



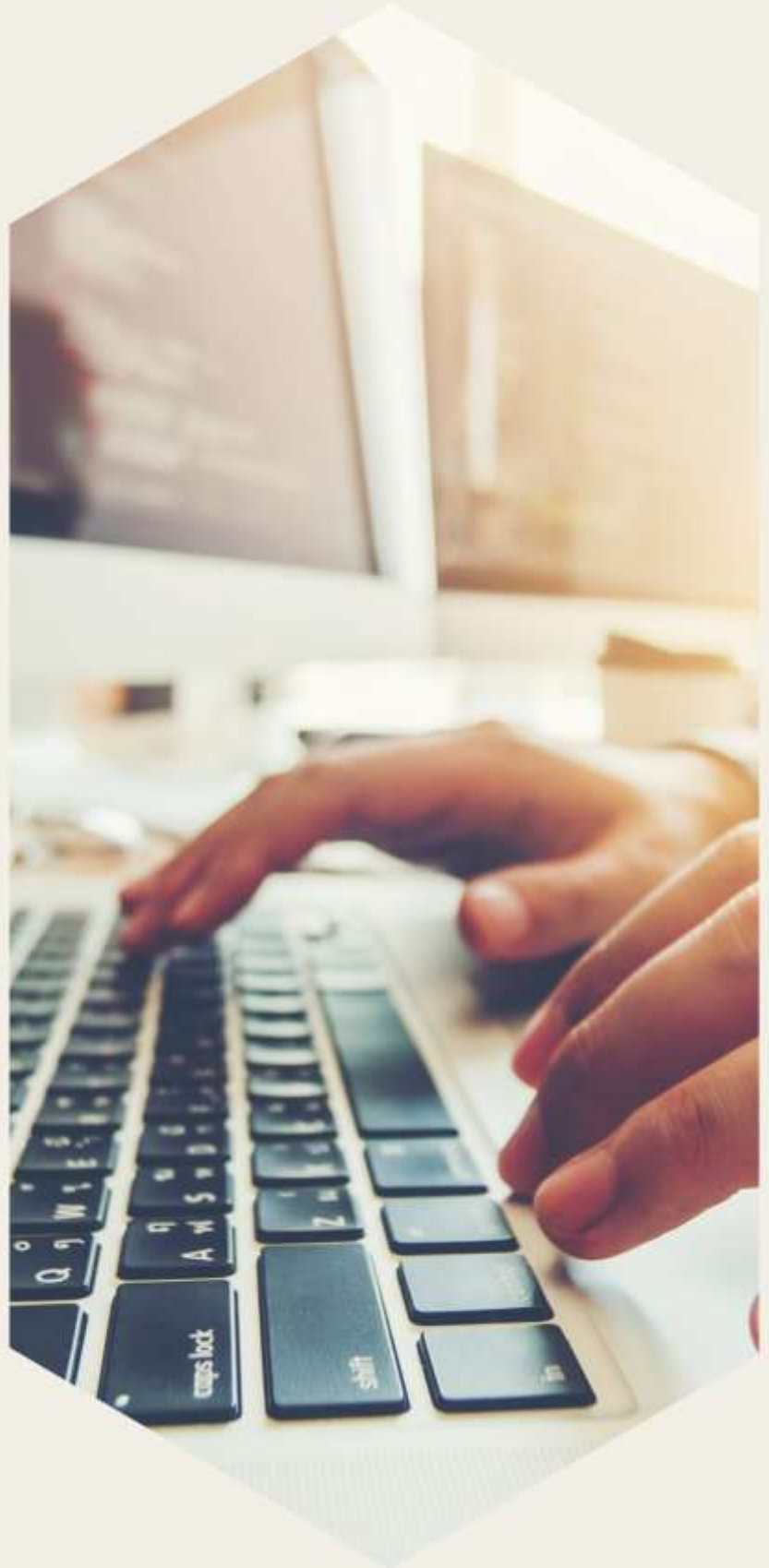
3.1.:

Εκπαιδευτικό Υλικό του έργου Code4SP

WP3:

Εκπαιδευτικό υλικό του Code4SP

Συντάχθηκε από:



Πληροφορίες για το έργο

Ακρωνύμιο του έργου: Code4SP

Τίτλος του έργου: Coding for Social Promotion

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

Ιστοσελίδα του έργου: www.code4sp.eu

Εταίρος Συντάκτης: CEPROF

Έκδοση Εγγράφου: 1

Ημερομηνία Παραγωγής: 11/02/2022

Ιστορικό Εγγράφου			
Ημερομηνία	Έκδοση	Συντάκτης	Περιγραφή
11/02/2022	1	CEPROF	Προσχέδιο

Περιεχόμενα

Πληροφορίες για το έργο.....	Error! Bookmark not defined.
Ενότητα:.....	Error! Bookmark not defined.
2. HTML.....	Error! Bookmark not defined.
Προαπαιτούμενες γνώσεις:	Error! Bookmark not defined.
Φόρτος Εργασίας:.....	Error! Bookmark not defined.
Περιγραφή:	Error! Bookmark not defined.
Μαθησιακά Αποτελέσματα:.....	Error! Bookmark not defined.
Απαιτούμενα Υλικά:	5
Σενάριο Μαθήματος:.....	Error! Bookmark not defined.
Υποενότητες:	Error! Bookmark not defined.
Επιπλέον πόροι:	Error! Bookmark not defined.
2.1. Τα βασικά της HTML.....	7
Η ανατομία ενός στοιχείου HTML.....	Error! Bookmark not defined.
Προγράμματα Περιήγησης	9
Δομή μιας σελίδας HTML.....	Error! Bookmark not defined.
Ιστορικό της HTML.....	10
Επεξεργαστές HTML.....	Error! Bookmark not defined.
Βασικά Παραδείγματα HTML	Error! Bookmark not defined. 3
Χαρακτηριστικά HTML	Error! Bookmark not defined. 7
Επικεφαλίδες HTML.....	Error! Bookmark not defined.
Παράγραφοι HTML	Error! Bookmark not defined.
Στυλ της HTML (Styles).....	Error! Bookmark not defined.
Μορφοποίηση κειμένου στην HTML.....	Error! Bookmark not defined.
Μορφοποίηση κειμένου στην HTML.....	Error! Bookmark not defined. 9
Σχόλια στην HTML	33
Χρώματα στην HTML.....	34
Σύνδεσμοι στην HTML	388
Εικόνες στην HTML.....	40
Εικονίδιο Συντόμευσης στην HTML (Favicon)	433
Πίνακες στην HTML	455

Λίστες στην HTML.....	499
Φόρμες στην HTML	51
Ενσωματωμένο πλαίσιο iFrame στην HTML	45
2.2. Προηγμένες έννοιες της HTML	60
Doctypes στην HTML.....	60
Διατάξεις στην HTML (Layouts).....	60
Επικεφαλίδα στην HTML (Head).....	62
Το στοιχείο base στην HTML (βασική διεύθυνση URL)	63
Το στοιχείο link στην HTML	64
Το στοιχείο style στην HTML.....	64
Το στοιχείο meta στην HTML	65
Το στοιχείο script στην HTML.....	67
Οντότητες στην HTML.....	69
Διεύθυνση URL στην HTML	71
Η Επικύρωση στην HTML.....	75
Λειτουργίες στην HTML5.....	76
Νέοι τύποι εισαγωγής της HTML5	Error! Bookmark not defined.
Ο Καμβάς στην HTML5.....	84
Τα γραφικά SVG στην HTML5	92
Ήχος στην HTML 5	98
Βίντεο στην HTML5.....	92
Αποθήκευση δεδομένων στον ιστό στην HTML5 (Web storage)	102
Εφαρμογές προσωρινής μνήμης στην HTML5 (Application Cache)	104
Web Workers στην HTML5	108
Αποστολή συμβάντων από τον διακομιστή στην HTML5 (Server-Sent Events)	Error! Bookmark not defined.
Γεωτοποθεσία και γεωντοπισμός μέσω HTML5 (Geolocation)	114
Σύρσιμο και απόθεση στην HTML5 (Drag and Drop)	Error! Bookmark not defined.
Αναφορές της HTML5	Error! Bookmark not defined.

