

Τεχνολογίες WWW

JavaScript

Δρ. Χαράλαμπος Καρανίκας

Τμήμα Μηχανικών Πληροφορικής

JavaScript -1

- Η JavaScript είναι η πιο δημοφιλής scripting γλώσσα στο WEB. Δουλεύει με όλους τους γνωστούς φυλλομετρητές (browsers), όπως Internet Explorer, Firefox, κτλ.
- Μία scripting γλώσσα είναι μια «ελαφριά» γλώσσα προγραμματισμού.
- Η JavaScript χρησιμοποιείται σε εκατομμύρια ιστοσελίδες για να προσθέσει λειτουργικότητες, να επαληθεύσει δεδομένα φορμών, να προσδιορίσει τον browsers του client(πελάτη), και πολλά περισσότερα.

JavaScript -2

- Σχεδιάστηκε για να προσθέσει διαδραστικότητα (interactivity) σε HTML σελίδες.
- Η JavaScript συνήθως γράφεται απευθείας μέσα στις HTML σελίδες.
- Η JavaScript είναι interpreted language (σημαίνει ότι τα scripts εκτελούνται χωρίς προκαταρκτική μεταγλώττιση)
- Όλοι μπορούν να χρησιμοποιήσουν την γλώσσα JavaScript χωρίς να απαιτείται η αγορά κάποιας άδειας.

JavaScript vs Java

- Java και JavaScript είναι δύο εντελώς διαφορετικές γλώσσες και στις δύο ιδέα και το σχεδιασμό!
- Java (developed by Sun Microsystems) είναι μια ισχυρή και πολύ πιο περίπλοκη γλώσσα προγραμματισμού - στην ίδια κατηγορία με C και C ++.

Τι μπορεί να κάνει JavaScript; - 1

- **Javascript δίνει στους σχεδιαστές HTML ένα εργαλείο προγραμματισμού** – Συντάκτες HTML δεν είναι συνήθως προγραμματιστές, αλλά JavaScript είναι μια γλώσσα scripting με πολύ απλή σύνταξη. Σχεδόν ο καθένας μπορεί να βάλει μικρά «αποσπάσματα» του κώδικα σε σελίδες HTML
- **JavaScript μπορεί να βάλει δυναμικό κείμενο σε μια σελίδα HTML** - A JavaScript statement like this: `document.write("<h1>" + name + "</h1>")` can write a variable text into an HTML page
- **JavaScript μπορεί να αντιδράσει σε γεγονότα** - Το JavaScript μπορεί να ρυθμιστεί ώστε να εκτελεστεί όταν συμβαίνει κάτι, όπως όταν μια σελίδα έχει ολοκληρώσει τη φόρτωση ή όταν ένας χρήστης κάνει κλικ σε ένα στοιχείο HTML

Τι μπορεί να κάνει JavaScript; - 2

- **JavaScript μπορεί να διαβάσει και να γράψει στοιχεία HTML** - Ένα Javascript μπορεί να διαβάσει και να αλλάξει το περιεχόμενο ενός στοιχείου HTML
- **JavaScript μπορεί να χρησιμοποιηθεί για την επικύρωση δεδομένων** - Ένα Javascript για να χρησιμοποιηθεί για την επικύρωση δεδομένων φόρμας, πριν υποβληθεί σε ένα διακομιστή. Αυτό σώζει το διακομιστή από επιπλέον επεξεργασία
- **JavaScript μπορεί να χρησιμοποιηθεί για την ανίχνευση του προγράμματος περιήγησης του επισκέπτη** - Ένα Javascript μπορεί να χρησιμοποιηθεί για την ανίχνευση του προγράμματος περιήγησης του επισκέπτη, και - ανάλογα με το πρόγραμμα περιήγησης - φορτώνει μια άλλη σελίδα που έχει σχεδιαστεί ειδικά για το συγκεκριμένο πρόγραμμα περιήγησης
- **JavaScript μπορεί να χρησιμοποιηθεί για να δημιουργήσουμε cookies** - Η JavaScript μπορεί να χρησιμοποιηθεί για την αποθήκευση και ανάκτηση πληροφοριών στον υπολογιστή του επισκέπτη

Χρήση Javascript για να γράψουμε κείμενο σε μια ιστοσελίδα

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
document.write("Hello World!");
```

