τη λίστα σαν ένα πίνακα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Select Elements with CSS Selectors</title>
</head>
<body>
    <ul>
        <li>Bread</li>
        <li class="tick">Coffee</li>
        <li>Pineapple Cake</li>
    </ul>

    <script>
    // Selecting all li elements
    var matches = document.querySelectorAll("ul li");

    // Printing the number of selected li elements
    document.write("Number of selected elements: " + matches.length + "<hr>")

    // Printing the content of selected li elements
    for(var elem of matches) {
        document.write(elem.innerHTML + "<br>");
    }

    // Applying line through style to first li element with class tick
    matches = document.querySelectorAll("ul li.tick");
    matches[0].style.textDecoration = "line-through";
    </script>
</body>
</html>
```

Εικόνα 96 – *Επιλογή στοιχείων με επιλογείς CSS (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

* Παρακαλώ σημειώστε ότι η μέθοδος querySelectorAll() υποστηρίζει ψευδο-κλάσεις CSS (pseudo-class) όπως :first-child, : last-child, :hover, κλπ. Ωστόσο, η μέθοδος αυτή θα επιστρέφει πάντα μια κενή λίστα για ψευδο-στοιχεία CSS (pseudo-element) όπως :: before, ::after, ::first-line, κλπ.

Στυλιστική ανάπτυξη DOM σε JavaScript

Στυλιστική ανάπτυξη στοιχείων DOM σε JavaScript

Η JavaScript παρέχει δυναμικούς τρόπους με τους οποίους μπορείτε να αλλάξετε την εμφάνιση των εγγράφων HTML μέσω της εφαρμογής διάφορων στυλ σε στοιχεία HTML. Σχεδόν όλα τα στυλ στοιχείων μπορούν να οριστούν όπως γραμματοσειρές, χρώματα,

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

περιθώρια, περιγράμματα, εικόνες φόντου, στοίχιση κειμένου, πλάτος και ύψος, θέση και ούτω καθεξής.

Στο σημείο αυτό, θα εξετάσουμε τις διάφορες μεθόδους που μπορούν να χρησιμοποιηθούν για να ορισμό ενός στυλ στην JavaScript.

Τρόπος εφαρμογής ενσωματωμένων στυλ σε στοιχεία (Inline Styles)

Η ιδιότητα στυλ χρησιμοποιείται για την εφαρμογή ενσωματωμένων στυλ απευθείας σε ένα συγκεκριμένο στοιχείο HTML. Η ιδιότητα στυλ χρησιμοποιείται στην JavaScript για να λάβει ή να καθορίσει το ενσωματωμένο στυλ ενός στοιχείου.

Στο ακόλουθο παράδειγμα, οι ιδιότητες χρώματος και γραμματοσειράς θα οριστούν για ένα στοιχείο με το αναγνωριστικό id="intro":

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Set Inline Styles Demo</title>
</head>
<body>
    <p id="intro">This is a paragraph.</p>
    <p>This is another paragraph.</p>

    <script>
    // Selecting element
    var elem = document.getElementById("intro");

    // Appling styles on element
    elem.style.color = "blue";
    elem.style.fontSize = "18px";
    elem.style.fontWeight = "bold";
    </script>
</body>
</html>
```

Εικόνα 97 – *Παράδειγμα Ενσωματωμένων Στυλ σε Στοιχεία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Συμβάσεις Ονοματολογίας Ιδιοτήτων CSS στην JavaScript

Είναι σημαντικό να αναφερθεί ότι πολλές από τις ιδιότητες CSS περιέχουν παύλες (-) στα ονόματά τους, όπως οι ιδιότητες μεγέθους γραμματοσειράς, εικόνας φόντου, διακόσμησης κειμένου κ.λπ. Ωστόσο, στην JavaScript, η παύλα είναι ένας δεσμευμένος τελεστής που συνεπάγεται με το αρνητικό πρόσημο "μείον" . Ως εκ τούτου, δεν είναι δυνατόν να γράψετε μια έκφραση με αυτόν τον τρόπο: elem.style.font-size.

Για την επίλυση του θέματος, τα ονόματα ιδιοτήτων CSS στην JavaScript που περιέχουν μία ή περισσότερες παύλες μετατρέπονται σε κεφαλαιοποιημένες λέξεις. Αυτό ουσιαστικά σημαίνει ότι οι παύλες αφαιρούνται και το πρώτο γράμμα μετά την παύλα κεφαλαιοποιείται. Για παράδειγμα, το μέγεθος γραμματοσειράς της ιδιότητας CSS γίνεται μέγεθος γραμματοσειράς στην ιδιότητα DOM.

Λήψη πληροφοριών στυλ από στοιχεία

Η ιδιότητα στυλ χρησιμοποιείται επίσης για να λάβει τα στυλ που εφαρμόζονται στα στοιχεία HTML.

Το παρακάτω παράδειγμα θα λάβει πληροφορίες στυλ από το στοιχείο με αναγνωριστικό id="intro":

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>JS Get Element's Style Demo</title>
</head>
<body>
    <p id="intro" style="color:red; font-size:20px;">This is a paragraph.</p>
    <p>This is another paragraph.</p>

    <script>
    // Selecting element
    var elem = document.getElementById("intro");

    // Getting style information from element
    alert(elem.style.color);  // Outputs: red
    alert(elem.style.fontSize);  // Outputs: 20px
    alert(elem.style.fontStyle);  // Outputs nothing
    </script>
</body>
</html>
```

Εικόνα 98 – *Παράδειγμα ιδιότητας στυλ (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Η ιδιότητα στυλ δεν είναι η πιο χρήσιμη όταν πρόκειται για τη λήψη πληροφοριών στυλ από τα στοιχεία, δεδομένου ότι επιστρέφει μόνο τους κανόνες στυλ που ορίζονται στην ιδιότητα στυλ του στοιχείου και όχι εκείνους που προέρχονται από αλλού, όπως οι κανόνες στυλ στα ενσωματωμένα φύλλα στυλ, ή τα εξωτερικά φύλλα στυλ.

Αν θέλετε να λάβετε τις τιμές όλων των ιδιοτήτων CSS που χρησιμοποιούνται για την ερμηνεία ενός στοιχείου, μπορείτε να χρησιμοποιήσετε τη μέθοδο window.getComputedStyle (), όπως φαίνεται στο παρακάτω παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>JS Get Computed Style Demo</title>
<style type="text/css">
    #intro {
        font-weight: bold;
        font-style: italic;
    }
</style>
</head>
<body>
    <p id="intro" style="color:red; font-size:20px;">This is a paragraph.</p>
    <p>This is another paragraph.</p>

    <script>
    // Selecting element
    var elem = document.getElementById("intro");

    // Getting computed style information
    var styles = window.getComputedStyle(elem);

    alert(styles.getPropertyValue("color"));  // Outputs: rgb(255, 0, 0)
    alert(styles.getPropertyValue("font-size"));  // Outputs: 20px
    alert(styles.getPropertyValue("font-weight"));  // Outputs: 700
    alert(styles.getPropertyValue("font-style"));  // Outputs: italic
    </script>
</body>
</html>
```

Εικόνα 99 – *Παράδειγμα window.getComputedStyle() (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

* Λάβετε υπόψη ότι η τιμή 700 για την ιδιότητα του βάρους μιας γραμματοσειράς CSS είναι η ίδια με τη λέξη-κλειδί Bold. Η λέξη-κλειδί "κόκκινο χρώμα" (colour keyword) είναι η ίδια με την μορφή δεκαεξαδικής τιμής rgb(255,0,0), η οποία είναι η σημειογραφία rgb ενός χρώματος.

Προσθήκη κλάσεων CSS στα στοιχεία

Ένας άλλος τρόπος για να πάρετε ή να καθορίσετε κλάσεις CSS σε στοιχεία HTML είναι χρησιμοποιώντας την ιδιότητα className. Η κλάση είναι μια δεσμευμένη λέξη στην JavaScript. Έτσι, η JavaScript χρησιμοποιεί την ιδιότητα className για να αναφερθεί στην τιμή της ιδιότητας κλάσης HTML.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ας δούμε το ακόλουθο παράδειγμα για να μάθουμε πώς να προσθέτουμε μια νέα κλάση ή να αντικαταστήσουμε όλες τις υπάρχουσες κλάσεις σε ένα στοιχείο με το αναγνωριστικό id="info":

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>JS Add or Replace CSS Classes Demo</title>
<style>
    .highlight {
        background: yellow;
    }
</style>
</head>
<body>
    <div id="info" class="disabled">Something very important!</div>

    <script>
    // Selecting element
    var elem = document.getElementById("info");

    elem.className = "note";  // Add or replace all classes with note class
    elem.className += " highlight";  // Add a new class highlight
    </script>
</body>
</html>
```

Εικόνα 100 – *Παράδειγμα προσθήκης κλάσεων στα στοιχεία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Ένας ακόμα καλύτερος τρόπος για να εργαστείτε με τις κλάσεις CSS είναι χρησιμοποιώντας την ιδιότητα classList για να λάβετε, να ρυθμίσετε ή να αφαιρέσετε εύκολα κλάσεις CSS από ένα στοιχείο. Αυτή η ιδιότητα υποστηρίζεται σε όλα τα μεγάλα προγράμματα περιήγησης εκτός από την Internet Explorer, πριν από την έκδοση 10.

Ας δούμε ένα παράδειγμα αυτής της ιδιότητας:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>JS classList Demo</title>
<style>
    .highlight {
        background: yellow;
    }
</style>
</head>
<body>
    <div id="info" class="disabled">Something very important!</div>

    <script>
    // Selecting element
    var elem = document.getElementById("info");

    elem.classList.add("hide");  // Add a new class
    elem.classList.add("note", "highlight");  // Add multiple classes
    elem.classList.remove("hide"); // Remove a class
    elem.classList.remove("disabled", "note"); // Remove multiple classes
    elem.classList.toggle("visible"); // If class exists remove it, if not add it

    // Determine if class exist
    if(elem.classList.contains("highlight")) {
        alert("The specified class exists on the element.");
    }
    </script>
</body>
</html>
```

Εικόνα 101 – *Ιδιότητα classList – Παράδειγμα Προσθήκης Κλάσεων σε Στοιχεία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Λήψη (Get) και Ορισμός (Set) Ιδιοτήτων DOM σε JavaScript

Δουλεύοντας με τις ιδιότητες

Οι ιδιότητες (attributes) είναι ειδικές λέξεις που χρησιμοποιούνται μέσα στην ετικέτα εκκίνισης ενός στοιχείου HTML για τον έλεγχο της συμπεριφοράς της ετικέτας ή την παροχή περισσότερων πληροφοριών σχετικά με αυτή.

Στο κεφάλαιο αυτό, θα εξετάσουμε τις διάφορες μεθόδους προσθήκης, αφαίρεσης ή αλλαγής ενός στοιχείου HTML.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Λήψη τιμής ιδιότητας ενός στοιχείου

Για να πάρετε την τρέχουσα τιμή της ιδιότητας ενός στοιχείου, μπορείτε να χρησιμοποιήσετε τη μέθοδο getAttribute (). Εάν η συγκεκριμένη ιδιότητα δεν βρεθεί στο στοιχείο, θα επιστρέψει την τιμή null.

Ας δούμε το ακόλουθο παράδειγμα παρακάτω:

```
<a href="https://www.google.com/" target="_blank" id="myLink">Google</a>

<script>
    // Selecting the element by ID attribute
    var link = document.getElementById("myLink");

    // Getting the attributes values
    var href = link.getAttribute("href");
    alert(href); // Outputs: https://www.google.com/

    var target = link.getAttribute("target");
    alert(target); // Outputs: _blank
</script>
```

Εικόνα 102 – *getAttribute() method – Παράδειγμα Λήψης Τιμής Ιδιότητας Στοιχείου (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-get-set-attributes.php)*

Ορισμός ιδιοτήτων στα στοιχεία

Αν θέλετε να ορίσετε μια ιδιότητα σε ένα καθορισμένο στοιχείο, μπορείτε να χρησιμοποιήσετε τη μέθοδο setAttribute (). Εάν η ιδιότητα υπάρχει ήδη στο στοιχείο, η τιμή θα ενημερωθεί. Εάν όχι, θα προστεθεί ένα νέο χαρακτηριστικό με καθορισμένο όνομα και τιμή.

Στο ακόλουθο παράδειγμα, θα προσθέσουμε μια κλάση και ένα απενεργοποιημένη ιδιότητα στο <button> στοιχείο:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```html
<button type="button" id="myBtn">Click Me</button>

<script>
    // Selecting the element
    var btn = document.getElementById("myBtn");

    // Setting new attributes
    btn.setAttribute("class", "click-btn");
    btn.setAttribute("disabled", "");
</script>
```

Εικόνα 103 – *setAttribute() method – Παράδειγμα Ορισμού Ιδιοτήτων στα Στοιχεία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-get-set-attributes.php)*

Αν θέλετε να ενημερώσετε ή να αλλάξετε την τιμή μιας υπάρχουσας ιδιότητας σε ένα στοιχείο, μπορείτε επίσης να χρησιμοποιήσετε τη μέθοδο setAttribute ().

Ας δούμε ένα παράδειγμα που θα ενημερώσει την τιμή της υπάρχουσας ιδιότητας href ενός στοιχείου σε οξυγώνιες αγκύλες (<a>):

```html
<a href="#" id="myLink">Tutorial Republic</a>

<script>
    // Selecting the element
    var link = document.getElementById("myLink");

    // Changing the href attribute value
    link.setAttribute("href", "https://www.tutorialrepublic.com");
</script>
```

Εικόνα 104 – *setAttribute() method – Παράδειγμα Ενημέρωσης ή Αλλαγής Ιδιοτήτων στα Στοιχεία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-get-set-attributes.php)*

Αφαίρεση ιδιοτήτων από στοιχεία

Για να αφαιρέσετε μια ιδιότητα από ένα συγκεκριμένο στοιχείο, μπορείτε να χρησιμοποιήσετε τη μέθοδο removeAttribute ().

Θυμηθείτε την ιδιότητα href που αλλάξαμε από το στοιχείο αγκύρωσης. Τώρα θα το αφαιρέσουμε στο ακόλουθο παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<a href="https://www.google.com/" id="myLink">Google</a>

<script>
    // Selecting the element
    var link = document.getElementById("myLink");

    // Removing the href attribute
    link.removeAttribute("href");
</script>
```

Εικόνα 105 – *Η μέθοδος removeAttribute() – Παράδειγμα Αφαίρεσης Ιδιοτήτων από τα Στοιχεία (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-get-set-attributes.php)*

Επεξεργασία DOM στην JavaScript

Επεξεργασία Στοιχείων DOM στην JavaScript

Μέχρι στιγμής, έχουμε μάθει πώς να επιλέγουμε και να διαμορφώνουμε στοιχεία HTML DOM. Τώρα, θα μάθουμε πώς να προσθέτουμε ή να αφαιρούμε στοιχεία DOM με δυναμικό τρόπο, πώς να λαμβάνουμε το περιεχόμενό τους και πολλά άλλα.

Προσθήκη Νέων Στοιχείων στο DOM

Η μέθοδος document.createElement () χρησιμοποιείται για τη δημιουργία ενός νέου στοιχείου σε ένα έγγραφο HTML. Δημιουργεί ένα νέο στοιχείο. Ωστόσο, δεν το προσθέτει στο DOM.

Απαιτείται ένα ξεχωριστό βήμα για την προσθήκη του στο Dom, όπως φαίνεται στο παρακάτω παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">Hello World!</h1>
    <p id="hint">This is a simple paragraph.</p>
</div>

<script>
// Creating a new div element
var newDiv = document.createElement("div");

// Creating a text node
var newContent = document.createTextNode("Hi, how are you doing?");

// Adding the text node to the newly created div
newDiv.appendChild(newContent);

// Adding the newly created element and its content into the DOM
var currentDiv = document.getElementById("main");
document.body.appendChild(newDiv, currentDiv);
</script>
```

Εικόνα 106 – *Παράδειγμα Προσθήκης Νέων Στοιχείων (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Στο παράδειγμα που μόλις είδαμε, το appendChild() χρησιμοποιείται για να προσθέσει το νέο στοιχείο στο τέλος οποιωνδήποτε άλλων θυγατρικών κόμβων κάτω από τον καθορισμένο γονικό κόμβο.

Έχετε επίσης την επιλογή να προσθέσετε το νέο στοιχείο πριν από οποιονδηποτε θυγατρικό κόμβο, όπως φαίνεται στο παρακάτω παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">Hello World!</h1>
    <p id="hint">This is a simple paragraph.</p>
</div>

<script>
// Creating a new div element
var newDiv = document.createElement("div");

// Creating a text node
var newContent = document.createTextNode("Hi, how are you doing?");

// Adding the text node to the newly created div
newDiv.appendChild(newContent);

// Adding the newly created element and its content into the DOM
var currentDiv = document.getElementById("main");
document.body.insertBefore(newDiv, currentDiv);
</script>
```

Εικόνα 107 – *Παράδειγμα Προσθήκης Νέων Στοιχείων (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Λήψη ή Ρύθμιση περιεχομένων HTML σε DOM

Αν θέλετε να λάβετε ή να ορίσετε τα περιεχόμενα των στοιχείων HTML, μπορείτε να χρησιμοποιήσετε την ιδιότητα innerHTML. Αυτή η ιδιότητα χρησιμοποιείται για τη ρύθμιση ή τη λήψη της σήμανσης HTML μέσα στο στοιχείο, το οποίο έχει το περιεχόμενο μεταξύ των ετικετών ανοίγματος και κλεισίματος.