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>)

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

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>
```

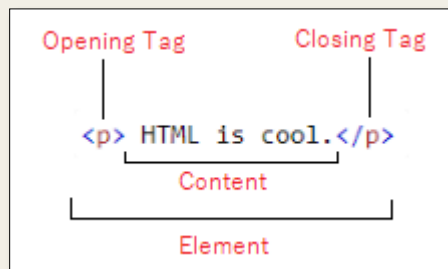
Εικόνα 2 – Κωδικοποίηση για την παράγραφο «Η HTML είναι κουλ» (Πηγή: Συντάκτης)

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 <TITILE> <Title>, <TiTiE>, , , κ.λπ., και να

λειτουργήσει. Ωστόσο, η καλύτερη πρακτική είναι να γράφετε όλες τις ετικέτες με πεζά γράμματα για συνέπεια και αναγνωσιμότητα.

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-παράγραφος), 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.

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

Προγράμματα Περιήγησης

Ο σκοπός ενός προγράμματος περιήγησης (Chrome, Edge, Firefox, Safari) είναι να διαβάσει 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)

Δομή μιας σελίδας HTML

Μια σελίδα HTML πρέπει να είναι δομημένη ως εξής:

```
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

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+
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

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>
```

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 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**.

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 .

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

```

```

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

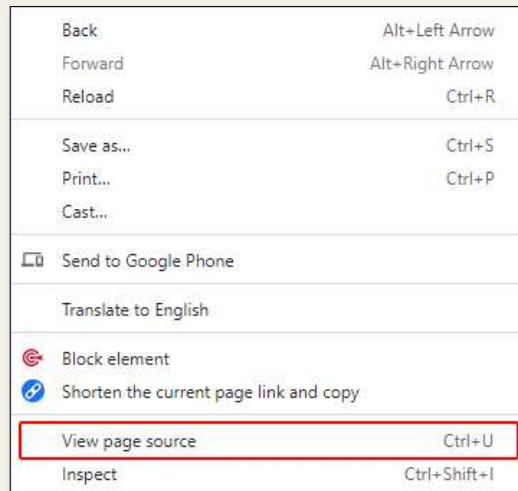
Πώς να προβάλετε μια πηγή HTML;

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.



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 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:
- `Visit W3Schools`
- **src** – η ετικέτα `` 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:

```

```

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.

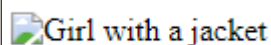
- **χαρακτηριστικά πλάτους και ύψους** – η ετικέτα `` πρέπει επίσης να περιλαμβάνει τα χαρακτηριστικά πλάτους και ύψους, τα οποία καθορίζουν το πλάτος και το ύψος της εικόνας (σε pixel):

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

- **alt** – το απαιτούμενο χαρακτηριστικό alt για την ετικέτα `` 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:

<pre><!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></pre>	<h2>The style Attribute</h2> <p>The style attribute is used to add styles to an element, such as color:</p> <p>This is a red paragraph.</p>
---	---

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

- **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 χώρων 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**.

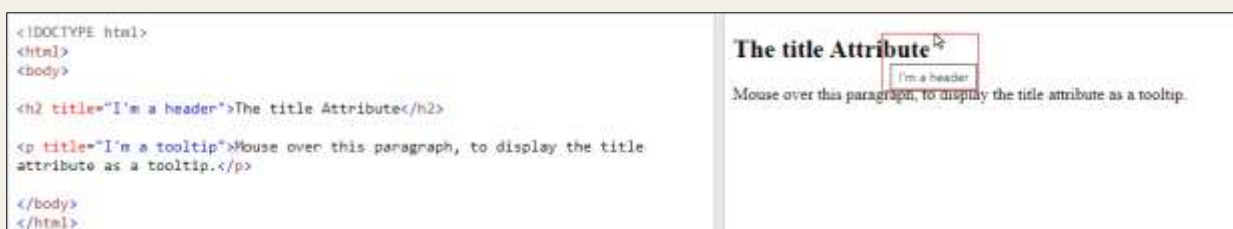
Το ακόλουθο παράδειγμα προσδιορίζει τα πορτογαλικά ως γλώσσα και την Πορτογαλία ως χώρα:

```
<!DOCTYPE
html > html lang =
en" 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:



Εικόνα 15 – Κώδικας για την εμφάνιση ενός εργαλείου «title» (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.

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

<h1>Code4SP 1</h1>
<h2>Code4SP 2</h2>
<h3>Code4SP 3</h3>
<h4>Code4SP 4</h4>
<h5>Code4SP 5</h5>
<h6>Code4SP 6</h6>