```
</script>
```

```
</body>
```

```
</html>
```



Hello World!

Η λέξη **document.write** είναι μια τυπική εντολή την Javascript για εγγραφή εξόδου σε μια σελίδα.

Χρήση JavaScript να γράψει ετικέτες HTML σε μια ιστοσελίδα

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
document.write("<h1>This is a header</h1>");
```

```
</script>
```

```
</body>
```

```
</html>
```

This is a header

Τα Scripts πού να τοποθετηθούν

- JavaScripts σε μια σελίδα θα εκτελεστεί αμέσως μόλις η σελίδα φορτώνεται στο πρόγραμμα περιήγησης. Αυτό δεν είναι πάντα αυτό που θέλουμε. Μερικές φορές θέλουμε να εκτελέσει ένα σενάριο, όταν μια σελίδα φορτώνει, και άλλες φορές όταν ένας χρήστης ενεργοποιεί ένα συμβάν.
- JavaScripts στο τμήμα του σώματος (body section) θα εκτελεστεί, κατά τη φόρτωση της σελίδας.
- JavaScripts στο τμήμα της κεφαλής (head section) θα εκτελεστεί όταν **καλείται**.
- Μπορείτε να τοποθετήσετε έναν απεριόριστο αριθμό scripts στο έγγραφό σας, ώστε να μπορείτε να έχετε scripts τόσο στο body section όσο και στο head section.

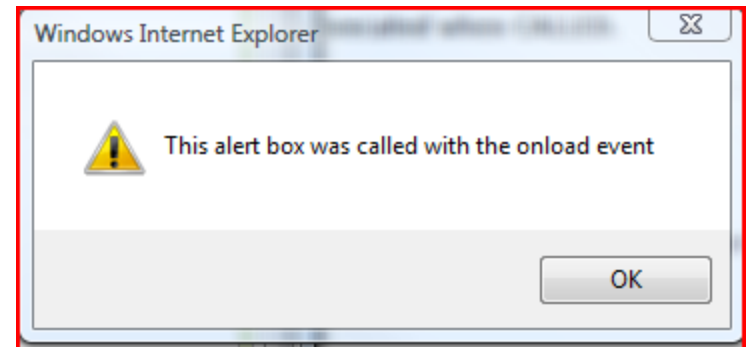
Head Section

Scripts που θα εκτελεστούν όταν καλούνται, ή όταν ένα συμβάν έχει προκληθεί, (scripts που περιέχουν functions) μπαίνουν στην ενότητα head. Στη συνέχεια, μπορούμε να είμαστε βέβαιοι ότι το script έχει φορτωθεί πριν καλείται η function του.

```
<html>
<head>
<script type="text/javascript">
function message()
{
alert("This alert box was called with the onload event");
}
</script>
</head>

<body onload="message()">

</body>
</html>
```

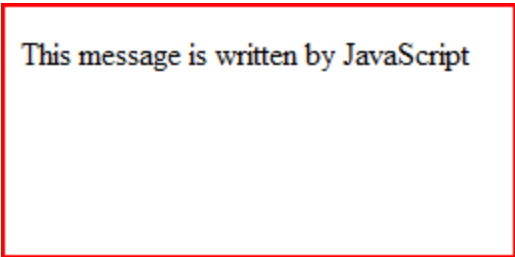


Body Section

Scripts που θα εκτελεστούν όταν η σελίδα φορτώνει πάνε στο τμήμα του body.

```
<html>  
<head>  
</head>
```

```
<body>  
<script type="text/javascript">  
document.write("This message is written by  
    JavaScript");  
</script>  
</body>  
  
</html>
```



This message is written by JavaScript

Χρησιμοποιώντας ένα εξωτερικό JavaScript

- Μερικές φορές μπορεί να θέλετε να εκτελέσετε την ίδια Javascript για πολλές σελίδες, χωρίς να χρειάζεται να γράψετε το ίδιο script σε κάθε σελίδα.
- Για να απλοποιηθεί αυτό, μπορείτε να γράψετε το JavaScript σε ένα εξωτερικό αρχείο. Αποθηκεύστε το εξωτερικό αρχείο JavaScript με .js επέκταση αρχείου.

Note: The external script cannot contain the <script> tag!

- To use the external script, point to the .js file in the "src" attribute of the <script> tag:

```
<html>
<head>
</head>
<body>
<script type="text/javascript" src="xxx.js">
</script>
<p>
The actual script is in an external script file called "xxx.js".
</p>

</body>
</html>
```

This text was written by an external script!

The actual script is in an external script file called "xxx.js".

