



Co-funded by the  
Erasmus+ Programme  
of the European Union



# Εκπαιδευτικό Υλικό JavaScript Υποενότητα 3 – JS & BOM

WP3: Εκπαιδευτικό Υλικό του Έργου Code4SP

Εκπονήθηκε από:



CITIZENS  
IN POWER



Center for Social  
Innovation



ZAUG  
gGmbH



Co-funded by the  
Erasmus+ Programme  
of the European Union





# JS & BOM



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



# Το Μοντέλο Αντικειμένου του Προγράμματος Περιήγησης (BOM)

Το Browser Object Model (BOM) επιτρέπει στην JavaScript να «μιλήσει» στο πρόγραμμα περιήγησης.

Δεν υπάρχουν επίσημα πρότυπα για το Browser Object Model (BOM).

Γεδομένου ότι τα σύγχρονα προγράμματα περιήγησης έχουν εφαρμόσει (σχεδόν) τις ίδιες μεθόδους και ιδιότητες για τη διαδραστικότητα της JavaScript, συχνά αυτά αναφέρονται και ως μέθοδοι και ιδιότητες του BOM.



## Το Αντικείμενο Window

Το αντικείμενο window υποστηρίζεται από όλα τα προγράμματα περιήγησης. Αντιπροσωπεύει το παράθυρο του προγράμματος περιήγησης. Όλα τα παγκόσμια αντικείμενα JavaScript, οι λειτουργίες και οι μεταβλητές γίνονται αυτόματα μέλη του αντικειμένου window. Οι καθολικές μεταβλητές είναι ιδιότητες του αντικειμένου window. Οι καθολικές λειτουργίες είναι μέθοδοι του αντικειμένου window.

Ακόμα και το αντικείμενο εγγράφου (του HTML DOM) είναι μια ιδιότητα του αντικειμένου window:

```
window.document.getElementById("header");
```

## Μέγεθος Window

Δύο συνολικά ιδιότητες μπορούν να χρησιμοποιηθούν για τον προσδιορισμό του μεγέθους window του προγράμματος περιήγησης. Και οι δύο ιδιότητες επιστρέφουν σε εικονοστοιχεία:

- `window.innerHeight` - το εσωτερικό ύψος του window του προγράμματος περιήγησης (σε εικονοστοιχεία)
- `window.innerWidth` - το εσωτερικό πλάτος του παραθύρου του προγράμματος περιήγησης (σε εικονοστοιχεία)

Το window του προγράμματος περιήγησης (η οπτική γωνία/ viewport) δεν περιέχει εικονίδια γραμμής εργαλείων ή γραμμής κύλισης (toolbars and scrollbars).

```
let w = window.innerWidth;  
let h = window.innerHeight;
```

## Οθόνη Window

Το αντικείμενο `window.screen` μπορεί να γραφτεί χωρίς το πρόθεμα του `window`.

Ιδιότητες:

- `screen.width`
- `screen.height`
- `screen.availWidth`
- `screen.availHeight`
- `screen.colorDepth`
- `screen.pixelDepth`



## Τοποθεσία window στην JavaScript (Location)

Το αντικείμενο `window.location` μπορεί να χρησιμοποιηθεί για τη λήψη της τρέχουσας διεύθυνσης σελίδας (URL) και για την ανακατεύθυνση του προγράμματος περιήγησης σε μια νέα σελίδα. Το αντικείμενο `window.screen` μπορεί να γραφτεί χωρίς το πρόθεμα του `window`.

Το αντικείμενο `window.screen` μπορεί να γραφτεί χωρίς το πρόθεμα του `window`.  
Μερικά παραδείγματα:

- Η ιδιότητα `window.location.href` επιστρέφει το `href` (URL) της τρέχουσας σελίδας
- Η ιδιότητα `window.location.hostname` επιστρέφει το όνομα τομέα (domain name) της φιλοξενίας ιστοσελίδων (web host)





## Ιδιότητα Window Location στην JavaScript

- Η ιδιότητα `window.location.pathname` επιστρέφει τη διαδρομή (path) και το όνομα αρχείου (filename) της τρέχουσας σελίδας
- Η ιδιότητα `window.location.protocol` επιστρέφει το πρωτόκολλο ιστού που χρησιμοποιήθηκε (http: ή https:)
- Η ιδιότητα `window.location.assign()` φορτώνει ένα νέο έγγραφο



## Ιστορικό Window στην JavaScript

Το αντικείμενο `window.screen` μπορεί να γραφτεί χωρίς το πρόθεμα του `window`. Για την προστασία του απορρήτου των χρηστών, υπάρχουν περιορισμοί στον τρόπο με τον οποίο η JavaScript μπορεί να έχει πρόσβαση σε αυτό το αντικείμενο. Μερικές μέθοδοι:

- `history.back()` - το ίδιο με το να κάνετε κλικ πίσω στο πρόγραμμα περιήγησης
- `history.forward()` - το ίδιο με το να κάνετε κλικ προς τα εμπρός στο πρόγραμμα περιήγησης

# Η μέθοδος history.back()

Η μέθοδος `history.back()` φορτώνει την προηγούμενη διεύθυνση URL στη λίστα ιστορικού. Αυτό είναι το ίδιο με το να κάνετε κλικ στο κουμπί πίσω στο πρόγραμμα περιήγησης.