</body>
</html>
```

Code4SP 1
Code4SP 2
Code4SP 3
Code4SP 4
Code4SP 5
Code4SP 6

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.

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.

<pre><!DOCTYPE html> <html> <body> <p>Code4SP helps me to code.</p> <p>I love Code4SP.</p> <p>Coding is so great!</p> </body> </html></pre>	<p>Code4SP helps me to code.</p> <p>I love Code4SP.</p> <p>Coding is so great!</p>
---	--

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

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:

<pre><!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></pre>	<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>
--	--

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.

<pre><!DOCTYPE html> <html> <body> <p>Code4SP really is
an amazing
project.</p> </body> </html></pre>	<p>Code4SP really is an amazing project.</p>
---	--

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**

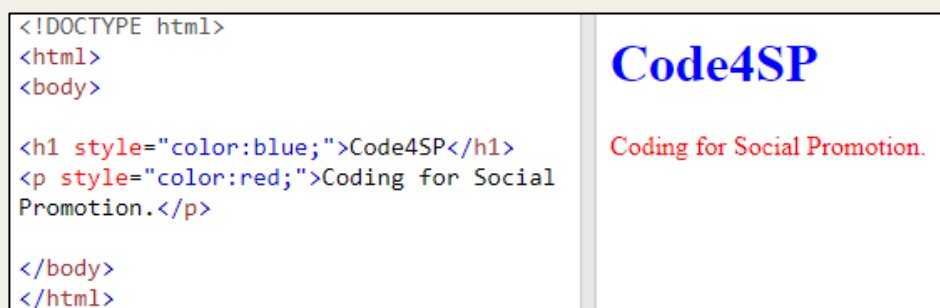
Η ιδιότητα *χρώματος* φόντου CSS καθορίζει το χρώμα φόντου για ένα στοιχείο HTML.



Εικόνα 20 – Ορισμός χρώματος φόντου (Πηγή: Συντάκτης)

- **Χρώμα κειμένου**

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



Εικόνα 21 – Ορισμός χρώματος φόντου (Πηγή: Συντάκτης)

- **Γραμματοσειρές**

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

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

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:

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

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

- **Στοίχιση Κειμένου**

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

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

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
	Έντονο κείμενο	Το στοιχείο της HTML καθορίζει το έντονο κείμενο, χωρίς καμία επιπλέον σημασία.	Έντονο 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 εναλλακτικό	<i>Πλάγιο κείμενο</i>

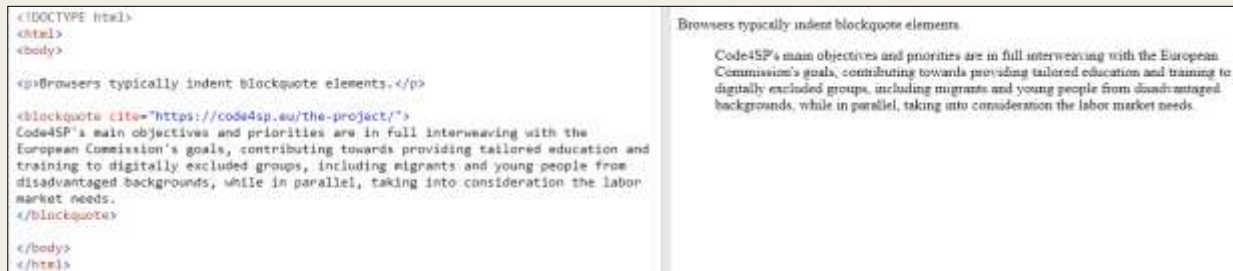
		ύφος ή διάθεση. Το περιεχόμενο στο εσωτερικό εμφανίζεται συνήθως σε πλάγια γραφή.	
<code></code>	Κείμενο με έμφαση	Το στοιχείο <code></code> της HTML ορίζει κείμενο με έμφαση. Το περιεχόμενο στο εσωτερικό εμφανίζεται συνήθως με πλάγια γραφή.	<i>Κείμενο με έμφαση</i>
<code><mark></code>	Επισημασμένο κείμενο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

		συνήθως υπογραμμίζουν το εισαγόμενο κείμενο:	
<code><sub></code>	Δείκτης κειμένου	Το στοιχείο <code><sub></code> 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 ₂ O.	Δείκτης κειμένου
<code><sup></code>	Εκθέτης κειμένου	Το στοιχείο <code><sup></code> 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

Μορφοποίηση κειμένου στην HTML

Η ετικέτα <blockquote> για παραθέσεις

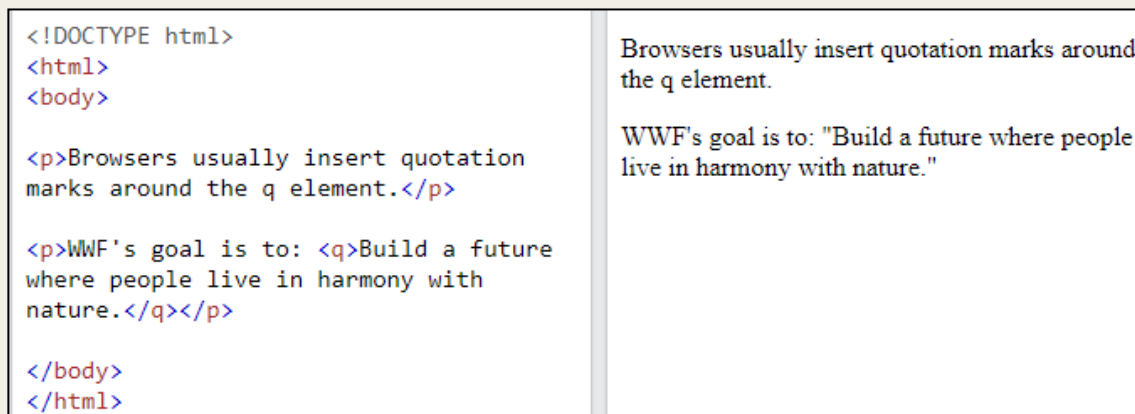
Η ετικέτα <blockquote> της HTML προσδιορίζει μια ενότητα που παρατίθεται από άλλη πηγή. Τα προγράμματα περιήγησης συνήθως εισάγουν στοιχεία <blockquote>, όπως φαίνεται παρακάτω:



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:

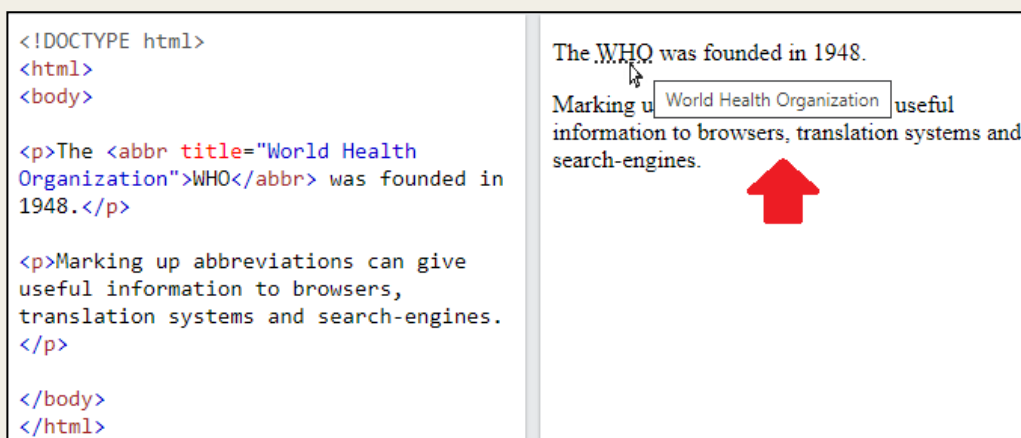


Εικόνα 26 – Το στοιχείο του σύντομου παραθέματος (Πηγή: https://www.w3schools.com/html/html_quotation_elements.asp)

Η 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:

<pre><!DOCTYPE html> <html> <body> <p>Please contact us:.</p> <address> https://code4sp.eu/
 https://www.facebook.com/Code4SP
 </address> </body> </html></pre>	<p>Please contact us:.</p> <p><i>https://code4sp.eu/</i> <i>https://www.facebook.com/Code4SP</i></p>
--	--

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:

<pre><!DOCTYPE html> <html> <body> <p><cite>The Scream</cite> by Edvard Munch, 1893.</p> </body> </html></pre>	 <p><i>The Scream</i> by Edvard Munch, 1893.</p>
---	--

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

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, ltr – για αριστερά προς τα δεξιά).

<pre><!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></pre>	<p>If your browser supports bi-directional override (bdo), the next line will be written from right to left (rtl):</p> <p>PS4edoC</p>
--	---

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:

```
<!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 σχολίων (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.

<pre><!DOCTYPE html> <html> <body> <p>Code4SP project.</p> <!-- <p>This content is hidden. </p> -- > <p>But this will appear.</p> </body> </html></pre>	<p>Code4SP project.</p> <p>But this will appear.</p>
---	--

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.



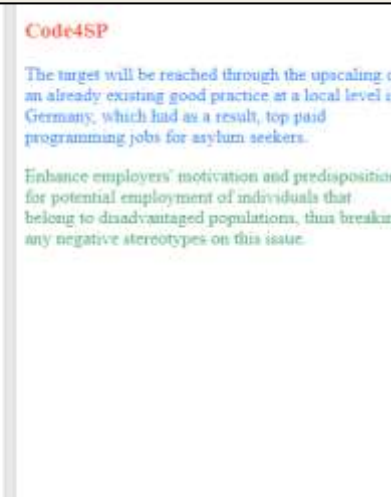
Εικόνα 33 – Ορισμένα ονόματα χρωμάτων που μπορούν να οριστούν (Πηγή: https://www.w3schools.com/html/html_colors.asp)

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

<pre><!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></pre>	
---	--

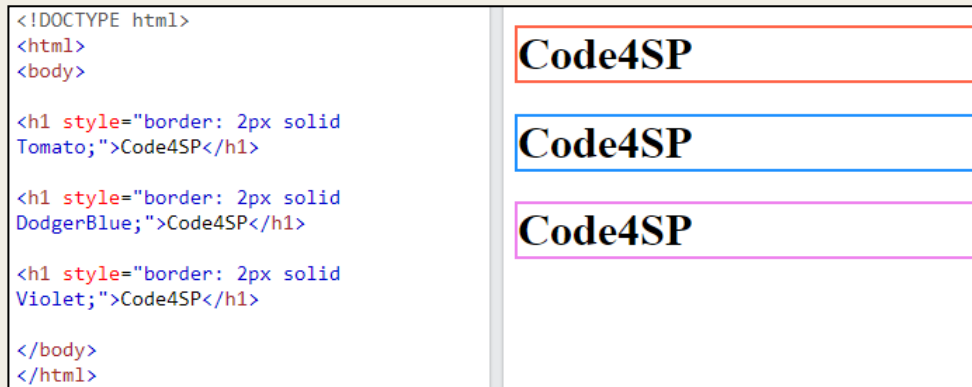
Εικόνα 34 – Καθορισμός του χρώματος φόντου μιας ιστοσελίδας (Πηγή:Συντάκτης)

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

<pre><!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></pre>	
---	---

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

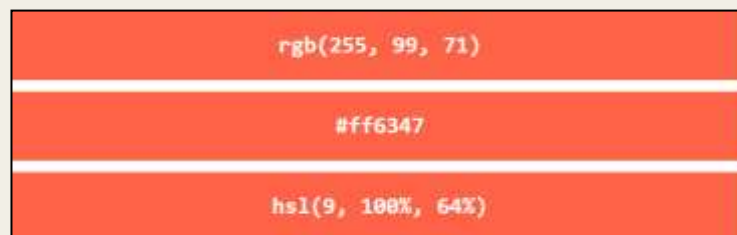
Επίσης, για το χρώμα των περιγραμμάτων:



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

Τιμές χρωμάτων

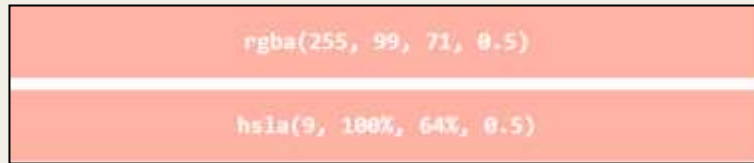
Τα χρώματα στην HTML ορίζονται με προκαθορισμένα ονόματα χρωμάτων ή με τιμές RGB, HEX, HSL, RGBA ή HSLA. Στα ακόλουθα τρία στοιχεία `<div>` το χρώμα φόντου τους καθορίστηκε με τις τιμές RGB, HEX και HSL:



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:

<pre><!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></pre>	<p>Same as color name "Tomato":</p> <p>rgb(255, 99, 71)</p> <p>#ff6347</p> <p>hsl(9, 100%, 64%)</p> <p>Same as color name "Tomato", but 50% transparent:</p> <p>rgba(255, 99, 71, 0.5)</p> <p>hsla(9, 100%, 64%, 0.5)</p> <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>
---	---

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.

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:

- **_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 **** 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`)
- 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 `` 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.



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

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 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 **<source>**, καθένα από τα οποία αναφέρεται σε διαφορετική πηγή εικόνας και ένα στοιχείο **** στο τέλος. Ομοίως, κάθε στοιχείο **<source>** 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

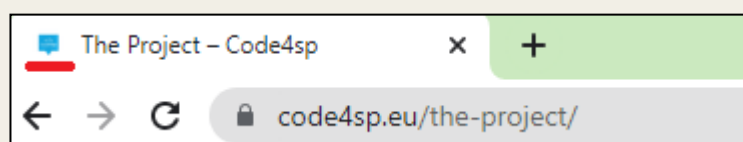
Ένας χάρτης εικόνων μας επιτρέπει να ορίσουμε τα κομβικά σημεία (hotspot) σε μια εικόνα που λειτουργεί ακριβώς όπως ένας υπερσύνδεσμος. Η βασική ιδέα πίσω από τη δημιουργία ενός χάρτη εικόνων είναι να δώσουμε έναν απλό τρόπο σύνδεσης διαφόρων τμημάτων μιας εικόνας χωρίς να τη διαιρέσουμε σε ξεχωριστά 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](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 **** 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:



Εικόνα 42 – Favicon 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>
```

Πίνακες στην HTML

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`,

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>

```

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:

```

<!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>)

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.tutorialpublic.com/html-tutorial/html-tables.php>)

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:

- **A) Μη Διατεταγμένες λίστες** — Χρησιμοποιείται για τη δημιουργία λίστας σχετικών στοιχείων, χωρίς συγκεκριμένη σειρά.
- **B) 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.

A) Μη Διατεταγμένες Λίστες

Μια Μη Διατεταγμένη Λίστα δημιουργείται χρησιμοποιώντας το στοιχείο `` και κάθε στοιχείο της λίστας ξεκινά με το στοιχείο ``. 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](#))

B) Διατεταγμένες Λίστες

Μια Διατεταγμένη Λίστα δημιουργείται χρησιμοποιώντας το στοιχείο `` στοιχείο και κάθε στοιχείο της λίστας ξεκινά με το στοιχείο `` στοιχείο. Οι διατεταγμένες λίστες χρησιμοποιούνται όταν η σειρά των στοιχείων της λίστας παίζει καθοριστικό ρόλο. Σημειώνεται με αριθμούς, ως εξής:

<pre> <!DOCTYPE html> <html lang="en"> <head> <title>HTML Ordered list</title> </head> <body> <h2>How to cook your own veggie burger</h2> 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. 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. 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 those nice grill marks.) Cook them until your desired doneness (around 125-130°F for medium rare, around 1 minute per side for each inch of thickness). But before you take them off the grill... ...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. Once your burgers are 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. </body> </html> </pre>	<h3>How to cook your own veggie burger</h3> <ol style="list-style-type: none"> 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 those nice grill marks.) Cook them until your desired doneness (around 125-130°F 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 are 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.

<pre> <!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> </pre>	<h3>What are bread and coffee?</h3> <p>Bread A baked food made of flour.</p> <p>Coffee A drink made from roasted coffee beans.</p>
--	--

Εικόνα 48 – Λίστες περιγραφών (Πηγή: Συντάκτης)

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>

```

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](#)).

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>`.

<pre><!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></pre>	<p>Login: <input type="text"/></p>
--	------------------------------------

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`.

<pre><!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></pre>	<p>Password: <input type="password"/></p>
--	---

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

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>
```

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

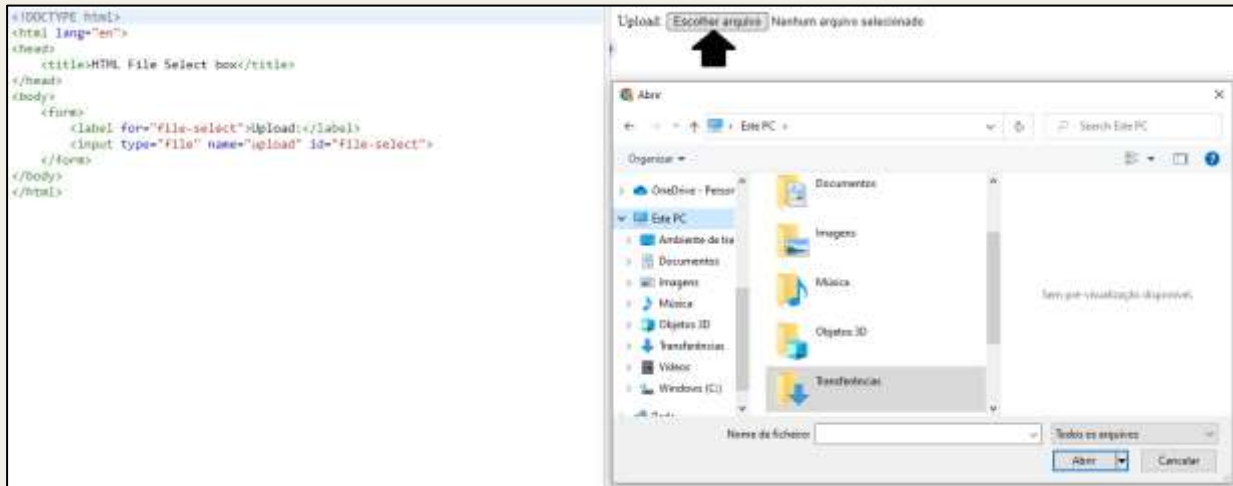
Τα **πλαίσια ελέγχου** παρέχουν στον χρήστη μία ή περισσότερες επιλογές από ένα προκαθορισμένο σύνολο επιλογών. Δημιουργείται χρησιμοποιώντας ένα στοιχείο `<input>` του οποίου το χαρακτηριστικό `type` έχει την τιμή του `checkbox`. Αν κάποιος προτιμά να δημιουργήσει ένα πλαίσιο ελέγχου (ή κουμπί επιλογής) 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>
```

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

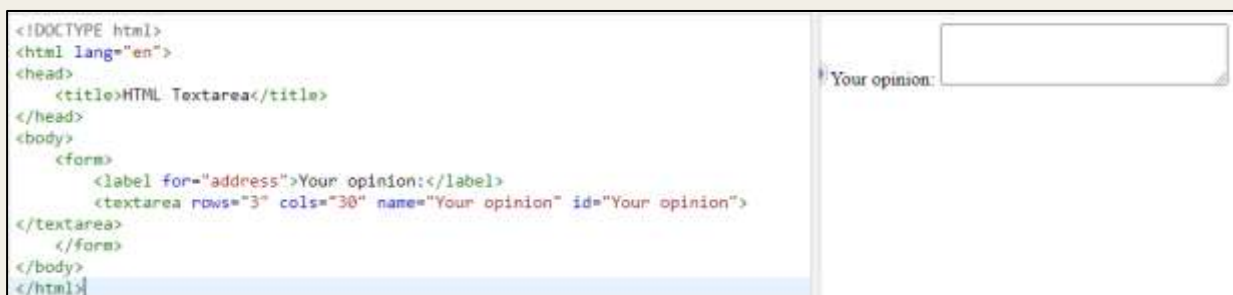
Τα πλαίσια επιλογής αρχείου επιτρέπουν σε έναν χρήστη να αναζητήσει ένα τοπικό αρχείο και να το στείλει ως συνημμένο με τα δεδομένα της φόρμας. Π.χ., το Google Chrome παρέχει ένα πεδίο επιλογής για την εισαγωγή αρχείου με το κουμπί «Αναζήτηση» που επιτρέπει στους χρήστες να επιλέξουν ένα αρχείο από τον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

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>

```

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 υποβολής, 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](#).