JavaScript Statements

- JavaScript είναι μια σειρά από statements που θα εκτελεστούν από το πρόγραμμα περιήγησης.
- JavaScript statement είναι μια εντολή στο πρόγραμμα περιήγησης. Ο σκοπός της εντολής είναι να πει το πρόγραμμα περιήγησης τι να κάνει.
 - `document.write("Hello Dolly");`
- Σε αντίθεση με HTML, JavaScript είναι case sensitive - ως εκ τούτου, να παρακολουθήσετε τη διάκριση κεφαλαία-πεζά προσεκτικά όταν γράφετε statement JavaScript, δημιουργείτε ή καλείτε μεταβλητές, αντικείμενα και functions.
- Το ερωτηματικό είναι προαιρετικό (κατά το πρότυπο JavaScript), και το πρόγραμμα περιήγησης υποτίθεται ότι ερμηνεύει το τέλος της γραμμής ως το τέλος του statement.

JavaScript blocks

- JavaScript statements μπορούν να ομαδοποιηθούν σε blocks. (τμήμα κώδικα)
- Blocks αρχίζουν με αριστερό { και τελειώνουν με δεξιό }.
- Ο σκοπός του block είναι να κάνει την ακολουθία των statements να εκτελεστούν μαζί.
- Κανονικά ένα block χρησιμοποιείται για να ομαδοποιήσει statements μαζί σε μια function ή condition (where a group of statements should be executed if a condition is met).

JavaScript Comments

Τα σχόλια Javascript μπορούν να χρησιμοποιηθούν για να κάνουν τον κώδικα πιο ευανάγνωστο.

- Single line comments start with //.

```
<script type="text/javascript">
// This will write a header:
document.write("<h1>This is a header</h1>");
// This will write two paragraphs:
document.write("<p>This is a paragraph</p>");
document.write("<p>This is another paragraph</p>");
</script>
```

- Multi line comments start with /* and end with */.

```
<script type="text/javascript">
/*
The code below will write
one header and two paragraphs
*/
document.write("<h1>This is a header</h1>");
document.write("<p>This is a paragraph</p>");
document.write("<p>This is another paragraph</p>");
</script>
```

Variables - 1

- Οι μεταβλητές είναι «δοχεία» για την αποθήκευση πληροφοριών.
- Τα ονόματα των μεταβλητών είναι case sensitive (y και το Y είναι δύο διαφορετικές μεταβλητές).
- Τα ονόματα των μεταβλητών πρέπει να αρχίζουν με ένα γράμμα ή το χαρακτήρα υπογράμμισης.
- Δηλώστε μεταβλητές JavaScript με το **var statement**.
 - var x; var x=5;
 - var carname; var carname="Volvo";
- Εάν αναθέσετε τιμές σε μεταβλητές που δεν έχουν ακόμη δηλωθεί, οι μεταβλητές αυτόματα θα δηλωθούν.

Variables - 2

```
<html>
<body>

<script type="text/javascript">
var firstname;
firstname="Hege";
document.write(firstname);
document.write("<br />");
firstname="Tove";
document.write(firstname);
</script>
```

<p>The script above declares a variable,
assigns a value to it, displays the value, change the value,
and displays the value again.</p>

```
</body>
</html>
```

Hege
Tove

The script above declares a variable, assigns a value to it, displays the value, change the value, and displays the value again.

JavaScript Arithmetic Operators

Operator	Description	Example	Result
+	Addition	$x = y + 2$	$x = 7$
-	Subtraction	$x = y - 2$	$x = 3$
*	Multiplication	$x = y * 2$	$x = 10$
/	Division	$x = y / 2$	$x = 2.5$
%	Modulus (division remainder)	$x = y \% 2$	$x = 1$
++	Increment	$x = ++y$	$x = 6$
--	Decrement	$x = --y$	$x = 4$

Operator	Example	Same As	Result
=	$x = y$		$x = 5$
+=	$x += y$	$x = x + y$	$x = 15$
-=	$x -= y$	$x = x - y$	$x = 5$
*=	$x *= y$	$x = x * y$	$x = 50$
/=	$x /= y$	$x = x / y$	$x = 2$
%=	$x \% = y$	$x = x \% y$	$x = 0$

The + Operator Used on Strings

```
txt1="What a very";  
txt2="nice day";  
txt3=txt1+txt2;      Txt3 □ "What a very nice day"
```

```
<html>  
<body>  
<script type="text/javascript">  
x=5+5;  
document.write(x);  
document.write("<br />");  
x="5"+"5";  
document.write(x);  
document.write("<br />");  
x=5+"5";  
document.write(x);  
document.write("<br />");  
</script>  
</body>  
</html>
```

10
55
55

The common rule is: If you add a number and a string, the result will be a string.