## Example

Create a back button on a page:

```
<html>
<head>
<script>
function goBack() {
  window.history.back()
}
</script>
</head>
<body>

<input type="button" value="Back" onclick="goBack()">

</body>
</html>
```

The output of the code above will be:

Back

## Πλοηγός Window (Navigator)

Το αντικείμενο `window.navigator` περιέχει πληροφορίες σχετικά με το πρόγραμμα περιήγησης του επισκέπτη. Το αντικείμενο `window.navigator` μπορεί να γραφτεί χωρίς πρόθεμα. Μερικά παραδείγματα:

- `navigator.appName`
- `navigator.appCodeName`
- `navigator.platform`

```
document.getElementById("demo").innerHTML =  
"navigator.appName is " + navigator.appName;
```

# Αναδυόμενα πλαίσια JavaScript (Popup Boxes)

## Πλαίσιο Ειδοποίησης (Alert Box)

Το πλαίσιο ειδοποίησης χρησιμοποιείται συχνά όταν θέλετε να βεβαιωθείτε ότι οι πληροφορίες φτάνουν στο χρήστη. Όταν αναδυθεί ένα πλαίσιο ειδοποίησης, ο χρήστης θα πρέπει να κάνει κλικ στο "ΟΚ" για να συνεχίσει. Η μέθοδος `window.alert()` μπορεί να γραφτεί χωρίς το πρόθεμα `window`.

```
alert("I am an alert box!");
```

## Επιβεβαίωση Πλαισίου (Confirm Box)

Μπορείτε να χρησιμοποιήσετε το πλαίσιο επιβεβαίωσης σε περίπτωση που θέλετε ο χρήστης να επαληθεύσει ή να αποδεχτεί κάτι. Όταν εμφανιστεί ένα πλαίσιο επιβεβαίωσης, ο χρήστης θα πρέπει να κάνει κλικ είτε στο “OK” είτε στο “CANCEL” για να συνεχίσει. Εάν ο χρήστης κάνει κλικ στο “OK”, το πλαίσιο επιστρέφει την τιμή «αληθές». Εάν ο χρήστης κάνει κλικ στην επιλογή «Ακύρωση», τότε το πλαίσιο επιστρέφει την τιμή «ψευδές». Η μέθοδος `window.confirm()` μπορεί να γραφτεί χωρίς πρόθεμα.

```
if (confirm("Press a button!")) {  
    txt = "You pressed OK!";  
} else {  
    txt = "You pressed Cancel!";  
}
```

## Πλαίσιο Προτροπής (Prompt Box)

Μπορείτε να χρησιμοποιήσετε ένα πλαίσιο προτροπής όταν θέλετε ο χρήστης να εισάγει μια τιμή πριν από την είσοδο σε μια σελίδα. Όταν εμφανιστεί ένα πλαίσιο προτροπής, ο χρήστης θα πρέπει να κάνει κλικ είτε στο “OK” είτε στο “CANCEL” για να συνεχίσει. Εάν ο χρήστης κάνει κλικ στο “OK”, το πλαίσιο επιστρέφει την τιμή «αληθές». Εάν ο χρήστης κάνει κλικ στην επιλογή “CANCEL”, το πλαίσιο επιστρέφει την τιμή «ψευδές».

```
let person = prompt("Please enter your name", "Harry Potter");
let text;
if (person == null || person == "") {
  text = "User cancelled the prompt.";
} else {
  text = "Hello " + person + "! How are you today?";
}
```

# Συμβάντα Συγχρονισμού JavaScript (Timing Events)

Το αντικείμενο window επιτρέπει την εκτέλεση του κώδικα σε καθορισμένα χρονικά διαστήματα. Αυτά τα χρονικά διαστήματα ονομάζονται συμβάντα συγχρονισμού. Οι δύο βασικές μέθοδοι που χρησιμοποιούνται με το JavaScript είναι:

- `setTimeout(function,milliseconds):`

Εκτελεί μια συνάρτηση, μετά από αναμονή ενός καθορισμένου αριθμού χιλιοστών του δευτερολέπτου.

- `setInterval(function,milliseconds):`

Εκτελεί το ίδιο με το `setTimeout()`, αλλά επαναλαμβάνει την εκτέλεση της συνάρτησης συνεχώς.

Το `setTimeout()` και το `setInterval()` είναι και οι δύο μέθοδοι του αντικειμένου HTML DOM Window.



# JavaScript Cookies

Τα Cookies είναι δεδομένα, αποθηκευμένα σε μικρά αρχεία κειμένου, στον υπολογιστή σας. Όταν ένας διακομιστής ιστού έχει στείλει μια ιστοσελίδα σε ένα πρόγραμμα περιήγησης, η σύνδεση τερματίζεται και ο διακομιστής ξεχνά τα πάντα για τον χρήστη. Τα Cookies εφευρέθηκαν για την επίλυση του προβλήματος «πώς να θυμάστε πληροφορίες σχετικά με τον χρήστη»:

- Όταν ένας χρήστης επισκέπτεται μια ιστοσελίδα, το όνομά του/της μπορεί να αποθηκευτεί σε ένα cookie.
- Την επόμενη φορά που ο χρήστης επισκέπτεται τη σελίδα, το cookie «θυμάται» το όνομά του/της.

Τα cookies αποθηκεύονται σε ζεύγη ονομάτων-τιμών όπως:

```
username = John Doe
```