```

<!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:

```

<!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)

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

<u>Χαρακτηριστικό</u>	<u>Περιγραφή</u>
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 <code>_blank</code> , <code>_self</code> , <code>_parent</code> και <code>_top</code> .
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 <code>method</code> είναι <code>post</code> .

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](#).

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`.



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)

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 >` , `< 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, συνιστάται 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)

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.
- 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>
```



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

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.

<pre><!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></pre>	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.

<pre><!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></pre>	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 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)

Το στοιχείο script στην HTML

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



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>`.

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 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	>	>
&	συμπλεκτικό σύμβολο	&	&
"	Εισαγωγικά	"	"
'	απόστροφος	'	'
¢	σεντ	¢	¢
£	λίρα	£	£
¥	ιαπωνικό γιεν	¥	¥
€	ευρώ	€	€
©	πνευματικά δικαιώματα	©	©
®	σήμα κατατεθέν	®	®
™	εμπορικό σήμα	™	™

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

οποιοδήποτε 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 Resource Locator (διεύθυνση ιστοσελίδας) και λειτουργεί ως παγκόσμια διεύθυνση εγγράφων και άλλων πόρων στον Παγκόσμιο Ιστό. 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.
- **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.

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>)

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 κώδικες 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 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.

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.

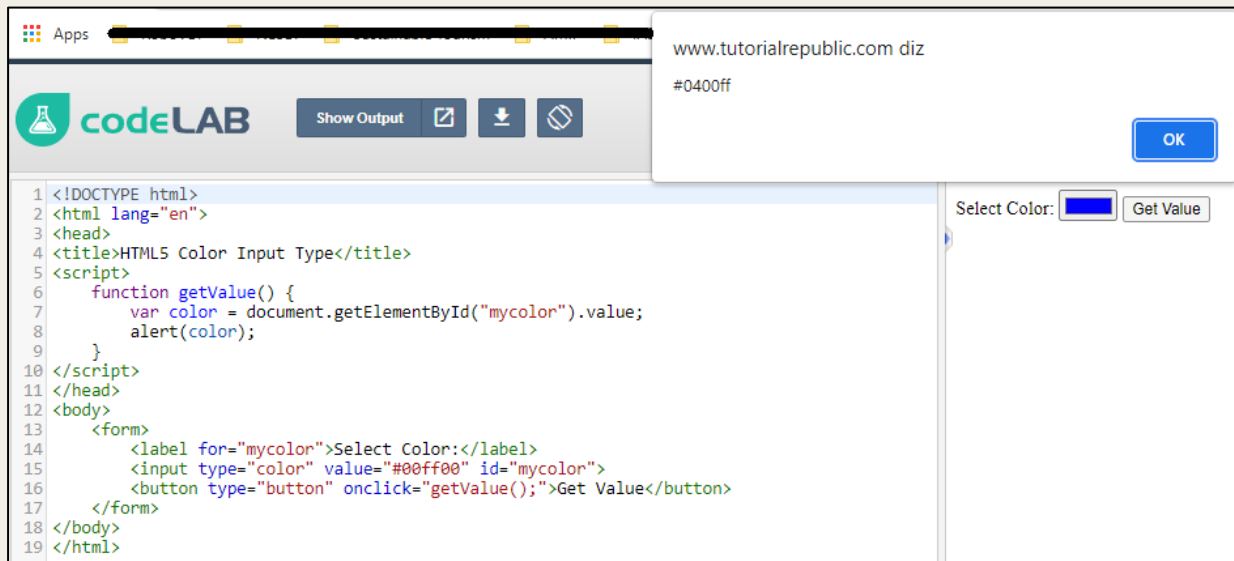
Ακολουθούν ορισμένοι νέοι τύποι εισαγωγής:

- 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.

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](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.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Date Input Type</title>
<script>
function getValue() {
var date = document.getElementById("mydate").value;
alert(date);
}
</script>
</head>
<body>
<form>
<label for="mydate">Select Date:</label>
<input type="date" value="2019-04-15" id="mydate">
<button type="button" onclick="getValue();">Get Value</button>
</form>
</body>
</html>
```

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.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Datetime-Local Input Type</title>
<script>
function getValue() {
var datetime = document.getElementById("mydatetime").value;
alert(datetime);
}
</script>
</head>
<body>
<form>
<label for="mydatetime">Choose Date and Time:</label>
<input type="datetime-local" id="mydatetime">
<button type="button" onclick="getValue();">Get Value</button>
</form>
</body>
</html>
```

Εικόνα 71 – Ο τύπος εισαγωγής τοπικής ημερομηνίας και ώρας (Πηγή:

<https://www.tutorialrepublic.com/codelab.php?topic=html5&file= color-input-type>)

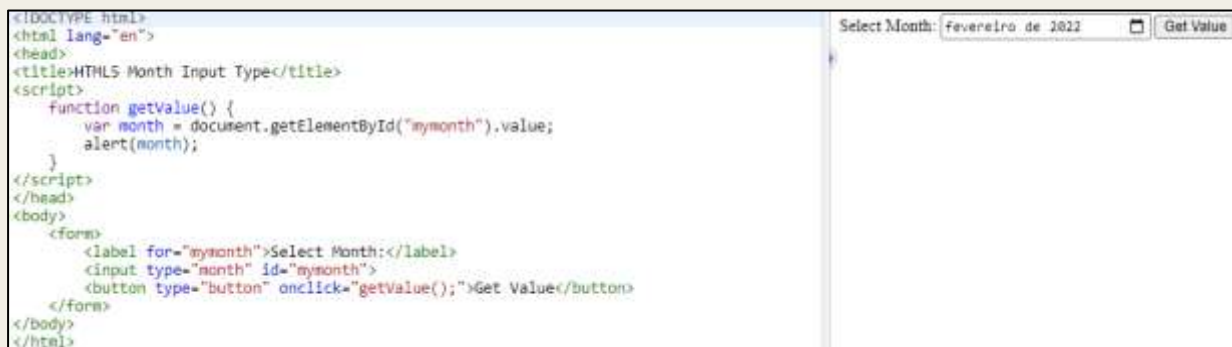
Ο καλύτερος τύπος εισαγωγής για να εισάγουν οι χρήστες τη διεύθυνση ηλεκτρονικού ταχυδρομείου τους, είναι ο τύπος εισαγωγής **email**. Είναι σαν ένας συνηθισμένος τύπος εισαγωγής κειμένου, αλλά εάν εφαρμοστεί σε συνδυασμό με το χαρακτηριστικό **required**, τα προγράμματα περιήγησης μπορούν να αναζητήσουν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