Comparison & Logical Operators

Given that **x=5**, the table below explains the comparison operators:

Operator	Description	Example
==	is equal to	x==8 is false
===	is exactly equal to (value and type)	x===5 is true x==="5" is false
!=	is not equal	x!=8 is true
>	is greater than	x>8 is false
<	is less than	x<8 is true
>=	is greater than or equal to	x>=8 is false
<=	is less than or equal to	x<=8 is true

Given that **x=6 and y=3**, the table below explains the logical operators:

Operator	Description	Example
&&	and	(x < 10 && y > 1) is true
	or	(x==5 y==5) is false
!	not	!(x==y) is true

If...Else Statements

```
<html>
<body>
<script type="text/javascript">
var d = new Date();
var time = d.getHours();
if (time<10)
{
document.write("<b>Good morning</b>");
}
else if (time>=10 && time<16)
{
document.write("<b>Good day</b>");
}
else
{
document.write("<b>Hello World!</b>");
}
</script>
<p>
This example demonstrates the if..else if...else statement.
</p>
</body>
</html>
```

Hello World!

This example demonstrates the if..else if...else statement.

Switch Statement

```
<html>
<body>
<script type="text/javascript">
var d = new Date();
theDay=d.getDay();
switch (theDay)
{
case 5:
    document.write("<b>Finally Friday</b>");
    break;
case 6:
    document.write("<b>Super Saturday</b>");
    break;
case 0:
    document.write("<b>Sleepy Sunday</b>");
    break;
default:
    document.write("<b>I'm really looking forward to this weekend!</b>");
}
</script>

<p>This JavaScript will generate a different greeting based on what day
    it is. Note that Sunday=0, Monday=1, Tuesday=2, etc.</p>

</body>
</html>
```

Super Saturday

This JavaScript will generate a different greeting based on what day it is. Note that Sunday=0, Monday=1, Tuesday=2, etc.

Popup Boxes

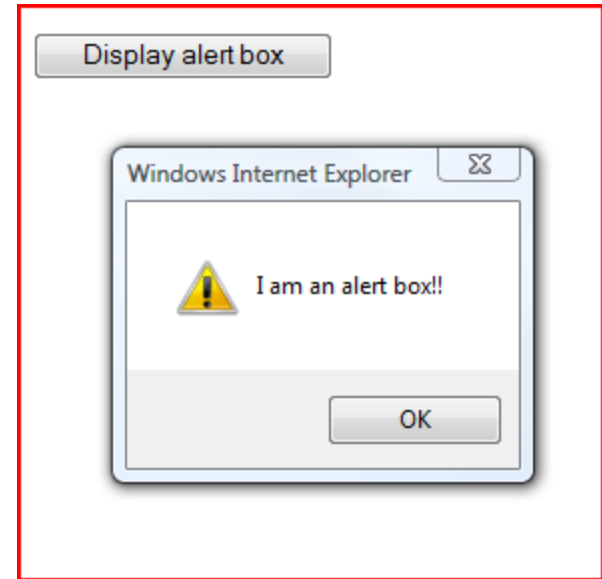
- Σε Javascript μπορούμε να δημιουργήσουμε τρία είδη popup boxes: Alert box, Confirm box, and Prompt box.
 - Ένα **alert box** χρησιμοποιείται συχνά αν θέλετε να βεβαιωθείτε ότι οι πληροφορίες έρχονται από το χρήστη. Όταν ένα πλαίσιο ειδοποίησης εμφανίζεται, ο χρήστης θα πρέπει να κάνει κλικ στο «OK» για συνέχεια.
 - `alert("sometext");`
 - Ένα **confirm box** χρησιμοποιείται συχνά αν θέλετε ο χρήστης να ελέγξει ή να αποδεχθεί κάτι. Όταν ένα παράθυρο επιβεβαίωσης εμφανίζεται, ο χρήστης θα πρέπει να κάνει κλικ στην επιλογή «OK» ή «Ακύρωση» για συνέχεια. Εάν ο χρήστης κάνει κλικ στο «OK», το πλαίσιο επιστρέφει true. Εάν ο χρήστης κάνει κλικ στο «Ακύρωση», το πλαίσιο επιστρέφει false
 - `confirm("sometext");`
 - Ένα **prompt box** χρησιμοποιείται συχνά αν θέλετε ο χρήστης να εισάγει μια τιμή πριν από την είσοδο σε μια σελίδα. Όταν ένα παράθυρο διαλόγου εμφανίζεται, ο χρήστης θα πρέπει να κάνει κλικ στην επιλογή «OK» ή «Ακύρωση» για να συνεχίσετε μετά την εισαγωγή μιας τιμής εισόδου. Εάν ο χρήστης κάνει κλικ στο «OK» το παράθυρο επιστρέφει την τιμή εισόδου. Εάν ο χρήστης κάνει κλικ στο «Ακύρωση» το παράθυρο επιστρέφει null.
 - `prompt("sometext","defaultvalue");`