Ας δούμε ένα παράδειγμα για να το κατανοήσουμε καλύτερα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">Hello World!</h1>
    <p id="hint">This is a simple paragraph.</p>
</div>

<script>
// Getting inner HTML conents
var contents = document.getElementById("main").innerHTML;
alert(contents); // Outputs inner html contents

// Setting inner HTML contents
var mainDiv = document.getElementById("main");
mainDiv.innerHTML = "<p>This is <em>newly inserted</em> paragraph.</p>";
</script>
```

Εικόνα 108 – *Λήψη ή Ρύθμιση περιεχομένων HTML σε DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Όπως μπορούμε να δούμε στο παράδειγμα, τα νέα στοιχεία εισάγονται πολύ εύκολα στο Dom με την εσωτερική ιδιότητα innerHTML. Ωστόσο, αυτή η ιδιότητα αντικαθιστά όλο το υπάρχον περιεχόμενο ενός στοιχείου.

Ως εκ τούτου, εάν δεν θέλετε να αντικαταστήσετε τα υπάρχοντα περιεχόμενα ενός στοιχείου, μπορείτε να χρησιμοποιήσετε τη μέθοδο insertAdjacentHTML (). Η μέθοδος αυτή λαμβάνει δύο παραμέτρους: την HTML που θα εισαχθεί και τη θέση της. Η θέση πρέπει να είναι μία από τις ακόλουθες: "prebegin", "afterbegin", "beforeend" και "afterend". Είναι επίσης σημαντικό να σημειωθεί ότι αυτή η μέθοδος υποστηρίζεται από όλα τα μεγάλα προγράμματα περιήγησης.

Στο παρακάτω παράδειγμα, μπορείτε να δείτε πώς λειτουργεί η τοποθέτηση:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<!-- beforebegin -->
<div id="main">
    <!-- afterbegin -->
    <h1 id="title">Hello World!</h1>
    <!-- beforeend -->
</div>
<!-- afterend -->

<script>
// Selecting target element
var mainDiv = document.getElementById("main");

// Inserting HTML just before the element itself, as a previous sibling
mainDiv.insertAdjacentHTML('beforebegin', '<p>This is paragraph one.</p>');

// Inserting HTML just inside the element, before its first child
mainDiv.insertAdjacentHTML('afterbegin', '<p>This is paragraph two.</p>');

// Inserting HTML just inside the element, after its last child
mainDiv.insertAdjacentHTML('beforeend', '<p>This is paragraph three.</p>');

// Inserting HTML just after the element itself, as a next sibling
mainDiv.insertAdjacentHTML('afterend', '<p>This is paragraph four.</p>');
</script>
```

Εικόνα 109 – *insertAdjacentHTML() method - Παράδειγμα Λήψης ή Ρύθμισης Περιεχομένων HTML στο DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

* Λάβετε υπόψη ότι για τις θέσεις beforebegin και afterend για να λειτουργήσει ο κόμβος πρέπει να είναι στο δέντρο Dom και να έχει ένα γονικό στοιχείο.
* Επίσης, κατά την εισαγωγή ενός HTML σε μια σελίδα, προσέξτε να μην χρησιμοποιείτε είσοδο χρήστη που δεν έχει διαφύγει (escaped)/καθαριστεί (sanitised), για να αποτρέψετε τις επιθέσεις τύπου XSS.

Αφαίρεση υπαρχόντων στοιχείων από το DOM

Για να αφαιρέσετε έναν θυγατρικό κόμβο από το Dom, μπορείτε να χρησιμοποιήσετε τη μέθοδο removeChild (). Αυτή η μέθοδος θα επιστρέψει επίσης τον κόμβο που αφαιρέθηκε.

Ας δούμε το ακόλουθο παράδειγμα παρακάτω:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">Hello World!</h1>
    <p id="hint">This is a simple paragraph.</p>
</div>

<script>
var parentElem = document.getElementById("main");
var childElem = document.getElementById("hint");
parentElem.removeChild(childElem);
</script>
```

Εικόνα 110 – *Αφαίρεση υπαρχόντων στοιχείων από το DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Μπορείτε επίσης να αφαιρέσετε το θυγατρικό στοιχείο χωρίς να γνωρίζετε το γονικό στοιχείο. Μπορείτε να βρείτε το θυγατρικό στοιχείο και να χρησιμοποιήσετε την ιδιότητα parentNode για να βρείτε τον γονέα του. Θα επιστρέψει τον γονέα του δοσμένου κόμβου στο δέντρο DOM.

```
<div id="main">
    <h1 id="title">Hello World!</h1>
    <p id="hint">This is a simple paragraph.</p>
</div>

<script>
var childElem = document.getElementById("hint");
childElem.parentNode.removeChild(childElem);
</script>
```

Εικόνα 111 – *Αφαίρεση υπαρχόντων στοιχείων από το DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Αντικατάσταση υπαρχόντων στοιχείων στο DOM

Έχετε επίσης την επιλογή να αντικαταστήσετε ένα στοιχείο στο HTML DOM με ένα άλλο χρησιμοποιώντας τη μέθοδο replaceChild (). Η μέθοδος αυτή λαμβάνει δύο παραμέτρους: τον κόμβο που πρόκειται να εισαχθεί και τον κόμβο που πρόκειται να

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

αντικατασταθεί. Η σύνταξη χρησιμοποιείται ως εξής: parentNode.replaceChild (newChild, oldChild);

```
<div id="main">
    <h1 id="title">Hello World!</h1>
    <p id="hint">This is a simple paragraph.</p>
</div>

<script>
var parentElem = document.getElementById("main");
var oldPara = document.getElementById("hint");

// Creating new elememt
var newPara = document.createElement("p");
var newContent = document.createTextNode("This is a new paragraph.");
newPara.appendChild(newContent);

// Replacing old paragraph with newly created paragraph
parentElem.replaceChild(newPara, oldPara);
</script>
```

Εικόνα 112 – *Αντικατάσταση υπαρχόντων στοιχείων στο DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Πλοήγηση στο DOM της JavaScript

Πλοήγηση μεταξύ κόμβων DOM

Μέχρι τώρα, θα πρέπει να σχηματίσουμε μια καλύτερη ιδέα για τον τρόπο με τον οποίο επιλέγονται τα επιμέρους στοιχεία σε μια ιστοσελίδα. Υπάρχουν πολλές περιπτώσεις όπου θα πρέπει να έχετε πρόσβαση σε στοιχεία θυγατρικών και γονικών κόμβων ή προγόνων (ancestor). Στην αρχή αυτού του υποκεφάλαιου, έχουμε αναφερθεί στους κόμβους (nodes) και τώρα θα δούμε πώς μπορούμε να έχουμε πρόσβαση στους διαφορετικούς τύπους κόμβων που υπάρχουν.

Οι DOM κόμβοι έχουν αρκετές ιδιότητες και μεθόδους που σας επιτρέπουν να περιηγηθείτε ή να διασχίσετε τη δομή του δέντρου DOM και να κάνετε τις απαραίτητες αλλαγές αρκετά εύκολα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πρόσβαση στους θυγατρικούς κόμβους

Οι ιδιότητες firstChild και lastChild σας επιτρέπουν να έχετε πρόσβαση στον πρώτο και τελευταίο άμεσο θυγατρικό κόμβο ενός κόμβου αντίστοιχα. Εάν ένας κόμβος δεν έχει θυγατρικό στοιχείο, θα επιστρέψει την τιμή null.

Ας ρίξουμε μια ματιά στο παρακάτω παράδειγμα:

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var main = document.getElementById("main");
console.log(main.firstChild.nodeName); // Prints: #text

var hint = document.getElementById("hint");
console.log(hint.firstChild.nodeName); // Prints: SPAN
</script>
```

Εικόνα 113 – *Παράδειγμα πρόσβασης σε θυγατρικούς κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

* Σημειώστε ότι ο κόμβος Name είναι μια ιδιότητα μόνο για ανάγνωση, η οποία επιστρέφει το όνομα του τρέχοντος κόμβου ως συμβολοσειρά. Για παράδειγμα, θα επιστρέψει το όνομα ετικέτας ενός κόμβου στοιχείου, #text για τον κόμβο κειμένου, #comment για τον κόμβο σχολίου, #document για τον κόμβο εγγράφου και ούτω καθεξής.

Στο παράδειγμα που μόλις είδαμε, το nodeName του πρώτου θυγατρικού κόμβου των κύριων στοιχείων DIV επέστρεψε το #text αντί για H1. Αυτό συμβαίνει επειδή ο κενός χώρος, δηλαδή τα διαστήματα (spaces), οι καρτέλες (tabs), οι νέες γραμμές (newlines) και ούτω καθεξής, θεωρούνται έγκυροι χαρακτήρες και γίνονται μέρος του δέντρου DOM με τη μορφή κόμβων #text. Στη συνέχεια, η ετικέτα που περιέχει μια νέα γραμμή πριν από το <h1> θα δημιουργήσει ένα #text κόμβο.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για να αποτρέψετε αυτό το πρόβλημα με τους κόμβους firstChild και lastChild που επιστρέφουν τους κόβμους #text ή #comment, μπορείτε να χρησιμοποιήσετε τις ιδιότητες firstElementChild και lastElementChild ως εναλλακτική λύση. Αυτές οι ιδιότητες θα επιστρέψουν μόνο το πρώτο και το τελευταίο στοιχείο του κόμβου αντίστοιχα. Ωστόσο, αυτό δεν θα λειτουργήσει στο Internet Explorer πριν από την έκδοση 9.

Το παρακάτω παράδειγμα θα σας βοηθήσει να το κατανοήσετε καλύτερα:

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var main = document.getElementById("main");
alert(main.firstElementChild.nodeName); // Outputs: H1
main.firstElementChild.style.color = "red";

var hint = document.getElementById("hint");
alert(hint.firstElementChild.nodeName); // Outputs: SPAN
hint.firstElementChild.style.color = "blue";
</script>
```

Εικόνα 114 – *Παράδειγμα πρόσβασης σε θυγατρικούς κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Για να αποκτήσετε πρόσβαση σε όλους τους θυγατρικούς κόμβους ενός δεδομένου στοιχείου, μπορείτε επίσης να χρησιμοποιήσετε την ιδιότητα childNodes. Λάβετε υπόψη ότι στον πρώτο θυγατρικό κόμβο προσαρτάται ο δείκτης 0.

Παρατηρήστε το παράδειγμα που παρατίθεται πιο κάτω:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var main = document.getElementById("main");

// First check that the element has child nodes
if(main.hasChildNodes()) {
    var nodes = main.childNodes;

    // Loop through node list and display node name
    for(var i = 0; i < nodes.length; i++) {
        alert(nodes[i].nodeName);
    }
}
</script>
```

Εικόνα 115 – *Παράδειγμα πρόσβασης στους Θυγατρικούς Κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Εδώ, η ιδιότητα childNodes επιστρέφει *όλους τους θυγατρικούς κόμβους*, συμπεριλαμβανομένων των μη στοιχειακών κόμβων όπως οι κόμβοι κειμένου και σχολίων.

Αν θέλετε να λάβετε *μια συλλογή μόνο από στοιχεία*, θα πρέπει να χρησιμοποιήσετε την ιδιότητα των θυγατρικών κόμβων.

Ας δούμε ένα παράδειγμα για να καταλάβουμε πώς χρησιμοποιείται αυτή ιδιότητα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var main = document.getElementById("main");

// First check that the element has child nodes
if(main.hasChildNodes()) {
    var nodes = main.children;

    // Loop through node list and display node name
    for(var i = 0; i < nodes.length; i++) {
        alert(nodes[i].nodeName);
    }
}
</script>
```

Εικόνα 116 – *Παράδειγμα πρόσβασης στους Θυγατρικούς Κόμβους (**Πηγή**:*
https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)

Πρόσβαση στους Γονικούς Κόμβους

Για να αποκτήσετε πρόσβαση στον γονικό κόμβο ενός συγκεκριμένου κόμβου στο δέντρο Dom, μπορείτε να χρησιμοποιήσετε την ιδιότητα parentNode.

* Σημειώστε ότι η ιδιότητα parentNode θα επιστρέφει πάντα μηδενικές τιμές (null values) για κόμβους εγγράφου επειδή δεν έχουν γονικές σχέσεις.

Στο παρακάτω παράδειγμα, μπορείτε να δείτε πώς χρησιμοποιείται η ιδιότητα parentNode:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var hint = document.getElementById("hint");
alert(hint.parentNode.nodeName); // Outputs: DIV
alert(document.documentElement.parentNode.nodeName); // Outputs: #document
alert(document.parentNode); // Outputs: null
</script>
```

Εικόνα 117 – *Παράδειγμα πρόσβασης σε Γονικούς Κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Είναι καλό να γνωρίζετε ότι οι κόβμοι που βρίσκονται στην κορυφή ενός δέντρου DOM μπορούν να καταστούν προσβάσιμοι απευθείας ως ιδιότητες εγγράφου (document properties). Έχουμε δει κάποια παραδείγματα σχετικά με τους κόμβους που βρίσκονται στην κορυφή ενός δέντρου DOM σε προηγούμενες ενότητες όπως το <html> στοιχείο, το οποίο μπορεί να καταστεί προσβάσιμο με την ιδιότητα document.documentElement. Επίσης, το <head> στοιχείο μπορεί να καταστεί προσβάσιμο με την ιδιότητα document.head και το <body> στοιχείο με την ιδιότητα document.body.

Υπάρχει επίσης η επιλογή να λάβετε μόνο κόμβους στοιχείων με την ιδιότητα parentElement, όπως φαίνεται στο παρακάτω παράδειγμα:

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var hint = document.getElementById("hint");
alert(hint.parentNode.nodeName); // Outputs: DIV
hint.parentNode.style.backgroundColor = "yellow";
</script>
```

Εικόνα 118 – *Παράδειγμα πρόσβασης στους Γονικούς Κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πρόσβαση στους Αδελφούς Κόμβους (ή Sibling Nodes)

Για να αποκτήσετε πρόσβαση στον προηγούμενο και τον επόμενο κόμβο σε ένα δέντρο Dom, μπορείτε να χρησιμοποιήσετε τις προηγούμενες ιδιότητες previousSibling και nextSibling αντίστοιχα.

Ας δούμε ένα παράδειγμα:

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p><hr>
</div>

<script>
var title = document.getElementById("title");
alert(title.previousSibling.nodeName); // Outputs: #text

var hint = document.getElementById("hint");
alert(hint.nextSibling.nodeName); // Outputs: HR
</script>
```

Εικόνα 119 – *Παράδειγμα πρόσβασης στους Αδελφούς Κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Για να παρακάμψετε οποιουσδήποτε κόμβους κενών κειμένων, μπορείτε να χρησιμοποιήσετε τις δηλώσεις previousElementSibling και nextElementSibling ως εναλλακτικές λύσεις για να λάβετε τα προηγούμενα και επόμενα στοιχεία αμφιθαλών κόμβων. Εάν δεν βρεθεί ένας τέτοιος κόμβος αδελφός, αυτές οι ιδιότητες θα επιστρέψουν μηδενικές τιμές.

Ας δούμε το ακόλουθο παράδειγμα παρακάτω:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
<div id="main">
    <h1 id="title">My Heading</h1>
    <p id="hint"><span>This is some text.</span></p>
</div>

<script>
var hint = document.getElementById("hint");
alert(hint.previousElementSibling.nodeName); // Outputs: H1
alert(hint.previousElementSibling.textContent); // Outputs: My Heading

var title = document.getElementById("title");
alert(title.nextElementSibling.nodeName); // Outputs: P
alert(title.nextElementSibling.textContent); // Outputs: This is some text.
</script>
```

Εικόνα 120 – *Παράδειγμα πρόσβασης στους Αδελφούς Κόμβους (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

Η ιδιότητα textContent που χρησιμοποιείται εδώ δηλώνει το περιεχόμενο του κειμένου ενός κόμβου και όλων των απογόνων του.

Τύποι Κόμβων DOM

Το δέντρο DOM αποτελείται από διαφορετικούς τύπους κόμβων που περιλαμβάνει στοιχεία, κείμενο, σχόλια και πολλά άλλα.

Κάθε κόμβος έχει μια ιδιότητα nodeType που μπορεί να σας βοηθήσει να καταλάβετε πώς μπορείτε να έχετε πρόσβαση και να επεξεργάζεστε τον εν λόγω κόμβο. Ο παρακάτω πίνακας παρέχει μια λίστα με τους πιο σημαντικούς και συχνά χρησιμοποιημένους τύπους κόμβων που πρέπει να γνωρίζει κανείς:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Constant	Value	Description
ELEMENT_NODE	1	An element node such as `<p>` or `<img>`.
TEXT_NODE	3	The actual text of element.
COMMENT_NODE	8	A comment node i.e. `<!-- some comment -->`
DOCUMENT_NODE	9	A document node i.e. the parent of `<html>` element.
DOCUMENT_TYPE_NODE	10	A document type node e.g. `<!DOCTYPE html>` for HTML5 documents.

Πίνακας 6 – *Πίνακας των πιο κοινών τύπων κόμβων DOM (**Πηγή**: https://www.tutorialrepublic.com/javascript-tutorial/javascript-dom-selectors.php)*

5.3. JavaScript & BOM

Window JavaScript - Το Μοντέλο Αντικειμένου του Προγράμματος Περιήγησης

Το Browser Object Model (BOM) επιτρέπει στην JavaScript να «μιλήσει» στο πρόγραμμα περιήγησης.

The Browser Object Model (BOM)

Δεν υπάρχουν επίσημα πρότυπα για το Browser Object Model (BOM).

Δεδομένου ότι τα σύγχρονα προγράμματα περιήγησης έχουν εφαρμόσει (σχεδόν) τις ίδιες μεθόδους και ιδιότητες για τη διαδραστικότητα της JavaScript, συχνά αυτά αναφέρονται, ως μέθοδοι και ιδιότητες του BOM.

Το αντικείμενο window (Window Object)

Το αντικείμενο window υποστηρίζεται από όλα τα προγράμματα περιήγησης. Αντιπροσωπεύει το παράθυρο του προγράμματος περιήγησης.
Όλα τα παγκόσμια αντικείμενα JavaScript, οι λειτουργίες και οι μεταβλητές γίνονται αυτόματα μέλη του αντικειμένου window.
Οι καθολικές μεταβλητές είναι ιδιότητες του αντικειμένου window.
Οι καθολικές λειτουργίες είναι μέθοδοι του αντικειμένου window.
Ακόμα και το αντικείμενο εγγράφου (του HTML DOM) είναι μια ιδιότητα του αντικειμένου window:

```
window.document.getElementById("header");
```

Εικόνα 121 *– Το Αντικείμενο Window στο JS BOM (**Πηγή:** https://www.tutorialrepublic.com/javascript-tutorial/javascript-window.php)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

είναι το ίδιο με:

```
document.getElementById("header");
```

***Εικόνα 122** – Το Αντικείμενο Window στο JS BOM – εναλλακτική εκδοχή (**Πηγή:** https://www.tutorialrepublic.com/javascript-tutorial/javascript-window.php)*

Μέγεθος Window

Δύο ιδιότητες μπορούν να χρησιμοποιηθούν για τον προσδιορισμό του μεγέθους του window του προγράμματος περιήγησης.

Και οι δύο ιδιότητες επιστρέφουν τα μεγέθη σε εικονοστοιχεία:

- window.innerHeight - το εσωτερικό ύψος του window του προγράμματος περιήγησης (σε εικονοστοιχεία)
- window.innerWidth - το εσωτερικό πλάτος του παραθύρου του προγράμματος περιήγησης (σε εικονοστοιχεία)

```
let w = window.innerWidth;
let h = window.innerHeight;
```

***Εικόνα 123** – Εναλλακτικές του Μεγέθους Window (**Πηγή:** https://www.tutorialrepublic.com/javascript-tutorial/javascript-window.php)*

Άλλες μέθοδοι window (παραθύρων)

Άλλες μέθοδοι

- window.open() - άνοιγμα νέου window
- window.close() - κλείσιμο του τρέχοντος window
- window.moveTo() - μετακίνηση του τρέχοντος window
- window.resizeTo() - αλλαγή μεγέθους του τρέχοντος window

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

JavaScript Window Screen

Το αντικείμενο window.screen περιέχει πληροφορίες σχετικά με την οθόνη του χρήστη.

Οθόνη Window

Το αντικείμενο window.screen μπορεί να γραφτεί χωρίς το πρόθεμα του window.

Ιδιότητες:

- screen.width
- screen.height
- screen.availWidth
- screen.availHyight
- screen.colorDepth
- screen.pixelDepth

Πλάτος οθόνης window

Η ιδιότητα screen.width επιστρέφει το πλάτος της οθόνης του επισκέπτη σε εικονοστοιχεία.

Example

Display the width of the screen in pixels:

```
document.getElementById("demo").innerHTML =
"Screen Width: " + screen.width;
```

Result will be:

```
Screen Width: 1280
```

Εικόνα 124 *–Η ιδιότητα πλάτους οθόνης screen.width (**Πηγή:** https://www.w3schools.com/js/js_window_screen.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ύψος οθόνης window

Η ιδιότητα screen.height επιστρέφει το ύψος της οθόνης του επισκέπτη σε εικονοστοιχεία.

Example

Display the height of the screen in pixels:

```
document.getElementById("demo").innerHTML =
"Screen Height: " + screen.height;
```

Result will be:

```
Screen Height: 720
```

Εικόνα 125 *– Η ιδιότητα ύψους οθόνης screen.height (**Πηγή:** https://www.w3schools.com/js/js_window_screen.asp)*

Διαθέσιμο πλάτος οθόνης window

Η ιδιότητα screen.availWidth επιστρέφει το πλάτος της οθόνης του επισκέπτη, σε εικονοστοιχεία, πλην των χαρακτηριστικών διεπαφής όπως η γραμμή εργασιών Window (taskbar).

Example

Display the available width of the screen in pixels:

```
document.getElementById("demo").innerHTML =
"Available Screen Width: " + screen.availWidth;
```

Result will be:

```
Available Screen Width: 1280
```

Εικόνα 126 *– Η ιδιότητα διαθέσιμου πλάτους οθόνης window screen.availWidth (**Πηγή:** https://www.w3schools.com/js/js_window_screen.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Διαθέσιμο Ύψος Οθόνης Window

Η ιδιότητα screen.availWidth επιστρέφει το πλάτος της οθόνης του επισκέπτη, σε εικονοστοιχεία, πλην τα χαρακτηριστικά διεπαφής όπως η γραμμή εργασιών Window (taskbar).

Εικόνα 127 *– Η ιδιότητα διαθέσιμου ύψους οθόνης window screen.availWidth (**Πηγή:** https://www.w3schools.com/js/js_window_screen.asp)*

Βάθος χρώματος οθόνης window

Η ιδιότητα screen.colorDepth επιστρέφει τον αριθμό των bit που χρησιμοποιήθηκαν για την εμφάνιση ενός χρώματος.

Όλοι οι σύγχρονοι υπολογιστές χρησιμοποιούν 24 bit ή 32 bit υλικό για την ανάλυση του χρώματος:

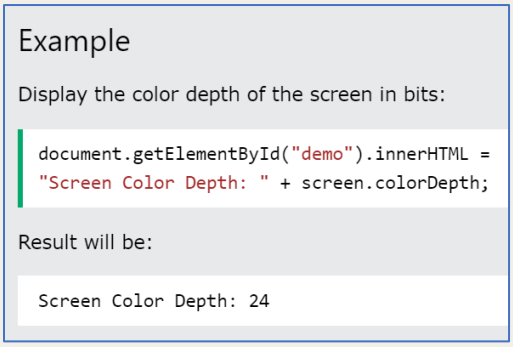
24 bits = 16.777.216 διαφορετικά «Αληθινά χρώματα» (True Colours)

32 bit = 4.294.967.296 διαφορετικά «Βαθιά χρώματα» (Deep Colors)

Οι παλαιότεροι υπολογιστές χρησιμοποιούσαν 16 bits: 65,536 διαφορετικά «Υψηλά Χρώματα» ("High Colors") για ανάλυση.

Οι πολύ παλιοί υπολογιστές, και τα παλιά κινητά τηλέφωνα χρησιμοποιούσαν 8 bits: 256 διαφορετικά "χρώματα VGA".

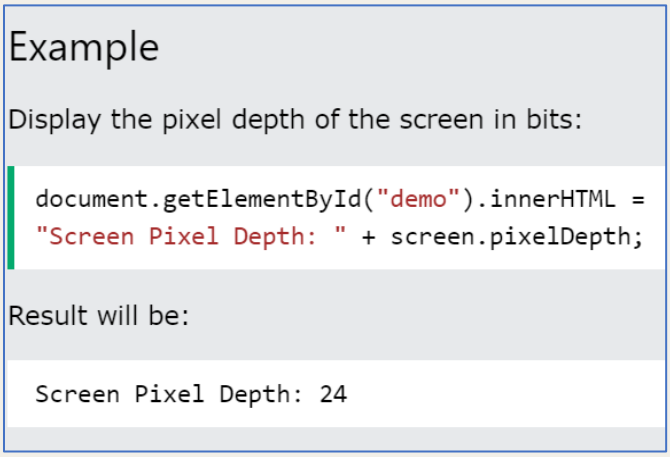
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 128 *– Η ιδιότητα βάθους χρώματος οθόνης window screen.colorDepth (**Πηγή:** https://www.w3schools.com/js/js_window_screen.asp)*

Βάθος εικονοστοιχείου οθόνης window

Η ιδιότητα screen.pixelDepth επιστρέφει το βάθος εικονοστοιχείου της οθόνης.



Εικόνα 129 *– Η ιδιότητα βάθους εικονοστοιχείου οθόνης window screen.pixelDepth (**Πηγή:** https://www.w3schools.com/js/js_window_screen.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τοποθεσία window στην JavaScript (Location)

Το αντικείμενο window.location μπορεί να χρησιμοποιηθεί για τη λήψη της τρέχουσας διεύθυνσης σελίδας (URL) και για την ανακατεύθυνση του προγράμματος περιήγησης σε μια νέα σελίδα.

Τοποθεσία Window

Το αντικείμενο window.screen μπορεί να γραφτεί χωρίς το πρόθεμα του window.

Για παράδειγμα:

- Η ιδιότητα window.location.href επιστρέφει το href (URL) της τρέχουσας σελίδας
- Η ιδιότητα window.location.hostname επιστρέφει το όνομα τομέα (domain name) της φιλοξενίας ιστοσελίδων (web host)
- Η ιδιότητα window.location.pathname επιστρέφει τη διαδρομή (path) και το όνομα αρχείου (filename) της τρέχουσας σελίδας
- Η ιδιότητα window.location.pro protocol επιστρέφει το πρωτόκολλο ιστού που χρησιμοποιήθηκε (http: ή https:)
- Η ιδιότητα window.location.assign() φορτώνει ένα νέο έγγραφο

Τοποθεσία Href Window

Η ιδιότητα window.location.href επιστρέφει τη διεύθυνση URL της τρέχουσας σελίδας.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Display the href (URL) of the current page:

```
document.getElementById("demo").innerHTML =
"Page location is " + window.location.href;
```

Result is:

```
Page location is https://www.w3schools.com/js/js_window_location.asp
```

Εικόνα 130 *– Η ιδιότητα τοποθεσίας href window window.location.href (**Πηγή:** https://www.w3schools.com/js/js_window_location.asp)*

Τοποθεσία Hostname Window

Η ιδιότητα window.location.hostname επιστρέφει το όνομα του internet host (της τρέχουσας σελίδας).

Example

Display the name of the host:

```
document.getElementById("demo").innerHTML =
"Page hostname is " + window.location.hostname;
```

Result is:

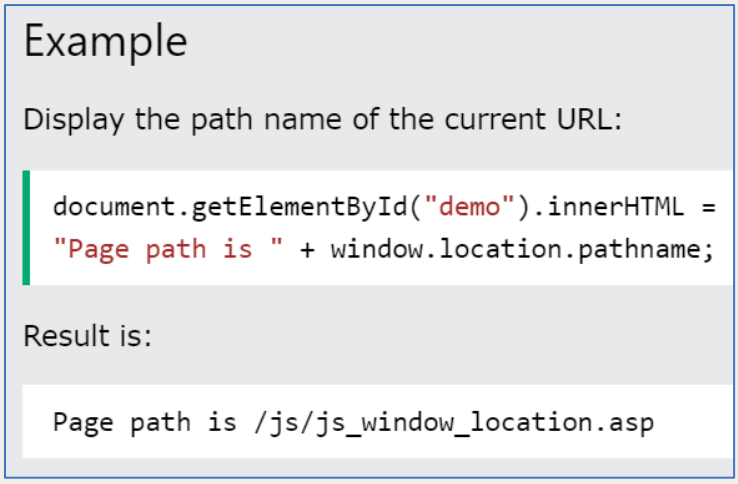
```
Page hostname is www.w3schools.com
```

Εικόνα 131 *– The Window window.location.hostname property (**Πηγή:** https://www.w3schools.com/js/js_window_location.asp)*

Τοποθεσία Pathname Window

Η ιδιότητα window.location.pathname επιστρέφει το pathname της τρέχουσας σελίδας.

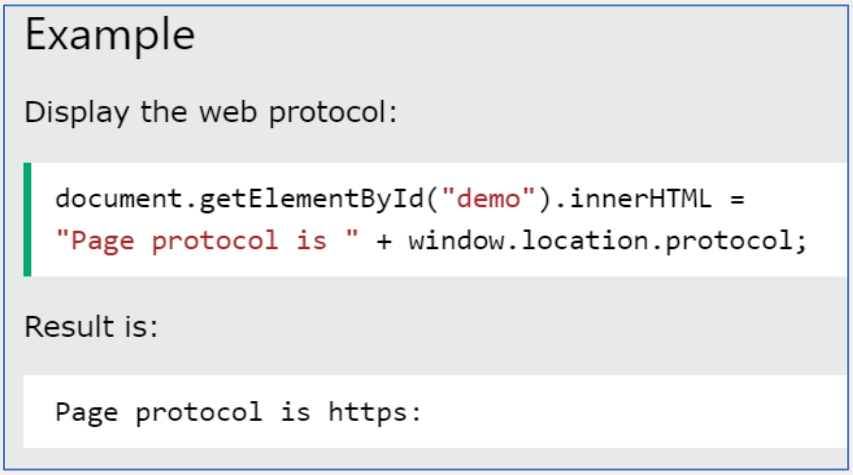
Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)



Εικόνα 132 *– Η ιδιότητα τοποθεσίας pathname window window.location.pathname (**Πηγή:** https://www.w3schools.com/js/js_window_location.asp)*

Τοποθεσίας Πρωτοκόλλου Window

Η ιδιότητα window.location.pro protocol επιστρέφει το πρωτόκολλο ιστού της σελίδας.



Εικόνα 133 *– Η ιδιότητα τοποθεσίας πρωτοκόλλου window window.location.protocol (**Πηγή:** https://www.w3schools.com/js/js_window_location.asp)*

Τοποθεσία Θύρας Window (Port)

Η ιδιότητα window.location.port επιστρέφει τον αριθμό της θύρας φιλοξενίας διαδικτύου (της τρέχουσας σελίδας).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Display the name of the host:

```
document.getElementById("demo").innerHTML =
"Port number is " + window.location.port;
```

Result is:

```
Port number is
```

Εικόνα 134 *– Η ιδιότητα τοποθεσίας θύρας window window.location.port (**Πηγή:** https://www.w3schools.com/js/js_window_location.asp)*

Ανάθεση τοποθεσίας παραθύρου (Window Location Assign)

Η ιδιότητα window.location.assign() φορτώνει ένα νέο έγγραφο.

Example

Load a new document:

```
<html>
<head>
<script>
function newDoc() {
  window.location.assign("https://www.w3schools.com")
}
</script>
</head>
<body>

<input type="button" value="Load new document" onclick="newDoc()">

</body>
</html>
```

Εικόνα 135 *– Η ιδιότητα ανάθεσης τοποθεσίας παραθύρου window.location.assign() (**Πηγή:** https://www.w3schools.com/js/js_window_location.asp)*

Ιστορικό window στην JavaScript

Το αντικείμενο window.history περιέχει το ιστορικό του προγράμματος περιήγησης.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ιστορικό Window

Το αντικείμενο window.history μπορεί να γραφτεί χωρίς το πρόθεμα του window.

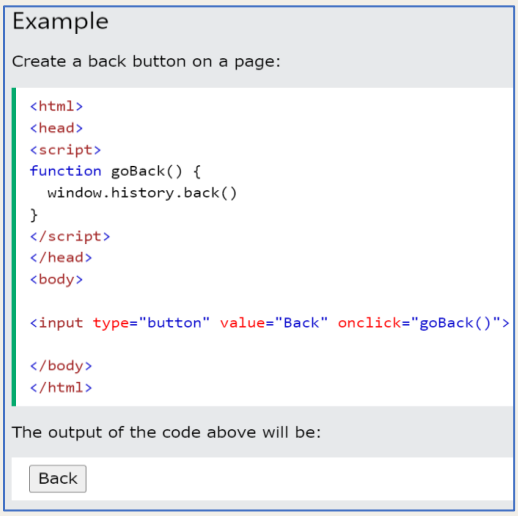
Για την προστασία του απορρήτου των χρηστών, υπάρχουν περιορισμοί στον τρόπο με τον οποίο η JavaScript μπορεί να έχει πρόσβαση σε αυτό το αντικείμενο.

Ορισμένες μέθοδοι:

- history.back() - το ίδιο με το να κάνετε κλικ πίσω στο πρόγραμμα περιήγησης
- history.forward() - το ίδιο με το να κάνετε κλικ προς τα εμπρός στο πρόγραμμα περιήγησης

Πίσω στο Ιστορικό Window (History Back)

Η μέθοδος history.back() φορτώνει την προηγούμενη διεύθυνση URL στη λίστα ιστορικού.
Αυτό είναι το ίδιο με το να κάνετε κλικ στο κουμπί Πίσω στο πρόγραμμα περιήγησης.



***Εικόνα 136** – Η ιδιότητα history.back() (**Πηγή:** https://www.w3schools.com/js/js_window_history.asp)*

Εμπρός στο Ιστορικό Window (History Forward)

Η μέθοδος history.forward() φορτώνει την επόμενη διεύθυνση URL στη λίστα ιστορικού. Αυτό είναι το ίδιο με το να κάνετε κλικ στο κουμπί Forward στο πρόγραμμα περιήγησης.

Example

Create a forward button on a page:

```
<html>
<head>
<script>
function goForward() {
  window.history.forward()
}
</script>
</head>
<body>

<input type="button" value="Forward" onclick="goForward()">

</body>
</html>
```

The output of the code above will be:

Forward

Εικόνα 137 *– Η ιδιότητα history.forward() (**Πηγή:** https://www.w3schools.com/js/js_window_history.asp)*

Πλοηγός window στην JavaScript

Το αντικείμενο window.navigator περιέχει πληροφορίες σχετικά με το πρόγραμμα περιήγησης του επισκέπτη.

Window Navigator

Το αντικείμενο window.screen μπορεί να γραφτεί χωρίς το πρόθεμα του window.

Για παράδειγμα:

- navigator.appName
- navigator.appCodeName
- navigator.platform

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Browser Cookies

The cookieEnabled property returns true if cookies are enabled, otherwise false:

```
document.getElementById("demo").innerHTML =
"cookiesEnabled is " + navigator.cookieEnabled;
```

Εικόνα 138 *– Cookies προγράμματος περιήγησης (**Πηγή**: https://www.w3schools.com/js/js_window_navigator.asp)*

Εφαρμογή προγράμματος περιήγησης (Browser Application Name)

Η ιδιότητα appName επιστρέφει το όνομα της εφαρμογής του προγράμματος περιήγησης.

Όνομα Κώδικα Εφαρμογής Προγράμματος Περιήγησης (Browser Application Code Name)

Η ιδιότητα appCodeName επιστρέφει το όνομα κώδικα της εφαρμογής του προγράμματος περιήγησης.

Η μηχανή του προγράμματος περιήγησης (Browser Engine)

Η ιδιότητα προϊόντος (product) επιστρέφει το όνομα προϊόντος της μηχανής του προγράμματος περιήγησης.

Έκδοση Προγράμματος Περιήγησης (Browser Version)

Η ιδιότητα appVersion επιστρέφει πληροφορίες έκδοσης σχετικά με το πρόγραμμα περιήγησης.

Ο Agent Προγράμματος Περιήγησης (Browser Agent)

Η ιδιότητα userAgent επιστρέφει την κεφαλίδα user-agent που αποστέλλεται από το πρόγραμμα περιήγησης στο διακομιστή.

Προσοχή!!!

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Οι πληροφορίες από το αντικείμενο του πλοηγού μπορεί συχνά να είναι παραπλανητικές και δεν πρέπει να χρησιμοποιούνται για την ανίχνευση εκδόσεων του προγράμματος περιήγησης επειδή:

- Διαφορετικά προγράμματα περιήγησης μπορούν να χρησιμοποιήσουν το ίδιο όνομα
- Τα δεδομένα του πλοηγού μπορούν να αλλάξουν από τον κάτοχο του προγράμματος περιήγησης
- Ορισμένα προγράμματα περιήγησης αυτοπροσδιορίζονται εσφαλμένα για να παρακάμψουν τις δοκιμές ιστότοπου
- Τα προγράμματα περιήγησης δεν μπορούν να αναφέρουν νέα λειτουργικά συστήματα, τα οποία θα κυκλοφορήσουν αργότερα από το πρόγραμμα περιήγησης

Η Πλατφόρμα του Προγράμματος Περιήγησης (Browser Platfrom)

Η ιδιότητα της πλατφόρμας επιστρέφει την πλατφόρμα του προγράμματος περιήγησης (λειτουργικό σύστημα).

Γλώσσα Προγράμματος Περιήγησης (Browser Language)

Η ιδιότητα γλώσσας επιστρέφει τη γλώσσα του προγράμματος περιήγησης.

Είναι το πρόγραμμα περιήγησης σε σύνδεση;

Η ιδιότητα onLine επιστρέφει την τιμή "αληθές" εάν το πρόγραμμα περιήγησης είναι συνδεδεμένο.

Είναι ενεργοποιημένη η Java;

Η μέθοδος javaEnabled () επιστρέφει την τιμή "αληθές" αν είναι ενεργοποιημένη η Java.

Αναδυόμενα πλαίσια JavaScript (Popup Boxes)

Το JavaScript έχει τρία είδη αναδυόμενων πλαισίων: πλαίσιο ειδοποίησης (alert box), πλαίσιο επιβεβαίωσης (confirm box) και πλαίσιο προτροπής (prompt box).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Πλαίσιο Ειδοποίησης (Alert Box)

Ένα πλαίσιο ειδοποίησης χρησιμοποιείται συχνά εάν θέλετε να βεβαιωθείτε ότι οι πληροφορίες φτάνουν στο χρήστη.

Όταν εμφανιστεί ένα πλαίσιο ειδοποίησης, ο χρήστης θα πρέπει να κάνει κλικ στο "OK" για να συνεχίσει.

Syntax:

```
window.alert("sometext");
```

Εικόνα 139 *– Σύνταξη πλαισίου ειδοποίησης (alert) (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Η μέθοδος window.alert() μπορεί να γραφτεί χωρίς το πρόθεμα window.

```
alert("I am an alert box!");
```

Εικόνα 140 *– Η μέθοδος window.alert() (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Επιβεβαίωση πλαισίου (Confirm Box)

A confirm box is often used if you want the user to verify or accept something.

When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.

If the user clicks "OK", the box returns true. If the user clicks "Cancel", the box returns false.

Syntax:

```
window.confirm("sometext");
```

Εικόνα 141 *– Η μέθοδος window.confirm() (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Η μέθοδος window.confirm() μπορεί να γραφτεί χωρίς το πρόθεμα window.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
if (confirm("Press a button!")) {
  txt = "You pressed OK!";
} else {
  txt = "You pressed Cancel!";
}
```

Εικόνα 142 *– Η μέθοδος window.confirm() (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Πλαίσιο Προτροπής (Prompt Box)

Ένα πλαίσιο προτροπής χρησιμοποιείται συχνά εάν θέλετε ο χρήστης να εισάγει μια τιμή πριν από την είσοδο σε μια σελίδα.

Όταν εμφανιστεί ένα πλαίσιο προτροπής, ο χρήστης θα πρέπει να κάνει κλικ είτε στο "OK" είτε στο "Cancel" για να προχωρήσει μετά την εισαγωγή μιας τιμής εισόδου.

Εάν ο χρήστης κάνει κλικ στο "OK", το πλαίσιο επιστρέφει την τιμή εισόδου. Εάν ο χρήστης κάνει κλικ στην επιλογή "Ακύρωση", το πλαίσιο επιστρέφει την τιμή "ψευδές".

Syntax:

```
window.prompt("sometext","defaultText");
```

Εικόνα 143 *– Η σύνταξη πλαισίου προτροπής (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Η μέθοδος window.prompt() μπορεί να γραφτεί χωρίς το πρόθεμα window.

```
let person = prompt("Please enter your name", "Harry Potter");
let text;
if (person == null || person == "") {
  text = "User cancelled the prompt.";
} else {
  text = "Hello " + person + "! How are you today?";
}
```

Εικόνα 144 *– Η μέθοδος window.prompt() (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αλλαγή γραμμής (Line Break)

Για να εμφανιστεί η λειτουργία αλλαγής γραμμής μέσα σε ένα αναδυόμενο πλαίσιο, χρησιμοποιήστε μια κάθετο backlash ακολουθούμενη από το χαρακτήρα n.

```
alert("Hello\nHow are you?");
```

Εικόνα 145 *– Εμφάνιση αλλαγής γραμμής μέσα σε αναδυόμενο πλαίσιο (**Πηγή:** https://www.w3schools.com/js/js_popup.asp)*

Γεγονότα Χρονισμού στην JavaScript (Timing Events)

Γεγονότα Χρονισμού

Το αντικείμενο window επιτρέπει την εκτέλεση του κώδικα σε καθορισμένα χρονικά διαστήματα.

Αυτά τα χρονικά διαστήματα ονομάζονται συμβάντα συγχρονισμού.

Οι δύο βασικές μέθοδοι που χρησιμοποιούνται με το JavaScript είναι:

- setTimeout(funciton,milliseconds):
  Εκτελεί μια συνάρτηση, μετά από αναμονή ενός καθορισμένου αριθμού χιλιοστών του δευτερολέπτου.
- setInterval(function,milliseconds):
  Εκτελεί το ίδιο με το setTimeout(), αλλά επαναλαμβάνει την εκτέλεση της συνάρτησης συνεχώς.

Το setTimeout() και το setInterval() είναι και οι δύο μέθοδοι του αντικειμένου HTML DOM Window.

Η μέθοδος setTimeout()

```
window.setTimeout(function, milliseconds);
```

Εικόνα 146 *– Η μέθοδος setTimeout() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η μέθοδος window.setTimeout() μπορεί να γραφτεί χωρίς το πρόθεμα window.
Η πρώτη παράμετρος είναι μια συνάρτηση που θα εκτελεστεί.

Η δεύτερη παράμετρος δείχνει τον αριθμό των χιλιοστών του δευτερολέπτου πριν από την εκτέλεση.

Example

Click a button. Wait 3 seconds, and the page will alert "Hello":

```
<button onclick="setTimeout(myFunction, 3000)">Try it</button>

<script>
function myFunction() {
  alert('Hello');
}
</script>
```

Εικόνα 147 *– Ορισμός χρονικού ορίου με τη μέθοδο window.setTimeout() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Πώς να σταματήσετε την εκτέλεση;

Η μέθοδος clearTimeout() σταματά την εκτέλεση της λειτουργίας που καθορίζεται στο setTimeout().

```
window.clearTimeout(timeoutVariable)
```

Εικόνα 148 *– Η μέθοδος clearTimeOut() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

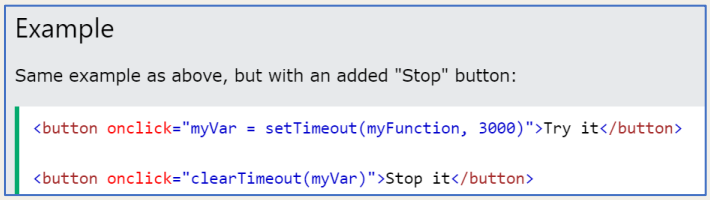
Η μέθοδος window.clearTimeout () μπορεί να γραφτεί χωρίς το πρόθεμα window.
Η μέθοδος clearTimeout() χρησιμοποιεί τη μεταβλητή που επέστρεψε από το setTimeout():

```
myVar = setTimeout(function, milliseconds);
clearTimeout(myVar);
```

Figure 149 *– η μέθοδος window.clearTimeout() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εάν η λειτουργία δεν έχει ήδη εκτελεστεί, μπορείτε να σταματήσετε την εκτέλεση καλώντας τη μέθοδο clearTimeout():



Εικόνα 150 *– Η μέθοδος clearTimeout() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Η μέθοδος setInterval()

Η μέθοδος setInterval() επαναλαμβάνει μια δοσμένη συνάρτηση σε κάθε δεδομένο χρονικό διάστημα.

```
window.setInterval(function, milliseconds);
```

Εικόνα 151 *– Η μέθοδος setInterval() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Η μέθοδος window.alert() μπορεί να γραφτεί χωρίς το πρόθεμα window.

Η πρώτη παράμετρος είναι μια συνάρτηση που θα εκτελεστεί.

Η δεύτερη παράμετρος υποδεικνύει τη διάρκεια του χρονικού διαστήματος μεταξύ κάθε εκτέλεσης.

Αυτό το παράδειγμα εκτελεί μια λειτουργία που ονομάζεται "myTimer" μία φορά κάθε δευτερόλεπτο (όπως ένα ψηφιακό ρολόι).

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Display the current time:

```
setInterval(myTimer, 1000);

function myTimer() {
  const d = new Date();
  document.getElementById("demo").innerHTML = d.toLocaleTimeString();
}
```

Εικόνα 152 *– Η μέθοδος window.setInterval() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Πώς να σταματήσετε την εκτέλεση;

Η μέθοδος clearInterval () σταματά τις εκτελέσεις της συνάρτησης που καθορίζεται στη μέθοδο setInterval ().

```
window.clearInterval(timerVariable)
```

Εικόνα 153 *– Η μέθοδος clearInterval() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Η μέθοδος window.alert() μπορεί να γραφτεί χωρίς το πρόθεμα window.

Η μέθοδος clearInterval() χρησιμοποιεί τη μεταβλητή που επέστρεψε από το setInterval():

```
let myVar = setInterval(function, milliseconds);
clearInterval(myVar);
```

Εικόνα 154 *– Η μεταβλητή που επιστρέφεται από την μέθοδο setInterval() (**Πηγή:** https://www.w3schools.com/js/js_timing.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Example

Same example as above, but we have added a "Stop time" button:

```
<p id="demo"></p>

<button onclick="clearInterval(myVar)">Stop time</button>

<script>
let myVar = setInterval(myTimer, 1000);
function myTimer() {
  const d = new Date();
  document.getElementById("demo").innerHTML = d.toLocaleTimeString();
}
</script>
```

Εικόνα 21 *– Καθαρό διάστημα από την μέθοδο clearinterval() (**Πηγή**: https://www.w3schools.com/js/js_timing.asp)*

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

5.4. JavaScript Advanced

Ημερομηνία και ώρα JavaScript (Date and Time)

Από προεπιλογή, η JavaScript χρησιμοποιεί τη ζώνη ώρας του προγράμματος περιήγησης και θα εμφανίζει μια ημερομηνία ως συμβολοσειρά πλήρους κειμένου. Επομένως, δεν χρειάζεται να την εισάγετε οι ίδιοι, αφού γίνεται από προεπιλογή.

Τα αντικείμενα ημερομηνίας περιέχουν μια σειρά μεθόδων για τη ρύθμιση, τη λήψη και την επεξεργασία των ημερομηνιών και ωρών.

Δημιουργία αντικειμένων ημερομηνίας

Μπορείτε να δημιουργήσετε ένα νέο αντικείμενο ημερομηνίας περνώντας την αναπαράσταση της συμβολοσειράς μιας ημερομηνίας στον κατασκευαστή (constructor) Date() ή περνώντας τα επιμέρους στοιχεία μιας ημερομηνίας ως παραμέτρους στον κατασκευαστή Date() (έτος, μήνας, ημέρα, ώρα, λεπτό, δευτερόλεπτο, χιλιοστό του δευτερολέπτου).

Στα ακόλουθα παραδείγματα μπορείτε να παρατηρήσετε πώς δημιουργούνται τα αντικείμενα ημερομηνίας:

- // Δημιουργία ενός αντικειμένου ημερομηνίας με την τρέχουσα ημερομηνία: const now = new Date(); //
- Δημιουργία ενός αντικειμένου με την τρέχουσα ημερομηνία και ώρα σε τυπική μορφή console.log(now); //
- Δημιουργία ενός αντικειμένου με καθορισμένη ημέρα, ημερομηνία και χρόνο console.log(now.toDateString()); //
- Δημιουργία ενός αντικειμένου με καθορισμένη ώρα, καθορισμένο λεπτό ή δευτερόλεπτο console.log(now.toTimeString());

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Δημιουργία ενός αντικειμένου με καθορισμένη ώρα, καθορισμένο λεπτό ή χιλιοστό του δευτερολέπτου console.log(now.toLocaleString());
- Δημιουργία ενός αντικειμένου με καθορισμένο χρόνο console.log(now.getFullYear());
- Δημιουργία ενός αντικειμένου με καθορισμένο μήνα (από 0-11) console.log(now.getMonth());
- Δημιουργία ενός αντικειμένου με καθορισμένη ημερομηνία (από 1-31) console.log(now.getDate());
  Δημιουργία ενός αντικειμένου με καθορισμένη μέρα της εβδομάδας (από 0-6) console.log(now.getDay());
- Δημιουργία ενός αντικειμένου με καθορισμένη ώρα (από 0-23) console.log(now.getHours());
- Δημιουργία ενός αντικειμένου με καθορισμένο λεπτό (από 0-59) console.log(now.getMinutes());
- Δημιουργία ενός αντικειμένου με καθορισμένο δευτερόλεπτο (από 0-59) console.log(now.getSeconds());
- Δημιουργία ενός αντικειμένου με καθορισμένο χιλιοστό του δευτερολέπτου (από 0-999) console.log(now.getMilliseconds());
- Δημιουργία ενός αντικειμένου με καθορισμένο αριθμό των χιλιοστών του δευτερολέπτου από την αρχή της ημέρας console.log(now.getTime());
- Δημιουργία ενός αντικειμένου με καθορισμένο αριθμό δευτερολέπτων από την 01.01.1970: console.log(now.getTime() / 1000);

Παραδείγματα εξόδων:

```
Thu, 22 Aug 2019 11:37:45 GMT
Thu Aug 22 2019 11:37:45 GMT+0000 (Coordinated Universal Time)
11:37:45 GMT+0000 (Coordinated Universal Time)
11:37:45 GMT+0000 (Coordinated Universal Time) 2019 8 22 4 11 37 45
Today is a Thursday, the 22nd of August, 2019.
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μπορείτε να δείτε ένα πρακτικό παράδειγμα εδώ.

Ο κατασκευαστής Date() μπορεί επίσης να λάβει μια ακέραια τιμή που αντιπροσωπεύει τον αριθμό των χιλιοστών του δευτερολέπτου από την 01.01.1970, ή την αναπαράσταση της συμβολοσειράς μιας ημερομηνίας.

Εάν περάσετε μια ακέραια τιμή που αντιπροσωπεύει τον αριθμό των χιλιοστών του δευτερολέπτου από την 01.01.1970, τότε ο κατασκευαστής θα δημιουργήσει ένα αντικείμενο ημερομηνίας που αντιπροσωπεύει την συγκεκριμένη ημερομηνία και ώρα.

Εάν περάσετε μια αναπαράσταση συμβολοσειράς μιας ημερομηνίας, ο κατασκευαστής θα προσπαθήσει να την αναλύσει για να δημιουργήσει ένα αντικείμενο ημερομηνίας που αναπαριστά την συγκεκριμένη ημερομηνία και ώρα. Αν η συμβολοσειρά δεν μπορεί να αναλυθεί, ο κατασκευαστής θα δημιουργήσει ένα μη έγκυρο αντικείμενο ημερομηνίας.

Δημιουργία αντικειμένων ημερομηνίας με Moment.js

Το Moment.js είναι μια βιβλιοθήκη της JavaScript που διευκολύνει την εργασία με τη δημιουργία ημερομηνιών και ωρών.

Για να χρησιμοποιήσετε το Moment.js, θα πρέπει πρώτα να το εγκαταστήσετε με έναν διαχειριστή πακέτων, όπως npm ή yarn:

```
npm install moment yarn add moment
```

Μετά την εγκατάσταση, μπορείτε να εισάγετε το Moment.js στο αρχείο της JavaScript εισάγοντας τη δήλωση:

```
import moment from 'moment';
```

Στο παρακάτω παράδειγμα χρησιμοποιούμε το Moment.js για να δημιουργήσουμε ένα αντικείμενο ημερομηνίας για την τρέχουσα ημερομηνία και ώρα και στη συνέχεια χρησιμοποιούμε τη μέθοδο format() για να εξάγουμε την ημερομηνία και την ώρα σε διαφορετικές μορφές:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
import moment from 'moment'; // Δημιουργία ενός αντικειμένου ημερομηνίας με την τρέχουσα ημερομηνία και ώρα const now = moment(); // Δημιουργία ενός αντικειμένου με την τρέχουσα ημερομηνία και ώρα σε τυπική μορφή console.log(now.format()); // Δημιουργία ενός αντικειμένου με καθορισμένη ημέρα, ημερομηνία και χρόνο console.log[now.format('dddd, MMMM D, YYYY')); // Δημιουργία ενός αντικειμένου με καθορισμένη ώρα, καθορισμένο λεπτό ή δευτερόλεπτο console.log(now.format('h:mm:ss a')); // Δημιουργία ενός αντικειμένου με καθορισμένη ημέρα, ημερομηνία, καθορισμένο μήνα, χρόνο, λεπτό και δευτερόλεπτο console.log[now.format('ddd, hA')] ·
```

Εξόδοι:

```
2019-08-22T15:25:33+04:00 Thursday, August 22, 2019 3:25:33 pm Thu, 3PM
```

Αριθμητικές Ημερομηνίες

Τα αντικείμενα ημερομηνίας μπορούν να επεξεργασθούν για να αλλάξετε ημερομηνίες και ώρες. Τα αντικείμενα ημερομηνίας παρέχουν μια μέθοδο set() που μπορεί να χρησιμοποιηθεί για τον ορισμό του έτους, του μήνα, της ημέρας, της ώρας, του λεπτού, του δευτερολέπτου και του χιλιοστού του δευτερολέπτου μιας ημερομηνίας.