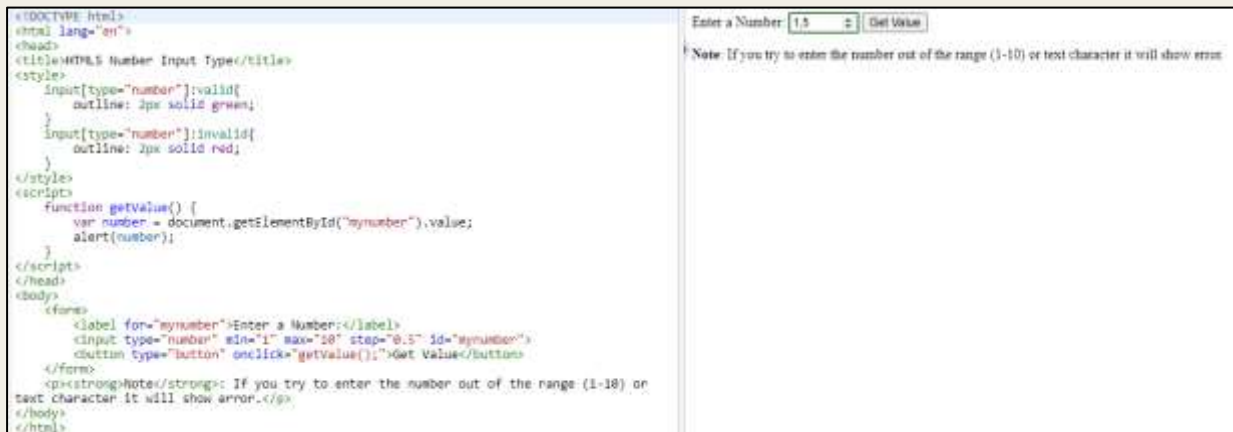
Εικόνα 72 – Ο τύπος εισαγωγής email (Πηγή: <https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php>)

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.



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

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.



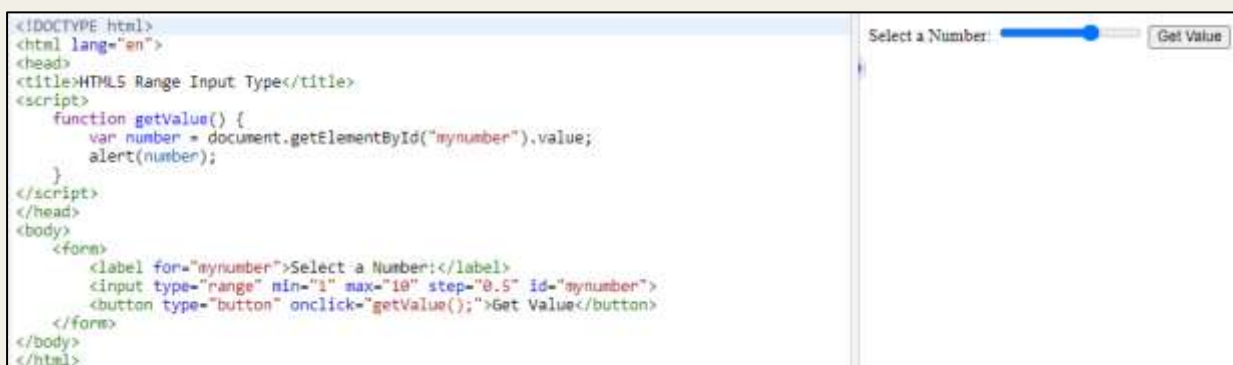
```

<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 Number Input Types</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 Types</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>)

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



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.



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

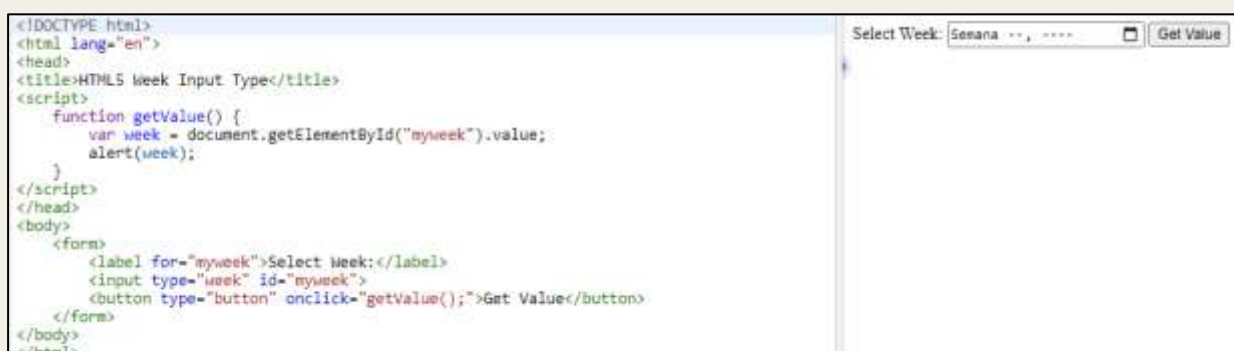
Όσον αφορά τον τύπο εισαγωγής **διεύθυνσης url (url input)**, μπορεί να χρησιμοποιηθεί για την εισαγωγή διευθύνσεων URL ή ιστοσελίδων. Το χαρακτηριστικό **multiple** μπορεί να χρησιμοποιηθεί για την εισαγωγή περισσότερων από μία διευθύνσεων URL. Επιπλέον, εάν το χαρακτηριστικό **required** δεν δύναται να χρησιμοποιηθεί, το

πρόγραμμα περιήγησης θα εκτελέσει αυτόματα επικύρωση για να διασφαλίσει ότι μόνο 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.



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.



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

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:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <title>Embedding Canvas Into HTML Pages</title>
6 <style>
7   canvas {
8     border: 1px solid #000;
9   }
10 </style>
11 <script>
12   window.onload = function() {
13     var canvas = document.getElementById("myCanvas");
14     var context = canvas.getContext("2d");
15     // draw stuff here
16   };
17 </script>
18 </head>
19 <body>
20   <canvas id="myCanvas" width="300" height="200"></canvas>
21 </body>
22 </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:

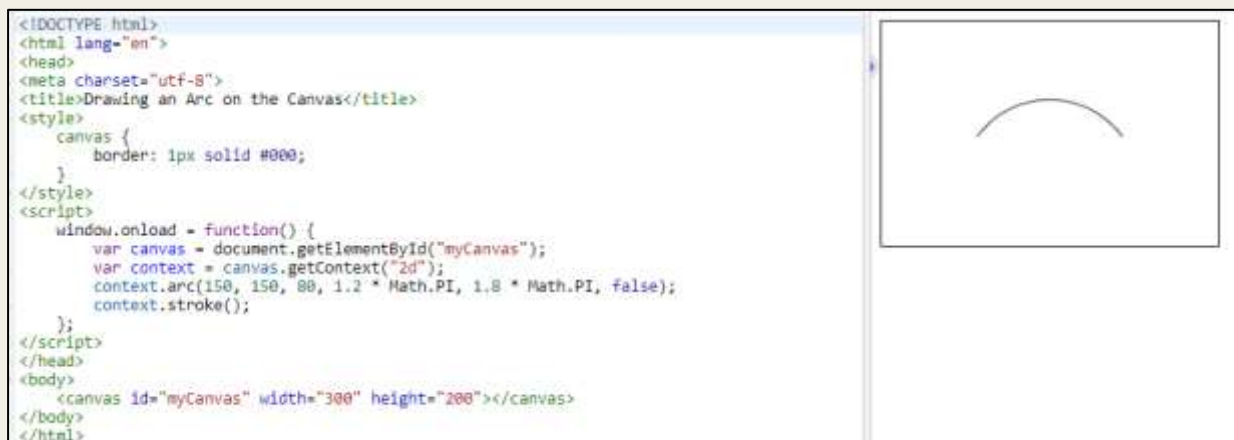


Εικόνα 82 – Οι διαδικασίες `moveTo()`, `lineTo()` και `stroke()` (Πηγή: <https://www.tutorialrepublic.com/html-tutorial/html5-new-input-types.php>)

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:



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:

```
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);
```



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

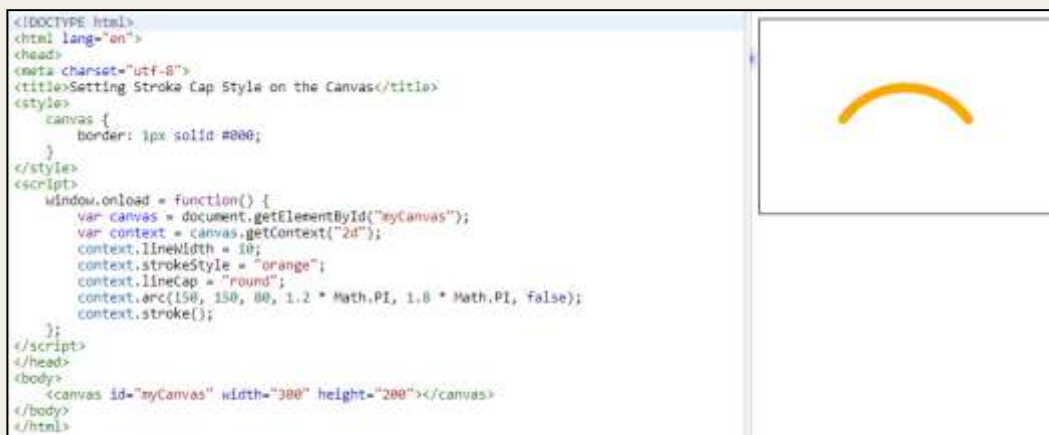
Όσον αφορά το **στυλ και το χρώμα σχεδίασης**, το προεπιλεγμένο χρώμα είναι το μαύρο, με πάχος 1 pixel. Ωστόσο, αυτά τα χαρακτηριστικά μπορούν να αλλάξουν χρησιμοποιώντας τις ιδιότητες **strokeStyle** και **lineWidth**, όπως φαίνεται στην *Εικόνα 86*.

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.



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>)



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



Εικόνα 88 – Αρχιεπιθέση 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.

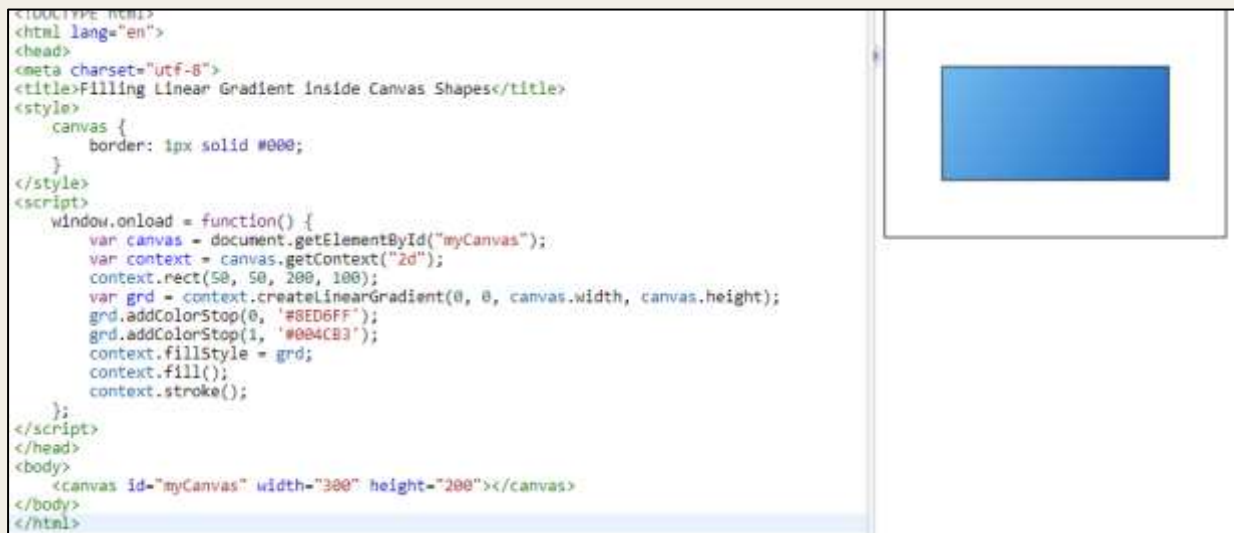
Η βασική σύνταξη για τη δημιουργία μιας **γραμμικής διαβάθμισης** είναι:

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

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

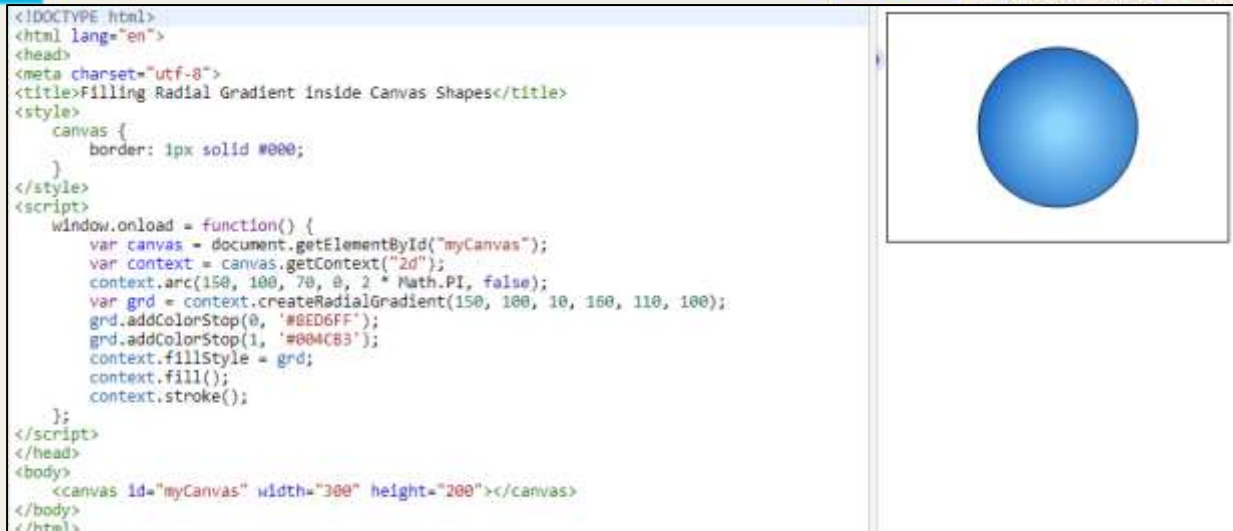
```
var grd = context.createRadialGradient (startX, startY, startRadius, endX, endY, endRadius);
```

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



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()** :



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 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.



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).

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.