Alert box

```
<html>
<head>
<script type="text/javascript">
function disp_alert()
{
alert("I am an alert box!!");
}
</script>
</head>
<body>
```

```
<input type="button" onclick="disp_alert()" value="Display
  alert box" />
```

```
</body>
</html>
```

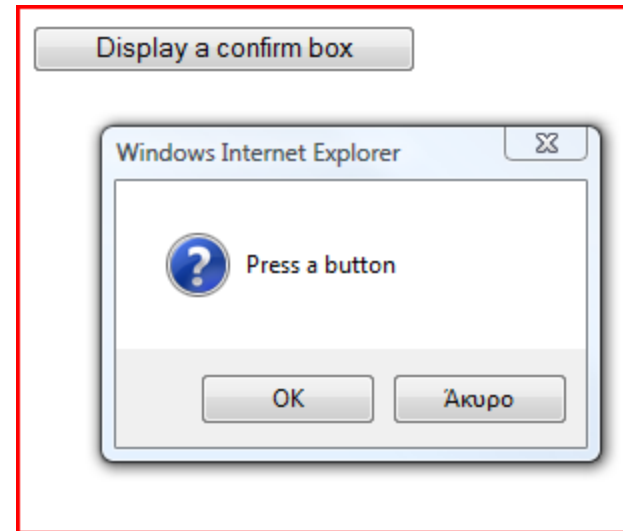


Confirm box

```
<html>
<head>
<script type="text/javascript">
function disp_confirm()
{
var r=confirm("Press a button");
if (r==true)
{
document.write("You pressed OK!");
}
else
{
document.write("You pressed Cancel!");
}
}
</script>
</head>
<body>

<input type="button" onclick="disp_confirm()" value="Display
a confirm box" />

</body>
</html>
```



You pressed OK!

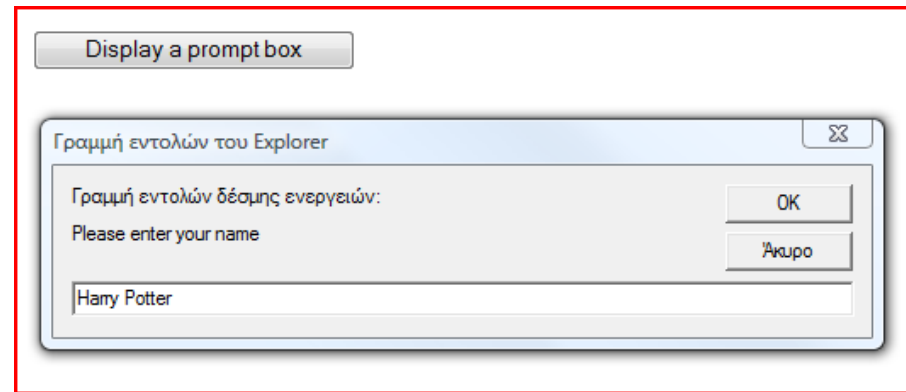
You pressed Cancel!

Prompt box

```
<html>
<head>
<script type="text/javascript">
function disp_prompt()
{
var name=prompt("Please enter your name","Harry Potter");
if (name!=null && name!="")
{
document.write("Hello " + name + "! How are you today?");
}
}
</script>
</head>
<body>

<input type="button" onclick="disp_prompt()" value="Display a
prompt box" />

</body>
</html>
```