Όταν χρησιμοποιείτε τη μέθοδο set(), το αντικείμενο ενημερώνεται, και έτσι δεν χρειάζεται να δημιουργήσετε ένα νέο αντικείμενο ημερομηνίας.

Το ακόλουθο παράδειγμα χρησιμοποιεί τη μέθοδο set() για να ορίσει την ώρα και το λεπτό μιας ημερομηνίας:

```
const now = new Date(); // Δημιουργία ενός αντικειμένου με καθορισμένη ώρα, καθορισμένο λεπτό και δευτερόλεπτο console.log (now.toLocaleString()); // Αλλάζει την ώρα σε 12 και το λεπτό σε 45 now.setHours(12); now.setMinutes(45); console.log(now.toLocaleString());
```

Έξοδοι:

```
11:56:09 AM 12:45:09 PM
```

Η μέθοδος setHours() μπορεί επίσης να χρησιμοποιηθεί για να ρυθμίσετε την ώρα και το λεπτό, και η μέθοδος setMinutes() μπορεί επίσης να χρησιμοποιηθεί για να ρυθμίσετε το λεπτό και το δευτερόλεπτο.

Ένα αντικείμενο ημερομηνίας διαθέτει επίσης getters για να λαμβάνει το έτος, μήνα, ημέρα, ώρα, λεπτό, δευτερόλεπτο και χιλιοστό του δευτερολέπτου μιας ημερομηνίας.

Το ακόλουθο παράδειγμα χρησιμοποιεί τη μέθοδο set() για να ορίσει την ώρα και το λεπτό μιας ημερομηνίας:

```
const now = new Date(); // Δημιουργία ενός αντικειμένου με καθορισμένη ώρα, καθορισμένο λεπτό και δευτερόλεπτο, console.log (now.toLocaleString ()); // Δημιουργία ενός αντικειμένου με καθορισμένη ώρα και λεπτό console.log (now.getHours() + ':' + now.getMinutes());
```

Έξοδοι:

```
11:56:09 AM 11:56
```

Όταν χρησιμοποιείτε τη μέθοδο set() , το αντικείμενο ενημερώνεται, και έτσι δεν χρειάζεται να δημιουργήσετε ένα νέο αντικείμενο ημερομηνίας.

Η μέθοδος setDate() μπορεί να χρησιμοποιηθεί για τον ορισμό της ημέρας μιας ημερομηνίας. Η μέθοδος setMonth() μπορεί να χρησιμοποιηθεί για τον ορισμό του μήνα μιας ημερομηνίας. Η μέθοδος setFullYear() μπορεί να χρησιμοποιηθεί για τον ορισμό του έτους μιας ημερομηνίας.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Η μέθοδος getDate() μπορεί να χρησιμοποιηθεί για να λάβει την ημέρα μιας ημερομηνίας. Η μέθοδος getMonth() μπορεί να χρησιμοποιηθεί για να λάβει το μήνα μιας ημερομηνίας. Το GetFullYear() μπορεί να χρησιμοποιηθεί για να λάβει το έτος μιας ημερομηνίας.

Το ακόλουθο παράδειγμα χρησιμοποιεί τις μεθόδους set() , setDate () , setMonth () , setFullYear () , get() , getDate () , getMonth () και getFullYear () για να ρυθμίσει και να λάβει το έτος, τον μήνα και την ημέρα μιας ημερομηνίας:

```
const now = new Date(); now.setDate (1); // Ορίζει την ημέρα στην πρώτη ημέρα του μήνα now.setMonth (0); // Ορίζει το μήνα στον Ιανουάριο now.setFullYear (2019); // Ορίζει το έτος στο 2019 (now.toLocaleString()); now.setDate(now.getDate() + 10); // Adds 10
```

Μαθηματικοί Τελεστές JavaScript

Με την JavaScript μπορείτε να εκτελέσετε μαθηματικές λειτουργίες. Αυτό μπορεί να γίνει με τη βοήθεια των τελεστών + (πρόσθεση), - (αφαίρεση), * (πολλαπλασιασμός), / (διαίρεση) και % (όρισμα).

Στο παρακάτω παράδειγμα, χρησιμοποιούμε τον τελεστή πρόσθεσης για να προσθέσουμε δύο αριθμούς:

```
var x = 10; var y = 5; var z = x + y;
```

Αυτό θα δώσει την τιμή του z ως 15.

Με τον ίδιο τρόπο, μπορούμε να χρησιμοποιήσουμε τους τελεστές - (αφαίρεση), * (πολλαπλασιασμός), / (διαίρεση) και % (όρισμα) για να εκτελέσουμε τις αντίστοιχες λειτουργίες τους.

Ο τελεστής ορίσματος επιστρέφει το υπόλοιπο μιας πράξης διαίρεσης. Για παράδειγμα, αν διαιρέσουμε το 10 με το 3, το υπόλοιπο θα είναι 1. Επομένως, το 10 % 3 θα δώσει το αποτέλεσμα 1.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εκτός από αυτές τις βασικές αριθμητικές πράξεις, το JavaScript παρέχει επίσης μερικές ενσωματωμένες μαθηματικές λειτουργίες. Αυτές οι λειτουργίες μπορούν να χρησιμοποιηθούν για την εκτέλεση πιο σύνθετων λειτουργιών.

Μερικές από τις πλέον κοινόχρηστες μαθηματικές λειτουργίες είναι:

- Math.abs(x) : Αυτή η συνάρτηση επιστρέφει την απόλυτη τιμή ενός αριθμού. Για παράδειγμα, η συνάρτηση Math.abs (-10) θα επιστρέψει 10.
- Math.ceil(x) : Αυτή η συνάρτηση στρογγυλοποιεί έναν αριθμό στην πλησιέστερη ακέραια τιμή. Για παράδειγμα, η συνάρτηση Math.ceil(4.7) θα επιστρέψει 5.
- Math.floor(x) : Αυτή η συνάρτηση στρογγυλοποιεί έναν αριθμό στην πλησιέστερη ακέραια τιμή. Για παράδειγμα, η συνάρτηση Math.floor(4.7) θα επιστρέψει 4.
- Math.max(x, y, ...) : Αυτή η συνάρτηση επιστρέφει τον μέγιστο αριθμό των ορισμάτων που πέρασαν σε αυτήν. Για παράδειγμα, η συνάρτηση Math.max(10, 5, 20) θα επιστρέψει 20.
- Math.min(x, y, ...) : Αυτή η συνάρτηση επιστρέφει τον ελάχιστο αριθμό ορισμάτων που πέρασαν σε αυτήν. Για παράδειγμα, η συνάρτηση Math.min (10, 5, 20) θα επιστρέψει 5.
- Math.pow(x, y) : Αυτή η συνάρτηση επιστρέφει την τιμή του x ανυψωμένη στη δύναμη του y. Για παράδειγμα, η συνάρτηση Math.pow (2, 3) θα επιστρέψει 8.
- Math.random() : Αυτή η συνάρτηση επιστρέφει έναν τυχαίο αριθμό μεταξύ 0 και 1.
- Math.sqrt(x) : Αυτή η συνάρτηση επιστρέφει την τετραγωνική ρίζα του x. Για παράδειγμα, η συνάρτηση Math.sqrt (16) θα επιστρέψει 4.

Μπορείτε να μάθετε περισσότερα για τα μαθηματικά αντικείμενα, τις ιδιότητες και τις μεθόδους τους κάνοντας αναφορά στις Μαθηματικές Λειτουργίες της JavaScript.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μετατροπές τύπου JavaScript

Μπορείτε να μετατρέψετε μια τιμή σε έναν συγκεκριμένο τύπο δεδομένων χρησιμοποιώντας ενσωματωμένες μεθόδους, όπως το parseInt() για τη μετατροπή μιας τιμής σε έναν ακέραιο αριθμό, ή το parseFloat() για τη μετατροπή μιας τιμής σε έναν αριθμό κινητής υποδιαστολής (float).

Μπορείτε επίσης να χρησιμοποιήσετε τη μέθοδο Number() για να μετατρέψετε μια τιμή σε έναν αριθμό, ή τη μέθοδο Boolean() για να μετατρέψετε μια τιμή σε Αληθή ή Ψευδή (True ή False).

Στο ακόλουθο παράδειγμα θα μετατρέψουμε μια συμβολοσειρά σε έναν αριθμό χρησιμοποιώντας τη μέθοδο Number():

```
var x = "100"; var y = Number(x); console.log(y); // 100
```

Σε αυτό το παράδειγμα έχουμε μια συμβολοσειρά με την τιμή "100". Μετατρέπουμε αυτή τη συμβολοσειρά σε έναν αριθμό χρησιμοποιώντας τη μέθοδο Number() και στη συνέχεια εκτυπώνουμε την τιμή του αριθμού στην κονσόλα.

Μπορείτε επίσης να χρησιμοποιήσετε τον τελεστή unary + για να μετατρέψετε μια τιμή σε έναν αριθμό. Για παράδειγμα:

```
var x = "100"; var y = +x; console.log(y); // 100
```

Σε αυτό το παράδειγμα έχουμε μια συμβολοσειρά με την τιμή "100". Χρησιμοποιούμε τον τελεστή unary + για να μετατρέψουμε τη συμβολοσειρά σε έναν αριθμό και στη συνέχεια εκτυπώνουμε την τιμή του αριθμού στην κονσόλα.

Μπορείτε να χρησιμοποιήσετε τη μέθοδο parseInt() για να μετατρέψετε μια

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

συμβολοσειρά σε έναν ακέραιο αριθμό ή τη μέθοδο parseFloat() για να μετατρέψετε μια συμβολοσειρά σε έναν αριθμό κινητής υποδιαστολής (float).

Για παράδειγμα:

```
var x = "100.50"; var y = parseInt(x); console.log(y); // 100 var z = parseFloat(x); console.log(z); // 100.5
```

Σε αυτό το παράδειγμα έχουμε μια συμβολοσειρά με την τιμή "100". Χρησιμοποιούμε τη μέθοδο parseInt() για να μετατρέψουμε τη συμβολοσειρά σε έναν ακέραιο αριθμό και στη συνέχεια εκτυπώνουμε την τιμή του ακέραιου στην κονσόλα.

Χρησιμοποιούμε επίσης τη μέθοδο parseFloat () για να μετατρέψουμε τη συμβολοσειρά σε έναν αριθμό κινητής υποδιαστολής (float) και στη συνέχεια εκτυπώνουμε την τιμή του αριθμού κινητής υποδιαστολής (float) στην κονσόλα.

Μπορείτε να χρησιμοποιήσετε τη μέθοδο Boolean() για να μετατρέψετε μια τιμή σε Αληθή ή Ψευδή.

Για παράδειγμα:

```
var x = "100"; var y = Boolean(x); console.log(y); // true
```

Σε αυτό το παράδειγμα έχουμε μια συμβολοσειρά με την τιμή "100". Μετατρέπουμε τη συμβολοσειρά σε Αληθή ή Ψευδή χρησιμοποιώντας τη μέθοδο Boolean() και στη συνέχεια εκτυπώνουμε την τιμή στην κονσόλα.

Μπορείτε να χρησιμοποιήσετε τον τελεστή !! για να μετατρέψετε μια τιμή σε Αληθή ή Ψευδή . Για παράδειγμα:

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
var x = "100"; var y = !!x; console.log(y); // true
```

Σε αυτό το έχουμε μια συμβολοσειρά με την τιμή "100". Θα χρησιμοποιήσουμε τον τελεστή !! για να μετατρέψουμε τη συμβολοσειρά σε Αληθή ή Ψευδή και στη συνέχεια εκτυπώνουμε την τιμή της στην κονσόλα.

Αν θέλετε να μετατρέψετε μια τιμή σε μια συμβολοσειρά, μπορείτε να χρησιμοποιήσετε τη μέθοδο toString().

Για παράδειγμα:

```
var x = "100"; var y = Number(x); console.log(y); // 100
```

Σε αυτό το παράδειγμα έχουμε έναν αριθμό με την τιμή 100. Μετατρέπουμε τον αριθμό σε μια συμβολοσειρά χρησιμοποιώντας τη μέθοδο toString() και στη συνέχεια εκτυπώνουμε τη συμβολοσειρά στην κονσόλα.

Αν θέλετε να μετατρέψετε μια τιμή σε έναν πίνακα, μπορείτε να χρησιμοποιήσετε τη μέθοδο Array.from().

Για παράδειγμα:

```
var x = "100"; var y = Array.from(x); console.log(y); // [ "1", "0",
"0" ]
```

Στο παραπάνω παράδειγμα έχουμε μια συμβολοσειρά με την τιμή "100". Μετατρέπουμε τη συμβολοσειρά σε έναν πίνακα (array) χρησιμοποιώντας τη μέθοδο Array.from() και στη συνέχεια εκτυπώνουμε τον πίνακα στην κονσόλα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αν θέλετε να μετατρέψετε μια τιμή σε ένα αντικείμενο, μπορείτε να χρησιμοποιήσετε τη μέθοδο Object().

Για παράδειγμα:

```
var x = "100"; var y = Object(x); console.log(y); // { "0": "1", "1": "0", "2": "0" }
```

Σε αυτό το παράδειγμα έχουμε μια συμβολοσειρά με την τιμή "100". Μετατρέπουμε τη συμβολοσειρά σε αντικείμενο χρησιμοποιώντας τη μέθοδο Object() και στη συνέχεια εκτυπώνουμε το αντικείμενο στην κονσόλα.

Ακροατές Γεγονότος JavaScript (Event Listener)

Ακροατές Γεγονότος DOM

Ένας ακροατής γεγονότος σας επιτρέπει να ορίσετε τις συναρτήσεις που θα εκτελούνται όταν ένα συγκεκριμένο συμβάν εμφανίζεται σε ένα στοιχείο στο DOM.

Υπάρχει ένας αριθμός διαφορετικών συμβάντων που μπορούν να ενεργοποιηθούν σε ένα στοιχείο του DOM, όπως όταν ένας χρήστης κάνει κλικ σε ένα κουμπί, ή όταν μετακινεί το ποντίκι από πάνω του ή ακόμα όταν τα περιεχόμενα ενός στοιχείου αλλάζουν.

Για να προσθέσετε έναν ακροατή γεγονότων σε ένα στοιχείο, πρέπει πρώτα να επιλέξετε το στοιχείο χρησιμοποιώντας μία από τις μεθόδους επιλογής DOM, όπως document.querySelector() ή document.getElementById() .

Μόλις επιλεγεί το στοιχείο, μπορείτε να χρησιμοποιήσετε τη μέθοδο addEventListener() για να προσαρτήσετε έναν ακροατή γεγονότων στο στοιχείο.

Η μέθοδος addEventListener() λαμβάνει δύο παραμέτρους: τον τύπο του συμβάντος και την συνάρτηση που θέλουμε να καλέσουμε όταν εμφανίζεται το συμβάν.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για παράδειγμα, για να προσθέσετε έναν ακροατή γεγονότων “κλικ” σε ένα στοιχείο κουμπιού, θα χρησιμοποιήσετε τον ακόλουθο κώδικα:

```
button . addEventListener ( "click" , function ( ) { console . log ( "The button was clicked!" ) ; } ) ;
```

Στον παραπάνω κώδικα, όταν κάνετε κλικ στο στοιχείο κουμπιού, θα εκτελεστεί η συνάρτηση που έχει περάσει στη μέθοδο addEventListener().

Μπορείτε επίσης να περάσετε μια συνάρτηση με το όνομα της στη μέθοδο addEventListener(), αντί να χρησιμοποιήσετε μια ανώνυμη συνάρτηση:

```
function handleClick ( ) { console . log ( "The button was clicked!" ) ; } button . addEventListener ( "click" , handleClick ) ;
```

Τύποι Γεγονότων DOM

Υπάρχει ένας αριθμός διαφορετικών συμβάντων που μπορούν να εμφανιστούν σε ένα στοιχείο Dom.

Τα πιο κοινά συμβάντα είναι:

- κλικ
- Μετακίνηση ποντικιού (mouseover)
- mouseout
- πάτημα πλήκτρων (keypress)
- αλλαγή (change)
- υποβολή (submit)

Μπορείτε να βρείτε την ολοκληρωμένη λίστα συμβάντων του DOM στο Mozilla Developer Network.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αντικείμενο συμβάντος DOM (Event Object)

Όταν καλείται μια συνάρτηση ακροατή γεγονότων (event listener), τότε περνάει ένα αντικείμενο συμβάντος ως όρισμα.

Το αντικείμενο συμβάντος περιέχει πληροφορίες σχετικά με το συμβάν που εμφανίστηκε, όπως το είδος του συμβάντος, το στοιχείο στο οποίο εμφανίστηκε το συμβάν και τυχόν δεδομένα σχετικά με το συμβάν.

Για παράδειγμα, το αντικείμενο συμβάντος κλικ περιέχει πληροφορίες σχετικά με το συμβάν κλικ, όπως οι συντεταγμένες x και y του κλικ.

Για να αποκτήσετε πρόσβαση σε ένα αντικείμενο συμβάντος μέσα από μια συνάρτηση ακροατή γεγονότων, μπορείτε να χρησιμοποιήσετε μια συγκεκριμένη λέξη-κλειδί.

Για παράδειγμα, για να καταχωρήσετε τις συντεταγμένες x και y ενός συμβάντος κλικ, θα χρησιμοποιήσετε τον ακόλουθο κώδικα:

```
button . addEventListener ( "click" , function ( ) { console . log ( "x: " + this . clientX + ", y: " + this . clientY ) ; } ) ;
```

Στον παραπάνω κώδικα, η λέξη-κλειδί «this» αναφέρεται στο αντικείμενο του συμβάντος και οι ιδιότητες clientX και clientY χρησιμοποιούνται για πρόσβαση στις συντεταγμένες x και y του συμβάντος κλικ.

DOM Event Delegation

Το DOM Event Delegation είναι μια τεχνική για την προσάρτηση ακροατή γεγονότων σε στοιχεία που δεν υπάρχουν ακόμα σε ένα έγγραφο DOM.

Αυτό είναι χρήσιμο όταν έχετε έναν μεγάλο αριθμό στοιχείων στα οποία θέλετε να προσθέσετε ακροατές γεγονότων, αλλά δεν θέλετε να προσθέσετε έναν ακροατή γεγονότων σε κάθε στοιχείο ξεχωριστά.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Για παράδειγμα, αν έχετε μια λίστα με 100 στοιχεία και θέλετε να προσθέσετε έναν ακροατή γεγονότων κλικ σε κάθε στοιχείο, μπορείτε να χρησιμοποιήσετε τη λειτουργία Event Delegation για να προσθέσετε έναν ακροατή γεγονότων ενός κλικ στο γονικό στοιχείο της λίστας, και να ρυθμίσετε αυτή τη λειτουργία ακροατή για να χειριστεί τα συμβάντα κλικ για όλα τα θυγατρικά στοιχεία.

Για να χρησιμοποιήσετε την λειτουργία αυτή, πρέπει πρώτα να επιλέξετε το γονικό στοιχείο των στοιχείων στα οποία θέλετε να προσθέσετε ακροατές γεγονότων.

Στη συνέχεια, μπορείτε να χρησιμοποιήσετε τη μέθοδο addEventListener() για να συνδέσετε έναν ακροατή γεγονότων στο γονικό στοιχείο.

Η μέθοδος addEventListener() λαμβάνει δύο παραμέτρους: τον τύπο του συμβάντος και μια συνάρτηση για εκτέλεση όταν εμφανίζεται το συμβάν.

Η λειτουργία που περνά στη μέθοδο addEventListener() θα εκτελείται κάθε φορά που το συμβάν εμφανίζεται σε οποιοδήποτε από τα θυγατρικά στοιχεία του γονικού στοιχείου.

Για παράδειγμα, για να προσθέσετε έναν ακροατή γεγονότων κλικ σε όλα τα στοιχεία μιας λίστας, θα χρησιμοποιήσετε τον ακόλουθο κώδικα:

```
var listItems = document . querySelectorAll ( "li" ) ; listItems . forEach ( function ( listItem ) { listItem . addEventListener ( "click" , function ( ) { console . log ( "The list item was clicked!" ) ; } ) ; } ) ;
```

Στον παραπάνω κώδικα, ένας ακροατής γεγονότων κλικ προστίθεται σε κάθε στοιχείο λίστας ξεχωριστά.

Αυτό μπορεί να είναι αναποτελεσματικό εάν υπάρχει μεγάλος αριθμός στοιχείων λίστας.

Επομένως, αντί να προσθέσετε ένα ακροατή γεγονότων σε κάθε στοιχείο ξεχωριστά, μπορείτε αντ' αυτού να χρησιμοποιήσετε την λειτουργία event delegation για να

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

προσθέσετε έναν ακροατή γεγονότων ενός κλικ στο γονικό στοιχείο των στοιχείων λίστας:

```
var list = document . querySelector ( "ul" ) ; list . addEventListener ( "click" , function ( event ) { if ( event . target . tagName === "LI" ) { console . log ( "The list item was clicked!" ) ; } } ) ;
```

Στον παραπάνω κώδικα, ένας ακροατής γεγονότων κλικ προστίθεται στο γονικό στοιχείο των στοιχείων της λίστας.

Η συνάρτηση που πέρασε στη μέθοδο addEventListener() ελέγχει την ιδιότητα tagName του στοιχείου που έκανε κλικ για να αναγνωρίζει αν είναι <li> στοιχείο.

Αν είναι <li> στοιχείο, τότε η συνάρτηση καταγράφει ένα μήνυμα στην κονσόλα.

Αυτή η τεχνική μπορεί να χρησιμοποιηθεί για να προσθέσει ακροατές γεγονότων σε οποιοδήποτε είδος στοιχείου, όχι μόνο σε στοιχεία λίστας.

Για παράδειγμα, μπορείτε να χρησιμοποιήσετε την λειτουργία event delegation για να προσθέσετε έναν ακροατή γεγονότων κλικ σε όλα τα κουμπιά σε ένα έγγραφο:

```
var buttons = document . querySelectorAll ( "button" ) ; buttons . forEach ( function ( button ) { button . addEventListener ( "click" , function ( ) { console . log ( "The button was clicked!" ) ; } ) ; } ) ;
```

Εναλλακτικά, μπορείτε να χρησιμοποιήσετε την λειτουργία event delegation για να προσθέσετε έναν ακροατή συμβάντων κλικ σε όλους τους συνδέσμους σε ένα έγγραφο:

```
var links = document . querySelectorAll ( "a" ) ; links . forEach ( function ( link ) { link . addEventListener ( "click" , function ( ) { console . log ( "The link was clicked!" ) ; } ) ; } ) ;
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Συνοπτική Παρουσίαση της Λειτουργίας Event Delegation

Στο μάθημα αυτό, μάθατε για τους ακροατές γεγονότων που μπορείτε να προσθέσετε σε οποιοδήποτε αντικείμενο DOM στην JavaScript.

Μάθατε πώς να χρησιμοποιείτε τη μέθοδο addEventListener() για να προσθέτετε ακροατές γεγονότων σε στοιχεία του DOM.

Επίσης, μάθατε για τα διάφορους τύπους συμβάντων που μπορούν να εμφανιστούν σε ένα στοιχείο DOM και πώς να αποκτάτε πρόσβαση στο αντικείμενο συμβάντος μέσα από μια συνάρτηση ακροατή γεγονότων.

Τέλος, μάθατε για την λειτουργία event delegation DOM και πώς να την χρησιμοποιείτε για να προσθέτετε ακροατές γεγονότων σε στοιχεία που δεν υπάρχουν ακόμα στο DOM.

Διάδοση γεγονότων JavaScript (Event Propagation)

Όταν συμβαίνει ένα γεγονός, αυτό μπορεί να αναγνωριστεί από την JavaScript και να ενεργοποιηθεί.

Για παράδειγμα, αν κάνετε κλικ σε ένα κουμπί σε μια ιστοσελίδα, μπορείτε να προγραμματίσετε την JavaScript με τέτοιο τρόπο ώστε να αναγνωρίζει το συμβάν για να προβαίνει σε συγκεκριμένες ενέργειες.

Το συμβάν που εμφανίζεται όταν ένας χρήστης κάνει κλικ σε ένα αντικείμενο μιας ιστοσελίδας ονομάζεται συμβάν κλικ.

Όταν κάνετε κλικ σε ένα στοιχείο, το συμβάν κλικ ενεργοποιείται στο στοιχείο στο οποίο πάτησε ο χρήστης.

Το συμβάν στη συνέχεια διαδίδεται στο δέντρο DOM.

Αυτό σημαίνει ότι εάν υπάρχει ένας διαχειριστής γεγονότων σε ένα γονικό στοιχείο, τότε αυτός θα ενεργοποιηθεί.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Εάν υπάρχει ένας διαχειριστής γεγονότων σε ένα στοιχείο grandparent, τότε αυτός θα ενεργοποιηθεί.

Αυτό διαδίδεται στο δέντρο DOM μέχρι το γεγονός να φτάσει στο στοιχείο ρίζας (root element) ή μέχρι το γεγονός να τελειώσει.

Όταν το γεγονός φτάνει στο στοιχείο ρίζας, σημαίνει ότι το γεγονός έχει αναδυθεί (bubble) σε όλο το δέντρο DOM.

Στη συνέχεια, το γεγονός θα διαδοθεί ξανά με τον ίδιο τρόπο προς τα κάτω του δέντρου DOM, δηλαδή από το στοιχείο ρίζας μέχρι το στοιχείο στον οποίο έγινε κλικ.

Αυτό ονομάζεται ανάδυση (bubbling)

Το γεγονός διαδίδεται από την κορυφή DOM μέχρι το ριζικό στοιχείο, και στη συνέχεια πάλι το ίδιο.

Μπορείτε να διαχειριστείτε τα γεγονότα στο δέντρο DOM οποιουδήποτε στοιχείου.

Μπορείτε επίσης να σταματήσετε τη διάδοση ενός γεγονότος.

Για παράδειγμα, μπορεί να έχετε έναν διαχειριστή γεγονότων σε ένα γονικό στοιχείο ο οποίος δεν θέλετε να εκτελείτε όταν ένα γεγονός εμφανίζεται σε ένα θυγατρικό στοιχείο.

Στην περίπτωση αυτή, μπορείτε να σταματήσετε τη διάδοση του γεγονότος προς την κορυφή του δέντρου DOM.

Αυτό λοιπόν ονομάζεται διάδοση γεγονότος (event propagation).

Το γεγονός διαδίδεται στο δέντρο DOM με ανοδική πορεία, αλλά σταματά στο γονικό στοιχείο.

Το γεγονός δεν μπορεί να φτάσει στο στοιχείο ρίζας και δεν διαδίδεται πίσω και προς τα κάτω στο δέντρο DOM.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Μπορείτε επίσης να σταματήσετε τη διάδοση ενός γεγονότος σε οποιοδήποτε σημείο του δέντρου DOM.

Για παράδειγμα, μπορεί να έχετε έναν διαχειριστή γεγονότων σε ένα γονικό στοιχείο ο οποίος δεν θέλετε να εκτελείτε όταν ένα γεγονός εμφανίζεται σε ένα θυγατρικό στοιχείο.

Στην περίπτωση αυτή, μπορείτε να σταματήσετε τη διάδοση του γεγονότος προς τα κάτω του δέντρου DOM.

Αυτό ονομάζεται καταγραφή γεγονότων (event capturing)

Το γεγονός διαδίδεται με ανοδική πορεία στο δέντρο DOM, και στη συνέχεια πίσω ξανά προς τα κάτω.

Ωστόσο, το γεγονός σταματά στο στοιχείο grandparent.

Το γεγονός δεν φτάνει στο στοιχείο grandchild.

Μπορείτε επίσης να σταματήσετε τη διάδοση ενός γεγονότος σε οποιοδήποτε σημείο του δέντρου DOM.

Όταν ένα γεγονός συμβαίνει σε ένα στοιχείο, το γεγονός αυτό μπορεί να αναγνωριστεί από την JavaScript και να ενεργοποιηθεί.

Το γεγονός διαδίδεται με ανοδική πορεία στο δέντρο DOM, και στη συνέχεια πίσω ξανά προς τα κάτω.

Μπορείτε να διαχειριστείτε τα γεγονότα σε οποιοδήποτε στοιχείο στο δέντρο DOM.

Μπορείτε επίσης να σταματήσετε τη διάδοση ενός γεγονότος.

Μέθοδοι Δανεισμού JavaScript (Borrowing Methods)

Θα χρησιμοποιήσουμε ένα παράδειγμα ενός αντικειμένου που περιέχει ένα ιδιότητα την οποία πρέπει να δανειστούμε.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
var myObject = {
someProperty: “foo”,
};
```

Μπορούμε να δανειστούμε την ιδιότητα someProperty από το myObject ως εξής:

```
var myProperty = myObject.someProperty;
```

Το myProperty θα περιέχει τώρα την τιμή "foo".

Μπορούμε επίσης να χρησιμοποιήσουμε την ίδια σύνταξη για να δανειστούμε μεθόδους από αντικείμενα.

Για παράδειγμα, αν έχουμε ένα αντικείμενο με μια μέθοδο που θέλουμε να δανειστούμε:

```
var myObject = {
someMethod: function() {
// do something
}
};
```

Μπορούμε να δανειστούμε τη μέθοδο someMethod ως εξής:

```
var myMethod = myObject.someMethod;
```

Μπορούμε τώρα να καλέσουμε μια μέθοδο με τη δήλωση myMethod όπως κάνουμε με μια συνάρτηση.

```
myMethod();
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ανύψωση JavaScript (Hoisting)

Η ανύψωση είναι μια συμπεριφορά πολύ ιδιαίτερη της JavaScript. Αποτελεί έναν όρο ο οποίος συνήθως συγχέεται ακόμα και από τους σχεδιαστές ιστοσελίδων.

Για να κατανοήσουμε την συμπεριφορά αυτή, θα πρέπει να κατανοήσουμε τον όρο «ανύψωση». Η ανύψωση είναι μια προεπιλεγμένη συμπεριφορά της JavaScript για την μετακίνηση όλων των δηλώσεων στην κορυφή του τρέχοντος πεδίου εφαρμογής (στην κορυφή της τρέχουσας δέσμης ενεργειών ή στην τρέχουσα συνάρτηση) .

Παρατηρήστε τον ακόλουθο κώδικα:

```
var foo = 1; function bar() { if (!foo) { var foo = 10; } console.log(foo); } bar();
```

Τί νομίζετε ότι θα είναι το αποτέλεσμα αυτού του κώδικα;

Αν υπολογίσατε ότι το αποτέλεσμα θα είναι 10, τότε λανθάνετε. Το αποτέλεσμα αυτού του κώδικα είναι στην πραγματικότητα 1.

Αυτό γιατί η JavaScript ανυψώνει μόνο δηλώσεις (declaration) και όχι αρχικοποιήσεις (initialization). Όταν λοιπόν εκτελείται μια δήλωση if, η τιμή foo είναι απροσδιόριστη (undefined) και επομένως το αποτέλεσμα είναι 10.

Αυτό μπορεί να προκαλεί συνήθως σύγχυση, αλλά είναι σημαντικό να κατανοούμε πώς λειτουργεί η συμπεριφορά ανύψωσης στην JavaScript. Αυτό θα σας βοηθήσει να αποφεύγετε κοινά λάθη.

Κλείσιμο της JavaScript (Function Closures)

Το κλείσιμο της JavaScript είναι μια εσωτερική λειτουργία η οποία έχει πρόσβαση σε μεταβλητές που ορίζονται στο εσωτερικό ή εκτός μιας συνάρτησης. Η λειτουργία κλεισίματος μπορεί να έχει συνολικά πρόσβαση 1. σε μεταβλητές που ορίζονται στο

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

εσωτερικό μιας συνάρτησης (δηλαδή μεταξύ των αγκύλων), 2. σε μεταβλητές που ορίζονται εκτός μιας συνάρτησης και 3. σε καθολικές μεταβλητές.

Ένα κλείσιμο της JavaScript μπορεί να προκύψει όταν μια συνάρτηση ορίζεται μέσα σε μια άλλη συνάρτηση και η εσωτερική αναφέρεται σε μεταβλητές της εξωτερικής. Τα κλεισίματα χρησιμοποιούνται εκτενώς στις βιβλιοθήκες της JavaScript όπως το jQuery.

Πιο κάτω παρατίθεται ένα παράδειγμα κλεισίματος της JavaScript:

```
function outerFunction(x) {
  var innerFunction = function(y) {
    return x + y;
  }
  return innerFunction;
}

var add5 = outerFunction(5);
var add10 = outerFunction(10);

console.log(add5(2)); // 7
console.log(add10(2)); // 12
```

Στο παραπάνω παράδειγμα, έχουμε μια συνάρτηση externalFunction η οποία έχει μια μοναδική παράμετρο x. Αυτή η συνάρτηση περιέχει μια συνάρτηση innerFunction η οποία έχει μια μοναδική παράμετρο y. Η συνάρτηση innerFunction επιστρέφει το άθροισμα των x και y.

Όταν καλούμε την συνάρτηση outerFunction με μια τιμή, η κλήση επιστρέφει την εσωτερική συνάρτηση. Όταν καλούμε τη συνάρτηση που επιστρέφεται από την externalFunction με μια τιμή, η κλήση επιστρέφει το άθροισμα της τιμής που περάσαμε στην externalFunction και της τιμής που περάσαμε στην innerFunction.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Στο παραπάνω παράδειγμα, έχουμε δύο κλεισίματα: add5 και add10. Όταν καλούμε το κλείσιμο add5(2), τότε αυτό επιστρέφει το 7 επειδή το 5 περνά εκτός της συνάρτησης, externalFunction, ως τιμή του x, και το 2 περνά στο εσωτερικό της συνάρτησης, innerFunction, ως τιμή του y. Όταν καλούμε το κλείσιμο add10(2), τότε αυτό επιστρέφει το 12 επειδή το 10 περνά στην externalFunction ως τιμή του x, και το 2 περνά στην innerFunction ως τιμή του y.

Ένα κλείσιμο είναι μια συνάρτηση που χρησιμοποιεί μεταβλητές που ορίζονται σε εξωτερικές συναρτήσεις οι οποίες έχουν προηγουμένως επιστρέψει. Στο ακόλουθο παράδειγμα, η εσωτερική συνάρτηση plus() χρησιμοποιεί τη μεταβλητή num που ορίστηκε στην επιστροφή της εξωτερικής συνάρτησης Notification ():

```
function returnNotification() {
var num = 42;

 function plus() {
 return num + 1;
 }

 return plus;
}

var result = returnNotification();

console.log(result()); // 43
```

Στο παραπάνω παράδειγμα, έχουμε μια συνάρτηση returnNotification(). Αυτή η συνάρτηση έχει μια τοπική μεταβλητή num και μια συνάρτηση plus(). Η συνάρτηση Plus() επιστρέφει την τιμή του NUM συν 1. Η συνάρτηση ReturnNotification () επιστρέφει τη συνάρτηση Plus().

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Αναθέτουμε την τιμή επιστροφής της συνάρτησης returnNotification () στο μεταβλητό αποτέλεσμα. Όταν καλούμε το result(), τότε αυτό καλεί τη συνάρτηση plus() και επιστρέφει την τιμή του num συν 1.