```

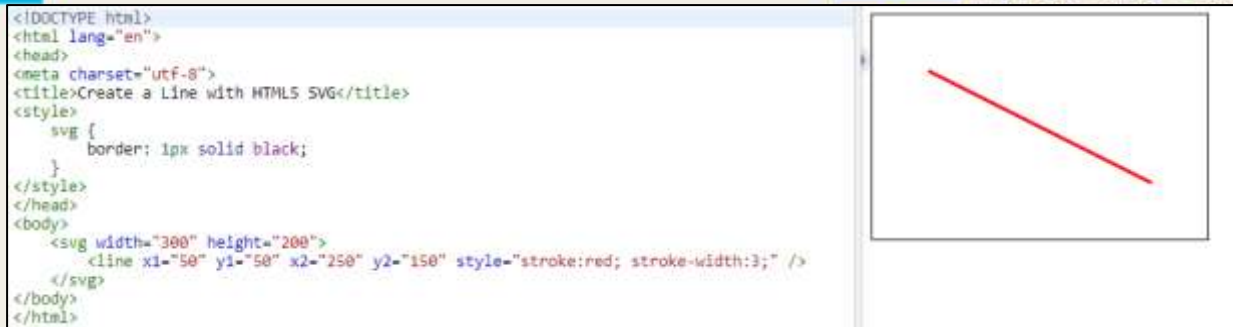
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Embedding SVG Into HTML Pages</title>
<style>
  svg {
    border: 1px solid black;
  }
</style>
</head>
<body>
  <svg width="300" height="200">
    <text x="10" y="20" style="font-size:14px;">
      Your browser support SVG.
    </text>
    <text x="10" y="30">
      Sorry, your browser does not support SVG.
    </text>
  </svg>
</body>
</html>

```

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)`.



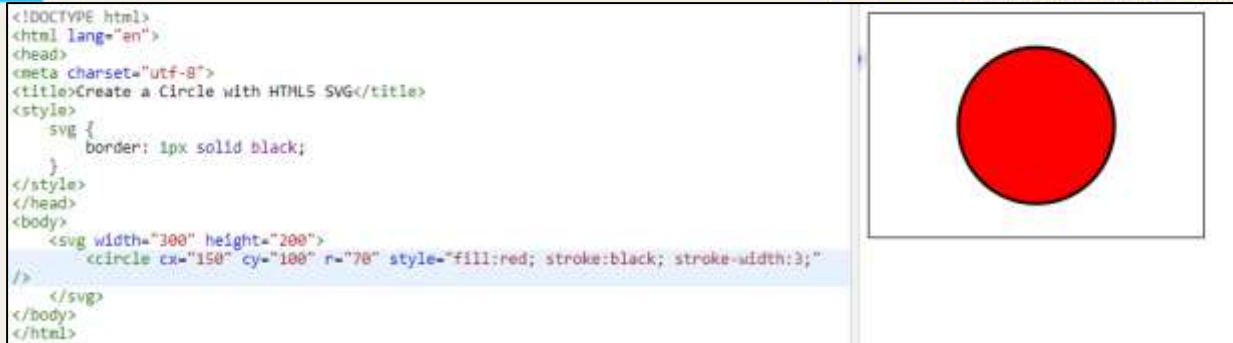
Εικόνα 93 – Σχεδιάζοντας μια ευθεία γραμμή με το στοιχείο `<line>` σε SVG (Πηγή: <https://www.tutorialrepublic.com/html-tutorial/html5-svg.php>)

Ο καταλληλότερος τρόπος για τον σχεδιασμό **ορθογωνίων** και τετραγώνων, είναι το στοιχείο `<rect>` σε SVG. Τα χαρακτηριστικά `x` και `y` του στοιχείου `<rect>` καθορίζουν τις συντεταγμένες της επάνω αριστερής γωνίας του ορθογωνίου. Τα χαρακτηριστικά `width` και `height` καθορίζουν το πλάτος και το ύψος του σχήματος.



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).



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 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.



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 `` για να μορφοποιήσετε το κείμενο που περιλαμβάνεται σε ένα στοιχείο `<text>`. Το κείμενο περιλαμβάνεται σε ξεχωριστά `span`, 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 επιλεγθούν ταυτόχρονα.



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

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 rixel)
Πολλά γραφικά στοιχεία, τα οποία γίνονται μέρος του δέντρου DOM της σελίδας	<u>Μεμονωμένο στοιχείο με παρόμοια λειτουργία με το στοιχείο <code></code>.</u>

SVG	Canvas
	<u>Το διάγραμμα canvas μπορεί να αποθηκευτεί σε μορφή PNG ή JPG</u>
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 rixelation (παραμόρφωση των εικονοστοιχείων)	Μικρή δυνατότητα κλιμάκωσης. Δεν είναι κατάλληλο για εκτύπωση σε υψηλότερη ανάλυση. Ενδέχεται να προκύψει rixelation (παραμόρφωση των εικονοστοιχείων)

Εικόνα 96 – Οι διαφορές μεταξύ SVG και Canvas (Πηγή: <https://www.tutorialrepublic.com/html-tutorial/html5-svg.php>)

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 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.

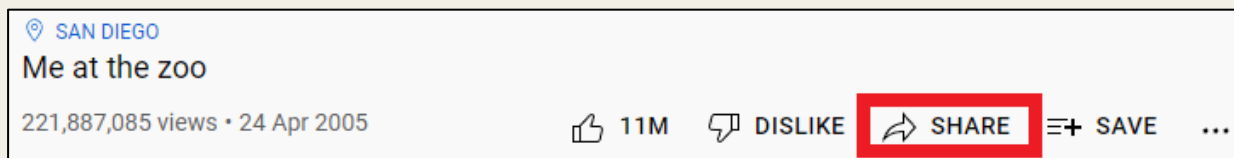
Το στοιχείο `<video>`, το οποίο εισήχθη πρόσφατα λειτουργεί στα περισσότερα από τα σύγχρονα προγράμματα περιήγησης. [Αυτό το παράδειγμα](#) εξηγεί πώς μπορείτε να εισαγάγετε ένα βίντεο με απλό τρόπο σε ένα έγγραφο HTML, χρησιμοποιώντας το προεπιλεγμένο σύνολο εντολών του προγράμματος περιήγησης, με μία 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 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 96 – Κουμπί «Κοινοποίηση» 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.

```
Embed Video ✕  
  
<iframe width="560" height="315"  
src="https://www.youtube.com/embed/  
jNQXAC9IVRw" title="YouTube video  
player" frameborder="0"  
allow="accelerometer; autoplay;  
clipboard-write; encrypted-media;  
gyroscope; picture-in-picture"  
allowfullscreen></iframe>
```

Εικόνα 97 – Επιλογή «Ενσωμάτωσης βίντεο» στο YouTube (Πηγή: Συντάκτης)

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 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.

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.

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 98* παρακάτω είναι ένα παράδειγμα ενός αρχείου manifest:

Example	Download
1	CACHE MANIFEST
2	# v1.0 : 10-08-2014
3	
4	CACHE:
5	# pages
6	index.html
7	
8	# styles & scripts
9	css/theme.css
10	js/jquery.min.js
11	js/default.js
12	
13	# images
14	/favicon.ico
15	images/logo.png
16	
17	NETWORK:
18	login.php
19	
20	FALLBACK:
21	/ /offline.html

Εικόνα 98 – 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.

- 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.

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 99* 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 99 – 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 100 – *about_blank#blocked* (Πηγή: <https://www.tutorialrepublic.com/html-tutorial/html5-application-cache.php>)

Web Workers στην HTML5

Αυτή η εποενοότητα 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 οποιοσδήποτε 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 συνεχίζοντας 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 101*:

```

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 101 – 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.

```

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>

```

Εικόνα 102 – 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:

<pre> 1 <!DOCTYPE html> 2 <html lang="en"> 3 <head> 4 <meta charset="utf-8"> 5 <title>Start/Stop Web Worker in HTML5</title> 6 <script> 7 // Set up global variable 8 var worker; 9 10 function startWorker() { 11 // Initialize web worker 12 worker = new Worker("/examples/js/worker.js"); 13 14 // Run update function, when we get a message from worker 15 worker.onmessage = update; 16 17 // Tell worker to get started 18 worker.postMessage("start"); 19 } 20 21 function update(event) { 22 // Update the page with current message from worker 23 document.getElementById("result").innerHTML = event.data; 24 } 25 26 function stopWorker() { 27 // Stop the worker 28 worker.terminate(); 29 } 30 </script> 31 </head> </pre>	<h3>Web Worker Demo</h3> <p> <input type="button" value="Start web worker"/> <input type="button" value="Stop web worker"/> </p>
---	--

Εικόνα 103 – 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 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 PHP¹ 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:

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.

```
1 <?php
2 header("Content-Type: text/event-stream");
3 header("Cache-Control: no-cache");
4
5 // Get the current time on server
6 $currentTime = date("h:i:s", time());
7
8 // Send it in a message
9 echo "data: " . $currentTime . "\n\n";
10 flush();
11 ?>
```

Εικόνα 104 – 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.

Τώρα όσον αφορά **στον τρόπο επεξεργασίας μηνυμάτων σε μια ιστοσελίδα**, το αντικείμενο **EventSource** 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.html**» και στη συνέχεια θα τοποθετηθεί στον ίδιο κατάλογο του έργου όπου βρίσκεται το «**server_time.php**».


```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <title>Using Server-Sent Events</title>
5 <script>
6     window.onload = function() {
7         var source = new EventSource("server_time.php");
8         source.onmessage = function(event) {
9             document.getElementById("result").innerHTML += "New time received
from web server: " + event.data + "<br>";
10        };
11    };
12 </script>
13 </head>
14 <body>
15     <div id="result">
16         <!--Server response will be inserted here-->
17     </div>
18 </body>
19 </html>

```

Εικόνα 105 – 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, 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 επιτρέπει 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.

<pre><!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></pre>	<p>Your current position is (Latitude: 41.0079509, Longitude: -8.6270736)</p> <p>Show Position</p>
--	--

Εικόνα 106 – 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.

```
<!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 106 – 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 `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 ιστοτόπου θα πρέπει να αποτρέψει τον προεπιλεγμένο χειρισμό του στοιχείου, με τη λειτουργία **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.

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).