Functions

- Αναφέραμε ότι για να κρατήσετε το πρόγραμμα περιήγησης από την εκτέλεση ενός script, όταν η σελίδα φορτώνει, μπορείτε να βάλετε το script σας σε function. Μια function είναι ένα επαναχρησιμοποιήσιμο μπλοκ κώδικα που θα εκτελεστεί από ένα γεγονός, ή όταν η function καλείται.
- Μπορείτε να καλέσετε μια function από οπουδήποτε εντός της σελίδας (ή ακόμα και από άλλες σελίδες, εάν η function είναι ενσωματωμένη σε ένα εξωτερικό αρχείο .js).
- Function μπορούν να οριστούν τόσο στο <head> και στο <body> τμήμα ενός εγγράφου. Ωστόσο, για να εξασφαλιστεί ότι η function φορτώνεται από το πρόγραμμα περιήγησης πριν καλεστεί, είναι καλύτερα να τη βάλετε στο <head>

Functions Example

```
<html>
<head>
<script type="text/javascript">
function product(a,b)
{
return a*b;
}
</script>
</head>
```

```
<body>
<script type="text/javascript">
document.write(product(4,3));
</script>
```

```
<p>The script in the body section calls a function with two
parameters (4 and 3).</p>
```

```
<p>The function will return the product of these two
parameters.</p>
```

```
</body>
```

```
</html>
```

12

The script in the body section calls a function with two parameters (4 and 3).

The function will return the product of these two parameters.

For Loop

- Οι βρόχοι σε JavaScript χρησιμοποιούνται για να εκτελέσουμε το ίδιο μπλοκ κώδικα για ένα συγκεκριμένο αριθμό φορών ή για όσο μια συγκεκριμένη συνθήκη είναι αληθής
- Υπάρχουν δύο διαφορετικά είδη βρόχων:
 - **for** - loops through a block of code a specified number of times
 - **while** - loops through a block of code while a specified condition is true

```
<html>
<body>
<script type="text/javascript">
var i=0;
for (i=0;i<=10;i++)
{
document.write("The number is " + i);
document.write("<br />");
}
</script>
</body>
</html>
```

```
<html>
<body>
<script type="text/javascript">
var i=0;
while (i<=10)
{
document.write("The number is " + i);
document.write("<br />");
i=i+1;
}
</script>
</body>
</html>
```

Events

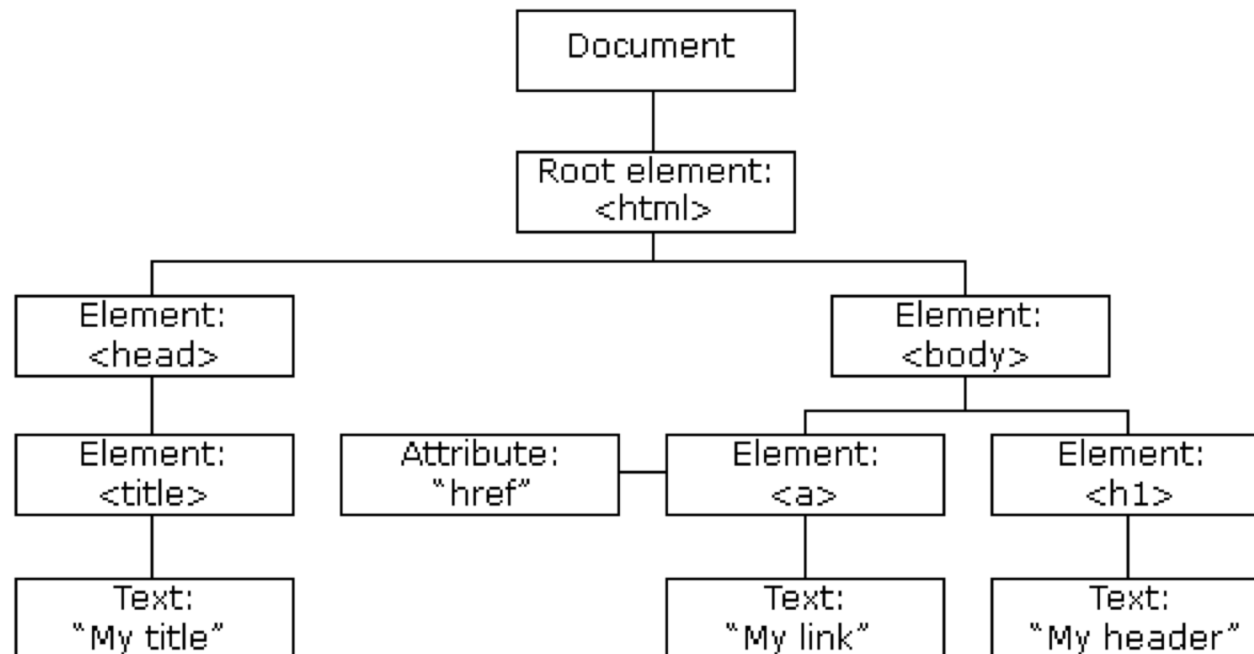
- Γεγονότα (Events) είναι ενέργειες (actions) που μπορούν να ανιχνευθούν από Javascript.
- Παραδείγματα γεγονότων:
 - A mouse click
 - A web page or an image loading
 - Mousing over a hot spot on the web page
 - Selecting an input box in an HTML form
 - Submitting an HTML form
 - A keystroke
- onload and onUnload
- onFocus (The event occurs when an element gets focus), onBlur (The event occurs when an element loses focus) and onChange
- onSubmit
- onMouseOver and onMouseOut

Form Validation (Επικύρωση δεδομένων φόρμας)

- JavaScript μπορεί να χρησιμοποιηθεί για την επικύρωση των δεδομένων εισόδου σε φόρμες HTML πριν από την αποστολή του περιεχομένου σε ένα διακομιστή.
- Δεδομένα φόρμας που συνήθως ελέγχονται από Javascript θα μπορούσε να είναι:
 - has the user left required fields empty?
 - has the user entered a valid e-mail address?
 - has the user entered a valid date?
 - has the user entered text in a numeric field?

The HTML DOM (Document Object Model)

- Όταν φορτώνεται μια ιστοσελίδα, το πρόγραμμα περιήγησης δημιουργεί το Document Object Model της σελίδας.
- Το μοντέλο HTML DOM είναι κατασκευασμένο σαν ένα δέντρο από αντικείμενα:



Dynamic HTML

Με το μοντέλο αντικειμένου (object model), JavaScript παίρνει ότι χρειάζεται για τη δημιουργία δυναμικών HTML:

- για να αλλάξετε όλα τα στοιχεία HTML στη σελίδα
- για να αλλάξει όλα τα HTML χαρακτηριστικά στη σελίδα
- για να αλλάξετε όλα τα στυλ CSS στη σελίδα
- για να αφαιρέσετε τα υπάρχοντα στοιχεία HTML και χαρακτηριστικά
- για να προσθέσετε νέα στοιχεία HTML και χαρακτηριστικά
- για να αντιδράσουμε σε όλα τα υπάρχοντα HTML γεγονότα στη σελίδα
- για να δημιουργήσετε νέα HTML γεγονότα στη σελίδα

Τι είναι το DOM;

- Το DOM είναι ένα W3C (Wide Web Consortium World) πρότυπο. Ορίζει ένα πρότυπο για την πρόσβαση σε έγγραφα.
- *“The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document.”*
- Το πρότυπο W3C DOM χωρίζεται σε 3 διαφορετικά μέρη:
 - Core DOM - standard model for all document types
 - XML DOM - standard model for XML documents
 - HTML DOM - standard model for HTML documents

Τι είναι το HTML DOM;

- Το HTML DOM είναι ένα πρότυπο μοντέλο αντικειμένου και διεπαφή προγραμματισμού (API) για την HTML. Ορίζει:
 - Τα στοιχεία HTML ως αντικείμενα
 - Τις ιδιότητες όλων των στοιχείων HTML
 - Τις μεθόδους για να αποκτήσετε πρόσβαση σε όλα τα στοιχεία HTML
 - Τα γεγονότα για όλα τα στοιχεία HTML
- Με άλλα λόγια το HTML DOM είναι ένα πρότυπο για το πώς να πάρετε, αλλάξετε, να προσθέσετε ή να διαγράψετε στοιχεία HTML.

JavaScript - HTML DOM Methods

- **Μέθοδοι HTML DOM** είναι ενέργειες που μπορείτε να εκτελέσετε (σε HTML στοιχεία).
- **Ιδιότητες HTML DOM** είναι τιμές (HTML στοιχεία) που μπορείτε να ορίσετε ή να αλλάξετε.

The DOM Programming Interface

- Το HTML DOM μπορεί να προσπελαστεί με JavaScript (και με άλλες γλώσσες προγραμματισμού).
- Στο DOM, όλα τα στοιχεία HTML ορίζονται ως αντικείμενα.
- Η διεπαφή προγραμματισμού είναι οι ιδιότητες και οι μέθοδοι του κάθε αντικειμένου.

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

<h1>My First Page</h1>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello World!";
</script>

</body>
</html>
```

My First Page

Hello World!

The getElementById Method

- Ο πιο συνηθισμένος τρόπος για να αποκτήσετε πρόσβαση σε ένα στοιχείο HTML είναι να χρησιμοποιήσετε το id του στοιχείου.
- Στο παραπάνω παράδειγμα η μέθοδος getElementById χρησιμοποιείται id = «demo» για να βρει το στοιχείο.
- Η innerHTML ιδιότητα είναι χρήσιμη για τη λήψη ή αντικατάσταση του περιεχομένου των στοιχείων HTML. (συμπεριλαμβανομένων των <html> και <body>)

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

<h1>My First Page</h1>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello World!";
</script>

</body>
</html>
```

The HTML DOM Document Object

- Το αντικείμενο έγγραφο (Document Object) αντιπροσωπεύει την ιστοσελίδα σας.
- Αν θέλετε να έχετε πρόσβαση σε κάθε στοιχείο σε μια σελίδα HTML, πρέπει πάντα να ξεκινάτε με την πρόσβαση στο αντικείμενο του εγγράφου.
- Παραδείγματα προσθήκης και διαγραφής στοιχείων

Method	Description
<code>document.createElement(<i>element</i>)</code>	Create an HTML element
<code>document.removeChild(<i>element</i>)</code>	Remove an HTML element
<code>document.appendChild(<i>element</i>)</code>	Add an HTML element
<code>document.replaceChild(<i>element</i>)</code>	Replace an HTML element
<code>document.write(<i>text</i>)</code>	Write into the HTML output stream

The HTML DOM Document Object

- Παραδείγματα αλλαγής HTML Στοιχείων

Method	Description
<code>element.innerHTML = new html content</code>	Change the inner HTML of an element
<code>element.attribute = new value</code>	Change the attribute value of an HTML element
<code>element.setAttribute(attribute, value)</code>	Change the attribute value of an HTML element
<code>element.style.property = new style</code>	Change the style of an HTML element

- Παραδείγματα Events Handlers

Method	Description
<code>document.getElementById(id).onclick = function(){code}</code>	Adding event handler code to an onclick event

Παράδειγμα 1

```
<!DOCTYPE html>
<html>
  <head>
    <title>W3Schools Demo</title>
  </head>
  <body>

    <p id="demo"></p>

    <script>
document.getElementById("demo").innerHTML =
"The title of this document is: " + document.title;
</script>

  </body>
</html>
```

The title of this document is: W3Schools Demo

Παράδειγμα 2

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

<p>The full URL of this document is: <br><span id="demo"></span>.</p>

<script>
document.getElementById("demo").innerHTML = document.URL
</script>

</body>
</html>
```

The full URL of this document is:
https://www.w3schools.com/js/tryit.asp?filename=tryjs_doc_url.

Παράδειγμα 3

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

<h2>What Can JavaScript Do?</h2>

<p>JavaScript can change HTML attributes.</p>

<p>In this case JavaScript changes the src (source) attribute of an image.</p>

<button onclick="document.getElementById('myImage').src='pic_bulbon.gif'">Turn on the
light</button>



<button onclick="document.getElementById('myImage').src='pic_bulboff.gif'">Turn off the
light</button>

</body>
</html>
```

JavaScript can change HTML attributes.

In this case JavaScript changes the src (source) attribute of an image.



Turn on the light

Turn off the light

JavaScript can change HTML attributes.

In this case JavaScript changes the src (source) attribute of an image.



Turn on the light

Turn off the light

Παράδειγμα 4

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

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can change the style of an HTML element.</p>

<button type="button"
onclick="document.getElementById('demo').style.fontSize='35px'">Click Me!</button>

</body>
</html>
```

JavaScript can change the style of an HTML element.

Click Me!

JavaScript can change the style of an HTML element.

Click Me!

Παράδειγμα 5

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

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can hide HTML elements.</p>

<button type="button" onclick="document.getElementById('demo').style.display='none'">Click
Me!</button>

</body>
</html>
```

JavaScript can hide HTML elements.

Click Me!

Click Me!

Παράδειγμα 6

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

<h2>What Can JavaScript Do?</h2>

<p>JavaScript can show hidden HTML elements.</p>

<p id="demo" style="display:none">Hello JavaScript!</p>

<button type="button" onclick="document.getElementById('demo').style.display='block'">Click Me!</button>

</body>
</html>
```

JavaScript can show hidden HTML elements.

Click Me!

JavaScript can show hidden HTML elements.

Hello JavaScript!

Click Me!

Επιπλέον Μελέτη

- www.w3schools
 - JavaScript tutorial
 - JS Objects
 - JS Advanced