Ένα κλείσιμο είναι μια συνάρτηση που χρησιμοποιεί μεταβλητές που ορίζονται σε εξωτερικές συναρτήσεις οι οποίες έχουν προηγουμένως επιστραφεί. Στο ακόλουθο παράδειγμα, η εσωτερική συνάρτηση plus() χρησιμοποιεί τη μεταβλητή num που ορίστηκε στην επιστροφή της εξωτερικής συνάρτησης Notification ():

```
function returnNotification() {
var num = 42;

 function plus() {
 return num + 1;
 }

 return plus;
}

var result = returnNotification();

console.log(result()); // 43
```

Στο παραπάνω παράδειγμα, έχουμε μια συνάρτηση returnNotification(). Αυτή η συνάρτηση έχει μια τοπική μεταβλητή num και μια συνάρτηση plus(). Η συνάρτηση Plus() επιστρέφει την τιμή του num συν 1. Η συνάρτηση ReturnNotification() επιστρέφει τη συνάρτηση Plus().

Αναθέτουμε την τιμή επιστροφής της συνάρτησης returnNotification() στο μεταβλητό αποτέλεσμα. Όταν καλούμε το result(), τότε αυτό καλεί τη συνάρτηση plus() και επιστρέφει την τιμή του num συν 1.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Τα κλεισίματα χρησιμοποιούνται εκτενώς στις βιβλιοθήκες της JavaScript όπως το jQuery. Στο παρακάτω παράδειγμα, χρησιμοποιούμε τη βιβλιοθήκη jQuery για να δημιουργήσουμε ένα κουμπί. Όταν κάνετε κλικ σε ένα κουμπί, εμφανίζεται ένα προειδοποιητικό παράθυρο (alert box) με το κείμενο "Hello world!":

```
$(function() {
var button = $ ('button');
button.click(function() {
alert('Hello world!');
  });
});
```

Στο παραπάνω παράδειγμα, έχουμε ένα στοιχείο κουμπιού με ένα αναγνωριστικό "button". Χρησιμοποιούμε τη βιβλιοθήκη jQuery για να επιλέξουμε το στοιχείο κουμπιού. Στη συνέχεια, χρησιμοποιούμε τη μέθοδο click() για να καταχωρήσουμε μια συνάρτηση που θα εκτελεστεί όταν πατηθεί το κουμπί.

Η λειτουργία που εκτελείται όταν πατιέται το κουμπί είναι ένα κλείσιμο. Έχει πρόσβαση στις μεταβλητές της εξωτερικής συνάρτησης, η οποία σε αυτή την περίπτωση είναι το στοιχείο κουμπιού. Όταν εκτελείται η λειτουργία, εμφανίζει ένα προειδοποιητικό παράθυρο με το κείμενο "Hello world!".

Αυστηρή Χρήση JavaScript (Strict Mode)

Τι είναι η αυστηρή χρήση της JavaScript;

Η αυστηρή χρήση ορίζει ότι ο κώδικας JavaScript πρέπει να εκτελείται σε αυστηρή λειτουργία, σύμφωνα με την οποία δεν μπορείτε να χρησιμοποιείτε μη δηλωμένες μεταβλητές. Η αυστηρή λειτουργία δεν είναι απλά ένα υποσύνολο, αλλά έχει σκόπιμα διαφορετική σημασιολογία από τον κανονικό κώδικα. Στα προγράμματα περιήγησης δεν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

απαιτείται η εκτέλεση κώδικα JS σε αυστηρή λειτουργία και πολλά από αυτά δεν την εφαρμόζουν.

Η αυστηρή λειτουργία διευκολύνει την εγγραφή "ασφαλούς" JavaScript. Η αυστηρή λειτουργία αλλάζει την προηγουμένως αποδεκτή "κακή σύνταξη" σε πραγματικά σφάλματα.

SyntaxError: Μη αναμενόμενη αξιολόγηση ή ορίσματα σε αυστηρή λειτουργία.

Η αυστηρή λειτουργία λύνει επίσης κάποια λάθη που απέτυχαν στην JavaScript.

- Διαγραφή μιας μεταβλητής ή μιας συνάρτησης.
- Χρήση μη δηλωμένων μεταβλητών.
- Διαγραφή μιας ιδιότητας ενός αντικειμένου.
- Προσπάθεια αλλαγής μιας ιδιότητας μόνο για ανάγνωση.
- Προσθήκη μιας ιδιότητας σε ένα αντικείμενο που δεν είναι επεκτάσιμο.
- Χρήση ενός αντικειμένου που έχει "παγώσει" (frozen) ή σφραγιστεί.
- Εγγραφή μιας ιδιότητας μόνο για ανάγνωση.
- Προσπάθεια αλλαγής ενός χαρακτηριστικού DontDelete σε μια ιδιότητα.
- Προσάρτηση μιας τιμής σε μια καθολική μεταβλητή μόνο για ανάγνωση.
- Αύξηση ή μείωση μιας ιδιότητας μόνο για ανάγνωση.
- Προσπάθεια αλλαγής ενός χαρακτηριστικού DontDelete σε μια ιδιότητα.
- Αντιστοίχιση μιας τιμής σε μια καθολική συνάρτηση μόνο για ανάγνωση.
- Προσπάθεια αλλαγής του πρωτότυπου μιας συνάρτησης.

Η αυστηρή λειτουργία αποτρέπει επίσης ή εκτελεί ρίψη των σφαλμάτων, όταν λαμβάνονται "ανασφαλείς" ενέργειες.

- Χρήση μη δηλωμένων μεταβλητών.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

- Δημιουργία μιας καθολικής μεταβλητής (χωρίς τη χρήση της δήλωσης var)
- Διαγραφή μιας μεταβλητής ή μιας συνάρτησης.
- Διαγραφή μιας ιδιότητας ενός αντικειμένου.
- Χρήση ενός αντικείμενο που έχει "παγώσει" (frozen) ή σφραγιστεί.
- Προσπάθεια αλλαγής ενός χαρακτηριστικού DontDelete σε μια ιδιότητα.
- Αντιστοίχιση μιας τιμής σε μια καθολική συνάρτηση μόνο για ανάγνωση.
- Αύξηση ή μείωση μιας ιδιότητας μόνο για ανάγνωση.
- Προσπάθεια αλλαγής του πρωτοτύπου μιας συνάρτησης.

Για να εκτελέσετε τον κώδικά σας σε αυστηρή λειτουργία, πρέπει να προσθέσετε την οδηγία "use strict" στην αρχή του κώδικά σας. Η οδηγία "use strict" δεν είναι μια συμβολοσειρά, αλλά μάλλον μια literal έκφραση που εμφανίζεται στην κορυφή ενός αρχείου, σεναρίου ή σώματος λειτουργίας.

```
"use strict";
```

Η οδηγία μπορεί να τοποθετηθεί είτε στην αρχή ενός σεναρίου είτε στην αρχή μιας συνάρτησης.

Παράδειγμα 1: Τοποθέτηση της οδηγίας στην αρχή ενός σεναρίου.

```
"use strict"; x = 3.14; // Αυτό θα προκαλέσει σφάλμα επειδή το x δεν δηλώνεται.
```

Παράδειγμα 2: Τοποθέτηση της οδηγίας στην αρχή μιας συνάρτησης.

```
function myFunction() { "use strict" ; y = 3.14; // Αυτό θα προκαλέσει σφάλμα επειδή το y δεν δηλώνεται. }
```

Εάν η αυστηρή λειτουργία δηλώνεται μέσα σε μια συνάρτηση, τότε έχει τοπική εμβέλεια (μόνο ο κώδικας μέσα στη συνάρτηση είναι σε αυστηρή λειτουργία). Εάν η αυστηρή

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

λειτουργία χρησιμοποιείται στην κορυφή ενός σεναρίου, τότε αυτή θα εφαρμοστεί σε ολόκληρο το σενάριο.

Εάν η αυστηρή λειτουργία χρησιμοποιείται στην κορυφή ενός σεναρίου, τότε αυτή θα εφαρμοστεί σε ολόκληρο το σενάριο.

Εάν η αυστηρή λειτουργία χρησιμοποιείται μέσα σε μια συνάρτηση, τότε θα εφαρμοστεί μόνο στη συνάρτηση στην οποία δηλώνεται.

Στην αυστηρή λειτουργία, οποιαδήποτε αναφορά σε μη δηλωμένη μεταβλητή θα προκαλέσει σφάλμα αναφοράς.

```
function myFunction() { "use strict" ; y = 3.14; // Αυτό θα προκαλέσει σφάλμα επειδή το y είναι μη δηλωμένη μεταβλητή. }
```

Δεν επιτρέπεται η χρήση της αυστηρής λειτουργίας σε μια μη δηλωμένη μεταβλητή, σε ένα μη δηλωμένο στοιχείο μιας παραμέτρου ή σε μια μη δηλωμένη ιδιότητα αφού αυτό θα επιφέρει σφάλμα.

Ανάλυση δεδομένων JSON στην JavaScript (Parsing)

Τι είναι η ανάλυση δεδομένων JSON;

JSON σημαίνει JavaScript Object Notation. Μια κοινή χρήση του JSON είναι η ανταλλαγή ή η αποθήκευση δεδομένων με οργανωμένο τρόπο που διευκολύνει την προσβασιμότητα σε αυτά. Ίσως το καλύτερο πλεονέκτημα του JSON είναι ότι η σύνταξή του μοιάζει με απλά αγγλικά, έτσι λοιπόν, έστω και αν δεν γνωρίζετε τίποτα για το προγραμματισμό ή την κωδικοποίηση, μπορείτε ωστόσο να καταλάβετε τις σημασίες που κρύβουν οι κώδικες που χρησιμοποιούνται!

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ανάλυση δεδομένων JSON στην JavaScript

Το παρακάτω παράδειγμα δείχνει πώς να αναλύετε μια συμβολοσειρά και να δημιουργείτε ένα αντικείμενο από αυτήν. Θα πρέπει πάντα να προτιμάτε την ανάλυση δεδομένων όπως παρουσιάζεται πιο παρακάτω:

```
var jsonString = '{"name":"John","age":30,"city":"New York"}'; var obj = JSON.parse (jsonString); console.log ('Name: ',obj['name'],' Age:',obj['age'], ' City:' , obj ['city'] ); // Outputs Name : John, age : 30 and city as New york
```

Ανάλυση δεδομένων JSON στην JavaScript

Το παρακάτω παράδειγμα δείχνει πώς να αναλύετε ένθετα δεδομένα (nested data) σε μια συμβολοσειρά σε ένα αντικείμενο χρησιμοποιώντας τη συνάρτηση ~ () (η οποία δεν συνιστάται). Μπορείτε επίσης να χρησιμοποιήσετε την ίδια μέθοδο για την ανάλυση ένθετων αντικείμενων, αλλά θα πρέπει να είστε προσεκτικοί κατά τη διαδικασία, διότι αν υπάρχουν λάθη, τότε το πρόγραμμά σας θα καταρρεύσει.

Κωδικοποίηση δεδομένων ως JSON στην JavaScript

Το παρακάτω παράδειγμα δείχνει τον τρόπο με τον οποίο μπορείτε να κωδικοποιήσετε δεδομένα σε μια συμβολοσειρά χρησιμοποιώντας τη συνάρτηση JSON.stringify ().

```
var obj = { name : 'John', age: 30, city:'New York' }; var jsonString = JSON.stringify (obj); console.log ('JSON String is ', jsonString); // Εξάγει ένα αντικείμενο συμβολοσειράς ως JSON
```

Διαχείριση Σφαλμάτων JavaScript (Error Handling)

Η JavaScript είναι μια πολύ ευέλικτη γλώσσα, που σημαίνει ότι υπάρχουν πολλοί τρόποι συγγραφής ενός κώδικα. Αυτή όμως η ευελιξία που παρέχει η JavaScrip ως προς την

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

συγγραφή κωδίκων μπορεί να οδηγήσει σε σφάλματα εάν δεν προσέχετε πως να συντάσσεται τη δομή τους.

Ένα από τα πιο συνηθισμένα λάθη που κάνουν οι προγραμματιστές είναι να υποθέτουν ότι όλος ο κώδικας θα εκτελεστεί σωστά. Αυτή η υπόθεση μπορεί να προκαλέσει προβλήματα επειδή η JavaScript είναι μια διερμηνευμένη γλώσσα, που σημαίνει ότι κάθε γραμμή κώδικα εκτελείται όπως διαβάζεται από τον διερμηνέα. Εάν υπάρχει κάποιο σφάλμα στη σύνταξη του κώδικά σας, ο διερμηνέας θα σταματήσει την εκτέλεσή του στο σημείο όπου εμφανίζεται το σφάλμα και θα εμφανίσει ένα μήνυμα σφάλματος.

Στο μάθημα αυτό θα μάθετε πώς να διαχειρίζεστε τα σφάλματα με επιδεξιότητα, έτσι ώστε το πρόγραμμά σας να μπορεί να συνεχίζει απρόσκοπτα τη λειτουργία του, έστω και αν υπάρχουν κάποια προβλήματα με τα δεδομένα που εισήχθησαν ή με κάποια μέρη στον κώδικα.

Το πρώτο πράγμα που πρέπει να κάνετε είναι να εντοπίσετε πού μπορεί να εμφανιστούν τυχόν σφάλματα στη σύνταξη του κώδικά σας. Για παράδειγμα, αν διαβάζετε δεδομένα από ένα αρχείο, υπάρχει η πιθανότητα το αρχείο να είναι στην πραγματικότητα ανύπαρκτο ή το περιεχόμενό του να έχει αλλοιωθεί. Εάν εργάζεστε με την εισαγωγή δεδομένων χρήστη, υπάρχει η πιθανότητα να εισάγετε μη έγκυρα δεδομένα. Αφού πρώτα εντοπίσετε τις πιθανές πηγές σφάλματος, μπορείτε να ξεκινήσετε ξανά την συγγραφή του κώδικα για να διορθώσετε τα λάθη που υπάρχουν σ' αυτόν.

Ένας κοινός τρόπος με τον οποίο μπορείτε να αντιμετωπίσετε τυχόν σφάλματα στην συγγραφή ενός κώδικα είναι η χρήση των μπλοκ try/catch. Αυτά τα μπλοκ σας επιτρέπουν να «δοκιμάσετε» έναν κώδικα και να «αρπάξετε» οποιοδήποτε σφάλμα εμφανιστεί κατά την ώρα της εκτέλεσης του. Η σύνταξη για τα μπλοκ try/catch πρέπει να έχει ως εξής:

```
try { // Ο κώδικας δοκιμής καταχωρείται εδώ } catch (error) { // Ο κώδικας για τον εντοπισμό των σφαλμάτων καταχωρείται εδώ }
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Ο κώδικας μέσα στο μπλοκ try θα εκτελεστεί πρώτος. Αν δεν προκύψουν σφάλματα, τότε ο κώδικας μέσα στο μπλοκ catch δεν θα εκτελεστεί ποτέ. Ωστόσο, εάν εμφανιστεί κάποιο σφάλμα, τότε η εκτέλεση θα μεταβεί απευθείας στο μπλοκ catch και οποιοσδήποτε κώδικας προκύψει από το μπλοκ block θα παραλειφθεί.

Ας δούμε ένα παράδειγμα του πώς λειτουργεί. Ας υποθέσουμε ότι έχετε μια συνάρτηση η οποία διαβάζει τα δεδομένα από ένα αρχείο και τα εκτυπώνει στην κονσόλα. Ωστόσο, υπάρχει η πιθανότητα το αρχείο να μην είναι υπαρκτό:

```
function readFile(filename) { try { // Code to read and print the contents of "filename" goes here } catch (error) { console.log("Error reading file: " + error); } }
```

Σε αυτό το παράδειγμα, η λειτουργία readFile θα προσπαθήσει να διαβάσει και να εκτυπώσει τα περιεχόμενα ενός αρχείου. Εάν παρουσιαστεί σφάλμα κατά την ανάγνωση του αρχείου, τότε θα "πιαστεί" από το μπλοκ catch και θα εκτυπωθεί στην κονσόλα.

Είναι σημαντικό να σημειωθεί ότι μόνο τα σφάλματα που προκύπτουν μέσα στο μπλοκ try θα "πιαστούν" από το μπλοκ catch. Εάν υπάρχει σφάλμα στο ίδιο το μπλοκ catch (για παράδειγμα, εάν ξεχάσετε να κλείσετε ένα άγκιστρο), τότε αυτό δεν θα "πιαστεί" και το πρόγραμμά σας θα καταρρεύσει.

Ένας άλλος τρόπος αντιμετώπισης σφαλμάτων είναι η χρήση του ενσωματωμένου αντικειμένου σφάλματος της JavaScript (built-in Error object). Το αντικείμενο σφάλματος μπορεί να χρησιμοποιηθεί για τη δημιουργία προσαρμοσμένων μηνυμάτων σφάλματος (custom error messages) τα οποία μπορούν στη συνέχεια να περαστούν σε μια δήλωση:

```
Throw new Error ("Αυτό είναι ένα προσαρμοσμένο μήνυμα σφάλματος")
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Όταν εκτελεστεί αυτός ο κώδικας, θα γίνει μια ρίψη σφάλματος και η εκτέλεση της τρέχουσας λειτουργίας θα σταματήσει. Οποιοσδήποτε επόμενος κώδικας στη λειτουργία δεν θα εκτελεστεί.

Μπορείτε επίσης να χρησιμοποιήσετε το αντικείμενο σφάλματος για να αντιμετωπίσετε σφάλματα που προκύπτουν σε έναν ασύγχρονο κώδικα. Για παράδειγμα, ας υποθέσουμε ότι θέλετε να εκτελέσετε μια λειτουργία που κάνει ένα αίτημα http και επιστρέφει τα δεδομένα απόκρισης:

```
function makeRequest(url) { return new Promise((resolve, reject) => { http.get(url, (response) => { // Code to handle the response goes here }); }); }
```

Αυτή η λειτουργία επιστρέφει μια υπόσχεση (promise), που σημαίνει ότι τα δεδομένα που επιστρέφονται από το αίτημα http θα είναι διαθέσιμα κάποια στιγμή στο μέλλον. Εάν προκύψει σφάλμα κατά την υποβολή του αιτήματος (για παράδειγμα, εάν η διεύθυνση URL δεν είναι έγκυρη), θα πιαστεί από τον ενσωματωμένο διαχειριστή σφαλμάτων της JavaScript και θα απορριφθεί με ένα μήνυμα σφάλματος.

Η απόρριψη αυτή θα επιφέρει την επιστροφή της υπόσχεσης με ένα σφάλμα, το οποίο μπορεί στη συνέχεια να αντιμετωπιστεί από τον κώδικα που ονομάζεται makeRequest:

```
makeRequest("http://www.example.com") .then((data) => { // Ο κώδικας διαχείρισης δεδομένων καταχωρείται εδώ }) .catch((error) => { console.log("Error making request: " + error); });
```

Στο παράδειγμα αυτό, έχουμε καταχωρήσει μια λειτουργία επιστροφής κλήσης για το γεγονός "then" που θα κληθεί αν η υπόσχεση επιλυθεί επιτυχώς. Καταχωρήσαμε επίσης μια λειτουργία επιστροφής κλήσης για το γεγονός "catch" το οποίο θα κληθεί αν προκύψει κάποιο σφάλμα. Αυτό μας επιτρέπει να αντιμετωπίσουμε τα σφάλματα επιδέξια και να συνεχίσουμε να εκτελούμε τον κώδικά μας ακόμα και αν υπάρχουν κάποια προβλήματα με τα δεδομένα εισόδου ή άλλα μέρη του κώδικα.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Κανονικές εκφράσεις JavaScript (Regular Expressions)

Οι κανονικές εκφράσεις είναι ένα πολυδύναμο εργαλείο της JavaScript που χρησιμοποιείται συχνά με δύο μεθόδους συμβολοσειράς: search () και replace (), δηλαδή για την εκτέλεση των τύπων λειτουργιών αναζήτησης ή αντικατάστασης κειμένου.

Μια κανονική έκφραση είναι μια ακολουθία χαρακτήρων που σχηματίζει ένα πρότυπο αναζήτησης. Για παράδειγμα, αν χρησιμοποιηθεί η κανονική έκφραση /A/ θα ταιριάξει με κάθε συμβολοσειρά που περιέχει ένα κεφαλαίο A: /A/.

Οι κανονικές εκφράσεις μπορούν να χρησιμοποιηθούν για την εκτέλεση ενός ευρέος φάσματος εργασιών, όπως:

- Για την εξαγωγή πληροφοριών από μια συμβολοσειρά (π.χ. μια διεύθυνση ηλεκτρονικού ταχυδρομείου),
- για την επικύρωση των στοιχείων εισόδου χρήστη (π.χ. για την δημιουργία ενός ισχυρού κωδικού πρόσβασης),
- για τις λειτουργίες αναζήτησης ή αντικατάστασης κειμένου σε μια συμβολοσειρά (π.χ. αλλάζοντας όλες τις εμφανίσεις του "a" σε "b"),
- για την μορφοποίηση κειμένου (π.χ. για την προσθήκη κομμάτων μεταξύ λέξεων),
- για την παραγωγή τυχαίων συμβολοσειρών (π.χ. για την δημιουργία μοναδικών αναγνωριστικών ή κωδικών πρόσβασης) και πολλά άλλα!

Στην JavaScript, οι κανονικές εκφράσεις είναι επίσης αντικείμενα. Αυτά τα αντικείμενα περιέχουν ένα σύνολο μεθόδων που μπορούν να χρησιμοποιηθούν για την εκτέλεση διαφόρων λειτουργιών σε συμβολοσειρές, όπως η αναζήτηση μοτίβων ή η αντικατάσταση κειμένου.

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Δημιουργία Κανονικών Εκφράσεων στην JavaScript

Υπάρχουν δύο τρόποι για να δημιουργήσετε κανονικές εκφράσεις στην JavaScript, είτε χρησιμοποιώντας μια κανονική literal έκφραση ή τη συνάρτηση κατασκευαστή RegExp().

Μια κανονική literal έκφραση είναι απλά μια συμβολοσειρά η οποία περιέχει ένα μοτίβο το οποίο θέλετε να ταιριάξετε, και το οποίο πρέπει να περικλείεται εντός δύο καθέτων (/). Για παράδειγμα:

```
const regex = /abc/ ; console .log(regex); // => /abc/
```

Ο παραπάνω κώδικας δημιουργεί ένα νέο αντικείμενο κανονικής έκφρασης που περιέχει το μοτίβο abc και το εκχωρεί στη μεταβλητή regex . Η τιμή αυτής της μεταβλητής μπορεί στη συνέχεια να χρησιμοποιηθεί όπως οποιοδήποτε άλλο αντικείμενο της JavaScript. Συγκεκριμένα, μπορούμε να χρησιμοποιήσουμε τις μεθόδους του για την εκτέλεση διαφόρων λειτουργιών στις συμβολοσειρές.

Η συνάρτηση κατασκευαστή RegExp() μπορεί επίσης να χρησιμοποιηθεί για τη δημιουργία κανονικών εκφράσεων. Αυτή η συνάρτηση παίρνει δύο ορίσματα: το πρώτο είναι μια συμβολοσειρά που περιέχει το μοτίβο που θέλετε να ταιριάξετε, και το δεύτερο είναι ένα προαιρετικό όρισμα "flags" που καθορίζει συγκεκριμένες ρυθμίσεις για το πώς πρέπει να λειτουργεί το regex. Για παράδειγμα:

```
const regex = new RegExp ( 'abc' ); console .log(regex); // => /abc/ const flagsRegex = new RegExp ( '123', 'i' ); console .log(flagsRegex); //=>/123/i; όπου το i είναι ένας τροποποιητής, ο οποίος τροποποιεί την αναζήτηση ώστε να μην κάνει διάκριση πεζών-κεφαλαίων)
```

Ο παραπάνω κώδικας δημιουργεί δύο νέα αντικείμενα κανονικής έκφρασης. Το πρώτο περιέχει το μοτίβο abc και είναι ευαίσθητο σε πεζά/κεφαλαία (η προεπιλεγμένη ρύθμιση). Το δεύτερο αντικείμενο έχει επίσης ένα μοτίβο 123, αλλά η αναζήτηση γίνεται έτσι ούτως ώστε να μην γίνεται διάκριση πεζών-κεφαλαίων, χάρη στον τροποποιητή i που περάστηκε ως το δεύτερο όρισμα του RegExp().

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Σημείωση: Στην JavaScript, οι επαναλήψεις που δημιουργούνται χρησιμοποιώντας κυριολεκτική σημειογραφία είναι αμετάβλητες. Δεν μπορούν να αλλάξουν τις ιδιότητες ή τις μεθόδους τους μετά τον ορισμό τους. Από την άλλη πλευρά, τα regexes που δημιουργούνται με το RegExp() μπορούν να τροποποιηθούν ανά πάσα στιγμή, επειδή είναι απλά μια άλλη λειτουργία του κατασκευαστή, όπως η Ημερομηνία , ο Πίνακας κ.λπ.

Επικύρωση φόρμας JavaScript (Form Validation)

Η JavaScript είναι μια γλώσσα σεναρίου από την πλευρά του πελάτη, που σημαίνει ότι η παραγωγή του τελικού περιεχομένου HTML πραγματοποιείται στο πρόγραμμα περιήγησης των επισκεπτών. Αυτό σας επιτρέπει να εκτελείτε λειτουργίες όπως π.χ. στην περίπτωση που θέλετε να ελέγξετε αν μια ηλεκτρονική διεύθυνση καταχωρήθηκε ορθά πριν αποσταλεί στον διακομιστή. Επίσης, αυτό μειώνει την πίεση στον διακομιστή επειδή τα δεδομένα φόρμας δεν είναι αναγκαίο να υποβληθούν μέχρι να γίνει ο έλεγχος σφαλμάτων από την JavaScript.

Το πρώτο πράγμα που πρέπει να κάνετε είναι να δημιουργήσετε μια φόρμα. Μπορείτε να χρησιμοποιήσετε τον κώδικα πιο κάτω:

```
<form action="yourpage.php" method="post"> <p>Name:</p><input type="text" name="name"><br /> <p>Email Address:</p><input type="text" name= "email"><br /> </form>
```

Αυτό θα δημιουργήσει μια βασική φόρμα με δύο πεδία, ένα για το όνομα και ένα για μια διεύθυνση ηλεκτρονικού ταχυδρομείου. Το χαρακτηριστικό δράσης (action attribute) λέει στο πρόγραμμα περιήγησης πού να στείλει τα δεδομένα όταν υποβληθούν, που στην προκειμένη περίπτωση είναι στο yourpage.php . Μπορείτε να αλλάξετε το χαρακτηριστικό δράσης σε οποιαδήποτε σελίδα θέλετε να επεξεργαστείτε τα δεδομένα φόρμας. Το χαρακτηριστικό της μεθόδου καθορίζει τον τρόπο αποστολής των δεδομένων, είτε το GET είτε το POST. Στις περισσότερες περιπτώσεις θα θέλετε να χρησιμοποιήσετε το POST επειδή είναι πιο ασφαλές από το GET, αλλά υπάρχουν

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

ορισμένες περιπτώσεις όπου η χρήση του GET μπορεί να είναι προτιμότερη (για παράδειγμα, αν θέλετε οι χρήστες να μπορούν να προσθέτουν στους σελιδοδείκτες τους αναζητήσεις που έχουν κάνει). Για περισσότερες πληροφορίες σχετικά με αυτές τις μεθόδους μπορείτε να παρακολουθήσετε το HTML Forms Tutorials: Introduction article στο οποίο αναπτύσσονται λεμπτομερώς.

Το επόμενο βήμα που πρέπει να κάνουμε είναι να προσθέσουμε ένα κουμπί υποβολής, έτσι ώστε οι επισκέπτες να μπορούν να μας στείλουν τα στοιχεία τους:

```
<input type="submit" value="Submit">
```

Μπορείτε να τοποθετήσετε αυτό το κουμπί οπουδήποτε μέσα στις ετικέτες φόρμας. Τώρα, αν αποθηκεύσετε τη σελίδα σας και κάνετε μια δοκιμή, τότε θα δείτε ότι όταν κάνετε κλικ στο κουμπί υποβολής ένα νέο παράθυρο θα ανοίξει μόνο με το yourpage.php σε αυτό (υποθέτοντας ότι αυτό έχετε ορίσει ως ενέργεια). Αυτό συμβαίνει επειδή δεν έχουμε πει στο πρόγραμμα περιήγησής μας να κάνει κάτι άλλο ακόμα - προς το παρόν το μόνο που κάνει η υποβολή της φόρμας είναι να μας στείλει σε μια άλλη σελίδα που δεν είναι πολύ χρήσιμη!

Έτσι, θα χρειαστεί η JavaScript για να ελέγξει τυχόν σφάλματα πριν από την αποστολή οποιωνδήποτε δεδομένων, προσθέτοντας τον κώδικα πιο κάτω:

```
<script type="text/javascript">
function checkForm() {
  var errorMsg = "";
  // Check each field to make sure it has a value.
  if (document.form1.name == "") {
     errorMsg += "- Please enter your name";
  } else if (document.form1.email == "" || document.form1.email_confirm != document.form1.email) {
     errorMsg += "- You must provide an e-mail address and confirm it by entering the same address again.";
  } else {
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
    return true;
  }
  // This line submits the form only when there are no errors in any of the fields, otherwise we display our message telling them what they need to do:
  alert(errormsg);
  return false;
}
</script>
<body onload="checkForm();" >

 ... rest of page here...

</body>
```

JavaScript Cookies

Η δημιουργία ενός cookie στην JavaScript είναι πολύ απλή. Απλά πρέπει να χρησιμοποιήσετε ένα αντικείμενο εγγράφου και την μέθοδο createCookie(). Η σύνταξη αυτής της λειτουργίας θα πρέπει να έχει ως εξής:

```
document.cookie = "name=value; expires=date";
```

Η παράμετρος όνομα (name parameter) απευθύνεται στο όνομα του cookie που θέλετε να δημιουργήσετε, ενώ η παράμετρος τιμή αναφέρεται στο περιεχόμενο του (συμβολοσειρά). Εάν θέλετε να θέσετε μια ημερομηνία λήξης σ΄ αυτό, τότε πρέπει να προσθέσετε μια άλλη ιδιότητα, που ονομάζεται expires με μια έγκυρη συμβολοσειρά Date() ή απλά περάστε μια μηδενική τιμή null, εάν δεν θέλετε να έχει μια ημερομηνία λήξης.

```
    document.cookie="username=John Doe;expires"+Date(30); //creates username cookies that will expire after 30 days from now
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

```
document.cookie="username=John Doe;expires"+Date(30*24); //creates username cookies that will expire after 30 days from now
```

Η ανάγνωση ενός cookie στην JavaScript είναι επίσης πολύ απλή, απλά πρέπει να χρησιμοποιήσετε το αντικείμενο του εγγράφου και τη μέθοδο getCookie(). Η σύνταξη αυτής της λειτουργίας θα πρέπει να έχει ως εξής:

```
var name = getCookie("name");
```

Η παράμετρος αντιπροσωπεύει το όνομα του cookie σας (συμβολοσειρά). Αυτή η λειτουργία επιστρέφει την μηδενική τιμή null αν δεν υπάρχει τέτοιο cookie ή έχει ήδη λήξει. Μπορείτε επίσης να διαβάσετε όλα τα υφιστάμενα cookies χρησιμοποιώντας μια άλλη ενσωματωμένη ιδιότητα που ονομάζεται document.cookies, η οποία περιέχει έναν πίνακα με όλα τα διαθέσιμα cookies της ιστοσελίδας σας, τα οποία διαχωρίζονται με το ερωτηματικό ";". Για παράδειγμα:

```
alert(document.cookies) ;//θα εμφανίσει κάτι σαν : userName1=value1;userName2=value2 etc... /*αυτός ο κώδικας δεν εμφανίζει τίποτα επειδή δεν υπάρχουν ενεργά cookies*/
```

Η ενημέρωση ενός cookie στην JavaScript είναι επίσης πολύ απλή. Απλά πρέπει να το δημιουργήσετε ξανά χρησιμοποιώντας το ίδιο όνομα και τη ίδια τιμή, αλλά αυτή τη φορά πρέπει επίσης να ορίσετε μια ημερομηνία λήξης που είναι μεγαλύτερη από την τρέχουσα (αν υπάρχει). Για παράδειγμα:

```
document.cookie = "name=newValue; expires=" + Date(Date().getTime()+1); /* αυτός ο κώδικας θα ενημερώσει τα cookies σας*/

document.cookie = "username=John Doe;expires"+Date(30*24); // δημιουργεί ένα cookie ονόματος χρήστη τα οποία θα λήξουν σε 30 μέρες από τα τώρα /*αυτός ο κώδικας δημιουργεί νέα cookies εάν δεν υπάρχουν ενεργά ή εάν τα υφιστάμενα δεν έχουν ενημερωθεί */
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

JavaScript Ajax : Αποστολή Αιτημάτων σε Server

Το Ajax είναι ένα σύνολο από Web development τεχνικές για τη δημιουργία διαδραστικών διαδικτυακών εφαρμογών. Στόχος του Ajax είναι η κατασκευή ιστοσελίδων που αποδίδουν πιο αποτελεσματικά το περιεχόμενό τους μέσω της ανταλλαγής μικρότερων όγκων δεδομένων με τους διακομιστές που λειτουργούν στο παρασκήνιο, ούτως ώστε ολόκληρη η σελίδα να μην χρειάζεται επαναφόρτωση κάθε φορά που ο χρήστης εκτελεί διαφορετικές ενέργειες πάνω σ' αυτή.

Για να χρησιμοποιήσετε το Ajax, θα πρέπει να χρησιμοποιήσετε μια από τις βιβλιοθήκες της JavaScript όπως το jQuery ή το Prototype. Αυτές οι βιβλιοθήκες παρέχουν έναν εύκολο τρόπο αποστολής και λήψης δεδομένων από έναν διακομιστή χωρίς να χρειάζεται ανανέωση της σελίδας.

Μόλις συμπεριλάβετε τη βιβλιοθήκη Ajax στην ιστοσελίδα σας, μπορείτε να ξεκινήσετε να υποβάλλετε αιτήματα στον διακομιστή. Για παράδειγμα, αν θέλατε να φορτώσετε κάποια δεδομένα από ένα αρχείο στον διακομιστή, θα χρησιμοποιούσατε τον ακόλουθο κώδικα:

```
$ .ajax({ url: 'data.json', success: function(data) { // do something with the data } });
```

Αυτός ο κωδικός θα υποβάλει ένα αίτημα Ajax στη διεύθυνση URL 'data.json'. Εάν το αίτημα είναι επιτυχές, τότε η επιτυχής λειτουργία στην επιστροφή κλήσης (callback) θα εκτελεστεί με τα δεδομένα από το διακομιστή ως όρισμα. (?)

Εάν θέλετε να στείλετε δεδομένα στον διακομιστή, μπορείτε να χρησιμοποιήσετε την επιλογή "δεδομένα" από την μέθοδο '$ .ajax ()':

```
$ .ajax({ url: 'saveData.php', type: 'POST', data: { name: 'John', age: 20 } });
```

Αυτός ο κώδικας θα κάνει μια αίτηση POST στη διεύθυνση URL 'saveData.php'. Τα δεδομένα που αποστέλλονται στο διακομιστή καθορίζονται στην επιλογή 'data' ως αντικείμενο. Σε αυτό το παράδειγμα, αποστέλλουμε δύο στοιχεία: "όνομα" και "ηλικία".

Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)

Το παρακάτω παράδειγμα θα σας δείξει πώς να υποβάλετε ένα αίτημα Ajax GET στην JavaScript:

```
$ .ajax({ url: 'getData.php', type: 'GET', success: function(data) { // do something with the data } });
```


Η υποστήριξη της Ευρωπαϊκής Επιτροπής για την έκδοση της παρούσας δημοσίευσης δε συνιστά επικύρωση/επικρότηση του περιεχομένου της, το οποίο αντικατοπτρίζει μόνο τις απόψεις των συντακτών της, και η Επιτροπή δε φέρει καμία ευθύνη για οποιαδήποτε τυχόν χρήση των πληροφοριών που περιέχονται σε αυτήν. (Αριθμός Έργου : 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN)