

# Σημειώσεις JavaScript

Νίκος Ζ. Ζάχαρης  
nzach@teipir.gr

# Εισαγωγή στη JavaScript

Η JavaScript είναι μια ελαφριά γλώσσα για την συγγραφή σεναρίων (scripting language) τα οποία θα ενσωματωθούν απευθείας σε μια Html σελίδα.

Η JavaScript δεν χρειάζεται μεταγλώττιση και η χρήση της, είναι ελεύθερη, χωρίς την απόκτηση κάποιας άδειας

Η JavaScript σχεδιάστηκε για να :

- α) μειώσει το φόρτο εργασίας στους εξυπηρετητές
- β) να προσθέσει αλληλεπίδραση στις HTML σελίδες

## Τι μπορεί να κάνει η JavaScript;

Η JavaScript δίνει στους γραφίστες ένα προγραμματιστικό εργαλείο, το οποίο είναι εύκολο στην εκμάθηση και η προσθήκη ενός σεναρίου σε μια σελίδα είναι σχετικά, μια εύκολη υπόθεση.

Η JavaScript μπορεί να προσθέσει κείμενο με δυναμικό τρόπο σε μια σελίδα.

Η JavaScript μπορεί να παραλάβει γεγονότα και να αντιδράσει σε αυτά, όπως το φόρτωμα μιας σελίδας, το πάτημα ενός κουμπιού κ.λπ.

Η JavaScript μπορεί να διαβάσει και να αλλάξει την κατάσταση ή το περιεχόμενο ενός HTML αντικειμένου.

Η JavaScript μπορεί να κάνει πιστοποίηση δεδομένων.

Η JavaScript μπορεί να ανιχνεύσει την έκδοση του φυλλομετρητή και εν συνεχεία να φορτώσει το σενάριο για το συγκεκριμένο πρόγραμμα

Η JavaScript μπορεί να διαβάσει και να δημιουργήσει cookies.

## Το πρώτο μου σενάριο.

Ανοίξτε το Σημειωματάριο (Notepad) και πληκτρολογήστε :

```
<html>
<body>
<script type="text/javascript">
    document.write("Το πρώτο πρόγραμμα")
</script>
</body>
</html>
```

Υπάρχει η συνήθεια στους προγραμματιστές να προσθέτουν το ; στο τέλος κάθε εντολής. Το ελληνικό ερωτηματικό είναι ΠΡΟΑΙΡΕΤΙΚΟ και δεν χρειάζεται να το προσθέτουμε στο τέλος κάθε εντολής εκτός από την περίπτωση που σε μια γραμμή συμπεριλάβουμε δύο εντολές :

```
document.write("Καλημέρα, "); document.write("το πρώτο πρόγραμμα")
```

## Σχόλια σε HTML και στην JavaScript

Δεν είναι υποχρεωτικό αλλά για λόγους συμβατότητας με παλαιότερους browsers τοποθετούμε όλες τις εντολές μέσα σε HTML σχόλια, όπως παρακάτω :

```
<html>
<body>
<script type="text/javascript">
  <!--
    document.write("Το πρώτο πρόγραμμα")
  //-->
</script>
</body>
</html>
```

Σχόλια στην JavaScript μπορούμε να κάνουμε σε μία γραμμή τοποθετώντας στην αρχή της κάθε μιας το // ενώ σε πολλές γραμμές /\* και \*/

```
<script type="text/javascript">
// Ένα σχόλιο σε μια γραμμή
document.write("Το πρώτο πρόγραμμα")
/*
Σε πολλές γραμμές
*/
</script>
```

## Που γράφω τα **scripts** στην JavaScript

Επαναλαμβάνω το συνδυασμό `<script ..> .... </script>` όσες φορές θέλω μέσα στο τμήμα `body` και `head`

```
<html>
<head>
  <title>First Script</title>
</head>
<body>
<script type="text/javascript">
document.write ("Γειά σας")
</script>
<br>Μια γραμμή σε html
<script type="text/javascript">
document.write ("<br>Δεύτερο σενάριο")
</script>
</body>
</html>
```

## Που γράφω τα scripts στην JavaScript (συνέχεια)

Σε εξωτερικά αρχεία :

**first.htm**

```
<html>
<head>
  <title>External Script</title>
</head>
<body>
  Εκτέλεση σεναρίου από
  <script src="abc.js" type="text/javascript">
</script>
</body>
</html>
```

**abc.js**

```
document.write ("<br>εξωτερικό αρχείο")
```

# Μεταβλητές (Variable)

Μια μεταβλητή είναι χώρος στην μνήμη του ΗΥ για την αποθήκευση δεδομένων. Τα περιεχόμενα μιας μεταβλητής μπορούμε να τα διαβάσουμε ή/και να τα τροποποιήσουμε.

```
<html>
<head>
  <title>Variable</title>
</head>
<body>
<script language="javascript" >
var x, y
x = 10
y = 20
document.write (x + y)
document.write("<br>")
x = "Good "
y = "Moring "
s = x + y
document.write (s)
</script>
</body>
</html>
```

Η λέξη var πριν την δήλωση των μεταβλητών είναι προαιρετική. Δηλαδή στο διπλανό πρόγραμμα θα μπορούσαμε να αγνοήσουμε την γραμμή

```
var x, y
```

Επίσης στην μεταβλητή υπάρχει η δυνατότητα να αλλάξουμε τα περιεχόμενα καθώς και τον τύπο τους δηλαδή να αποθηκεύσουμε αρχικά αριθμούς, μετά αλφαριθμητικά.

# Αριθμητικοί Τελεστές

| Τελεστής | Πράξη           | Παράδειγμα                   | Αποτέλεσμα  |
|----------|-----------------|------------------------------|-------------|
| +        | Πρόσθεση        | $x=6$<br>$y=3$<br>$x+y$      | 9           |
| -        | Αφαίρεση        | $x=4$<br>$y=1$<br>$x-y$      | 3           |
| *        | Πολλαπλασιασμός | $x=2$<br>$y=4$<br>$x*y$      | 8           |
| /        | Διαίρεση        | $25/5$<br>$7/2$              | 5<br>3.5    |
| %        | Υπόλοιπο        | $5\%2$<br>$10\%7$<br>$10\%2$ | 1<br>3<br>0 |
| ++       | Αύξηση κατά ένα | $x=3$<br>$x++$               | $x=4$       |
| --       | Μείωση κατά ένα | $x=5$<br>$x--$               | $x=4$       |

# Τελεστές Απόδοσης Τιμής

| Τελεστής | Παράδειγμα | Εναλλακτικά | Σε όλα τα παραδείγματα θεωρούμε αρχικά<br>$x = 10$<br>$y = 7$ |
|----------|------------|-------------|---|
| =        | $x=y$      |             | $x = 7$<br>$y = 7$  |
| +=       | $x+=y$     | $x=x+y$     | $x = 17$<br>$y = 7$   |
| -=       | $x-=y$     | $x=x-y$     | $x = 3$<br>$y = 7$  |
| *=       | $x*=y$     | $x=x*y$     | $x = 70$<br>$y = 7$   |
| /=       | $x/=y$     | $x=x/y$     | $x = 1,42$<br>$y = 7$   |
| %=       | $x\%=y$    | $x=x\%y$    | $x = 3$<br>$y = 7$  |

## Παράδειγμα Δοκιμής Τελεστών

```
<html>
<head>
  <title>Operators</title>
</head>
<body>
<script language="javascript" >
x = 10
y = 20
x *= y
document.write (x)
</script>
</body>
</html>
```

# Η συνθήκη if

Η δήλωση if μας επιτρέπει να εκτελέσουμε διαφορετικό κώδικα ανάλογα των συνθηκών. Η σύνταξη της εντολής είναι :

```
if(συνθήκη)
{
    // εντολές
}
```

Αν η συνθήκη είναι αληθές τότε και μόνο τότε θα εκτελεστούν οι εντολές

```
if(συνθήκη)
{
    // εντολές IF
}
else
{
    // εντολές ELSE
}
```

Αν η συνθήκη είναι αληθές τότε και μόνο τότε θα εκτελεστούν οι εντολές IF διαφορετικά θα εκτελεστούν οι εντολές του ELSE

```
if(συνθήκη) {
    // εντολές συνθήκης
}
else if(συνθήκη2) {
    // εντολές συνθήκης2
}
else if(συνθήκη3) {
    // εντολές συνθήκης3
}
else {
    // εντολές else
}
```

Θα εκτελεστούν μόνο οι εντολές για τις οποίες αληθεύει η συνθήκη τους, διαφορετικά θα εκτελεστούν οι εντολές του else

# Συνθήκη

Μια συνθήκη περιλαμβάνει ένα τελεστή σύγκρισης ή μια λογική τιμή.

## Τελεστές Σύγκρισης

| Τελεστής           | Περιγραφή                        | Σε όλα τα παραδείγματα θεωρούμε αρχικά<br>$x = 10$<br>$y = 7$<br>$z = "10"$ |
|--------------------|----------------------------------|---|
| <code>==</code>    | Είναι ίσες οι τιμές              | $x==z$ Αληθές<br>$x==y$ Ψευδές  |
| <code>===</code>   | Είναι ίσες οι τιμές και οι τύποι | $x===z$ Ψευδές  |
| <code>!=</code>    | Άνισο                            | $x!=y$ Αληθές   |
| <code>&gt;</code>  | Μεγαλύτερο από                   | $x>y$ Αληθές  |
| <code>&lt;</code>  | Μικρότερο από                    | $x<y$ Ψευδές  |
| <code>&gt;=</code> | Μεγαλύτερο ή ίσο                 | $x>=y$ Αληθές   |
| <code>&lt;=</code> | Μικρότερο ή ίσο                  | $x<=y$ Ψευδές   |

## Παράδειγμα Τελεστών Σύγκρισης

```
<html>
<head>
  <title>Comparison Operators </title>
</head>
<body>
<script language="javascript" >
x = 10
y = 20
if (x >= y)
{
  document.write (x)
}
else
{
  document.write (y)
}
</script>
</body>
</html>
```

```
<html>
<head>
  <title>Comparison Operators </title>
</head>
<body>
<script language="javascript" >
x = 10
y = 20
z = (x >= y)
if (z)
{
  document.write (x)
}
else
{
  document.write (y)
}
</script>
</body>
</html>
```

## Σύνθετες Συνθήκες

Μπορούμε να συνδυάσουμε δυο συνθήκες με την χρήση λογικών τελεστών. Για παράδειγμα, αν θέλουμε η τιμή της μεταβλητής  $x$  να είναι μεγαλύτερη από 0 (συνθηκη A) και μικρότερη από 10 (συνθήκη B) τότε θα πρέπει να δημιουργήσουμε μια σύνθετη συνθήκη.

```
if(x>0 && x < 10) {  
    //εντολές  
}
```

## Λογικοί Τελεστές

| Τελεστής | Περιγραφή     | Σε όλα τα παραδείγματα θεωρούμε<br>$x = 10$<br>$y = 7$              |
|----------|---------------|---|
| &&       | Λογικό ΚΑΙ    | $(x < 10 \ \&\& \ y > 1)$ Ψευδές<br>$(x > 5 \ \&\& \ y > 1)$ Αληθές |
|          | Λογικό Η      | $(x == 6 \    \ y == 7)$ Αληθές                                     |
| !        | Λογική Άρνηση | $!(x == y)$ Αληθές  |

## Παράδειγμα χρήσης του if

```
<html>
<head>  <title>Display Greetings</title> </head>
<body>
<script language="javascript" >
var d = new Date()
var time = d.getHours()

if (time<9) {
document.write("<b>Good morning</b>")
}
else if (time>10 && time<16) {
document.write("<b>Good day</b>")
}
else {
document.write("<b>Good Evening</b>")
}
</script>

</body>
</html>
```

# Συναρτήσεις

Μία συνάρτηση είναι ένα σύνολο από εντολές το οποίο ενοποιούμε κάτω από ένα όνομα. Μια συνάρτηση μπορεί να την ενεργοποιήσει ένα γεγονός (π.χ. το πάτημα ενός κουμπιού) ή να την καλέσουμε με το ονομά της. Ο λόγος που δημιουργούμε συναρτήσεις είναι για να επαναχρησιμοποιήσουμε τις εντολές της, σε διαφορετικά σημεία στο σενάριο.

Η σύνταξη της συνάρτησης είναι :

```
function ΌνομαΣυνάρτησης(ΜεταβλητήΑ, ΜεταβλητήΒ,.....)
{
    // εντολές
    Επιστροφή τιμής
}
```

Δεν έχουν όλες οι συναρτήσεις είσοδο από μεταβλητές (ορίσματα) και επίσης υπάρχουν συναρτήσεις που δεν επιστρέφουν τιμή.

# Παράδειγμα δημιουργίας μιας συνάρτησης

Θα κάνουμε μια απλή συνάρτηση που δέχεται δύο αριθμούς και επιστρέφει το άθροισμά τους.

A) Θα επιλέξουμε ένα όνομα για την συνάρτηση, το οποίο θα περιγράφει την εργασία που επιτελείται μέσα σε αυτήν. Π.χ. CalcSum

B) Δεύτερον σκεφτόμαστε τα ορίσματα που χρειάζεται για να δουλέψει η συνάρτηση μας. Στην περίπτωση του αθροίσματος χρειαζόμαστε δύο αριθμούς.

Γ) Εν συνεχεία λύνουμε το πρόβλημα. Υπολογίζουμε το άθροισμα

Δ) Επιστρέφουμε το αποτέλεσμα

## Παράδειγμα

Θα δημιουργήσουμε την συνάρτηση **CalcSum**

```
<html>
<head>
<script type="text/javascript">
function CalcSum(a,b)
{
    sum = a + b
    return sum
}
</script>
</head>
<body>
<script language="javascript" >
x = 3
y = 7
document.write (CalcSum(x,y))
document.write("<br>")
document.write (CalcSum(30,60))
</script>
</body>
</html>
```

## Βελτιωμένη έκδοση του παραδείγματος με την συνάρτηση CalcSum

Θα δώσουμε την δυνατότητα στο χρήστη να μπορεί να  
πληκτρολογήσει τις τιμές,

Θα μετατρέψουμε τους χαρακτήρες σε αριθμούς

Δημιουργούμε την διεπαφή για την εισαγωγή των δεδομένων.

```
<html>
<head>
</head>
<body>
<form>
<table>
<tr><td>Πρώτος αριθμός</td><td><input type="text" id="txt1"></td></tr>
<tr><td>Δευτερος αριθμός</td><td><input type="text" id="txt2"></td></tr>
<tr><td>Αποτέλεσμα</td><td><input type="text" id="res"></td></tr>
<tr><td colspan="2" align="center" >
<input type="button" value="Calculate...." onClick="Calculate()">
</td></tr>
</table>
</form>
</body>
</html>
```

Μέσα στις ετικέτες <head> και </head> τοποθετούμε τις εντολές :

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

```
function CalcSum(a,b) {  
    sum = Number(a) + Number(b)  
    return sum  
}
```

Η συνάρτηση **Number** μετατρέπει ένα αλφαριθμητικό σε αριθμό. Εναλλακτικά θα μπορούσαμε να χρησιμοποιήσουμε τις συναρτήσεις **parseInt** και την **parseFloat** για την μετατροπή σε ακέραιο και σε δεκαδικό αριθμό αντίστοιχα.

```
function Calculate() {  
x = document.getElementById('txt1').value  
y = document.getElementById('txt2').value  
if(x == "" || y == "") {  
    alert("Error")  
}  
else {  
    sum = CalcSum(x, y )  
    document.getElementById('res').value = sum  
}}  
</script>
```

Η συνάρτηση **getElementById** επιλέγει το html στοιχείο, μέσα στο κείμενο με το συγκεκριμένο **id**. Την τιμή (**value**) του στοιχείου, μπορούμε να την διαβάσουμε ή να την τροποποιήσουμε με νέα τιμή.

Οι συναρτήσεις `Number`, `parseInt`, `parseFloat` μετατρέπουν ένα αλφαριθμητικό σε αριθμό και αν αποτύχει η μετατροπή τότε επιστρέφουν την τιμή `NaN` (Not a Number). Για να αποφύγουμε προβλήματα, από την πλευρά του χρήστη, σχετικά με την εισαγωγή χαρακτήρων αντί αριθμών και την εμφάνιση του μηνύματος `NaN` θα μπορούσαμε να κάνουμε τον παρακάτω έλεγχο :

```
sum = CalcSum(x, y )
if (isNaN(sum))
{
    alert("Not a valid number")
}
else
{
    document.getElementById('res').value = sum
}
```

Η JavaScript παρέχει την συνάρτηση `isNaN` με σκοπό να ελέγξουμε αν μια έκφραση (μεταβλητή, αλφαριθμητικό) είναι `NaN` ή όχι. Το αποτέλεσμα που επιστρέφει η συνάρτηση είναι αλήθεια (είναι αριθμός) ή ψέματα (δεν είναι αριθμός).

## Προσθήκη κουτιού επιλογής (select box) για την αριθμητική πράξη

|   |                                  |
|---|----------------------------------|
| Πρώτος αριθμός                              | <input type="text"/>             |
| Δευτερος αριθμός                            | <input type="text"/>             |
| Αποτέλεσμα                                  | <input type="text"/>             |
| Πράξη                                       | <input type="button" value="+"/> |
| <input type="button" value="Calculate..."/> |                                  |

1. Αλλαγές στη διεπαφή

α) Προσθέτουμε στο πίνακα μια νέα γραμμή

β) Προσθέτουμε το select box

```
<tr><td>Πράξη</td>  
  <td><select id="Select1">  
    <option selected="selected" value="+">+</option>  
    <option value="-">-</option>  
    <option value="*">*</option>  
    <option value="/">/</option>  
    <option></option>  
  </select>  
</td></tr>
```

## 2. Αλλαγές στη Calculate()

α) Αποθηκεύουμε την τιμή του Select Box

β) Εκτελούμε την CalcSum με μια επιπλέον μεταβλητή

```
.....  
y = document.getElementById('txt2').value  
z = document.getElementById('select1').value  
if(x == "" || y == "") {  
.....  
  
else {  
    sum = CalcSum(x, y, z )  
    if (isNaN(sum)) {  
.....
```

### 3. Αλλαγές στη CalcSum

α) Ανάλογα με την πράξη υπολογίζουμε το αποτέλεσμα

```
function CalcSum(a,b, praxi) {  
  if (praxi == "+") {  
    sum =Number(a) + Number(b)  
  }  
  else if(praxi == "-") {  
    sum =Number(a) - Number(b)  
  }  
  else if(praxi == "*") {  
    sum =Number(a) * Number(b)  
  }  
  else {  
    sum =Number(a) / Number(b)  
  }  
  
  return sum  
}
```

## Σενάριο JavaScript για εναλλαγή περιεχομένου εικόνας

Δημιουργούμε την βασική σελίδα με μια αναφορά η οποία ενεργοποιείται με το πάτημα μιας εικόνας.

Θα χρειαστούμε δύο εικόνες σε ίδιες διαστάσεις, τις οποίες θα ονομάσουμε



buttonUp.gif



buttonDown.gif

```
<html>
<head>
</head>
<body>
<a href="http://www.ekdd.gr"></a>
<br>
</body>
</html>
```

Στο τμήμα του head προσθέτουμε τις παρακάτω εντολές :

```
<script type="text/javascript">
<!--
buttonup = new Image;
buttonup.src = "buttonUp.gif";
buttondown = new Image;
buttondown.src = "buttonDown.gif";

function MouseOverRoutine(ButtonName) {
if (ButtonName=="TheButton ") {
    document.TheButton.src = buttondown.src;
} }

function MouseOutRoutine(ButtonName) {
if (ButtonName=="TheButton") {
    document.TheButton.src = buttonup.src;
} }
//-->
</script>
```

Στο τμήμα του υπερσυνδέσμου προσθέτουμε τις παρακάτω εντολές :

```
<a href="http://www.ekdd.gr"  
onmouseover="MouseOverRoutine('TheButton')"  
onmouseout="MouseOutRoutine('TheButton')"></a>
```

# Επαναλήψεις

Οι επαναλήψεις μας επιτρέπουν να εκτελέσουμε το ίδιο σύνολο από εντολές ενώ αληθεύει μια συνθήκη.

## Η επανάληψη for

Η σύνταξη της εντολής είναι :

```
for (ΑρχικήΤιμή; Συνθήκη; Βήμα)
{
    // εντολές
}
```

```
<html>
<body>
<script type="text/javascript">
var i=0

for (i=0;i<=5;i++) {
    document.write("The number is " + i)
    document.write("<br />")
}
</script>
</body>
</html>
```

## Η επανάληψη while

Η σύνταξη της εντολής είναι :

```
while (Συνθήκη)
{
    // εντολές
}
```

```
<html>
<body>
<script type="text/javascript">
var i=0

while (i<=5) {
    document.write("The number is " + i)
    document.write("<br />")
    i = i + 1
}
</script>
</body>
</html>
```

## Η επανάληψη do..while

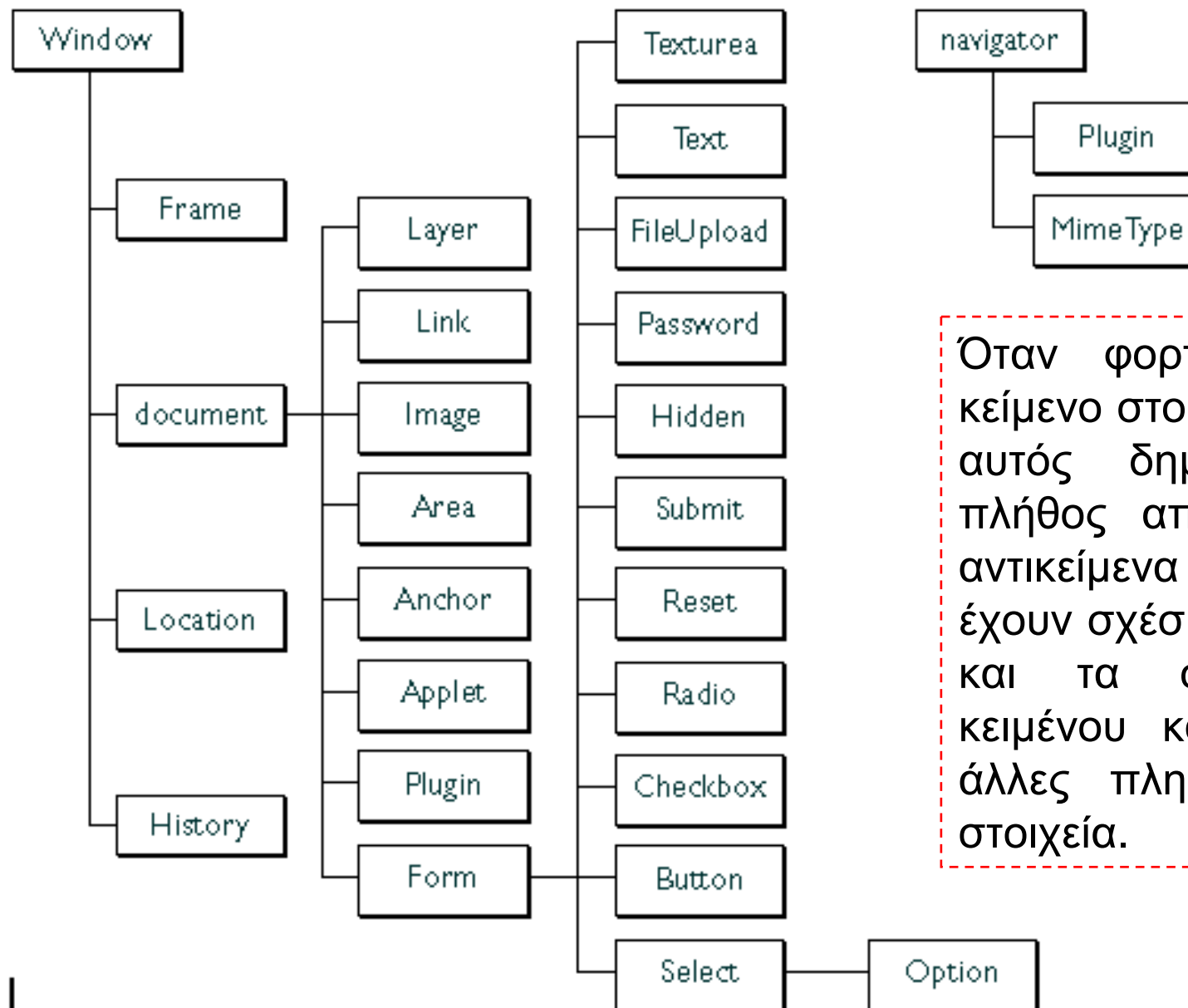
Η σύνταξη της εντολής είναι :

```
do
{
    // εντολές
} while (Συνθήκη)
```

```
<html>
<body>
<script type="text/javascript">
var i=0

do {
    document.write("The number is " + i)
    document.write("<br />")
    i = i + 1
} while (i<=5)
</script>
</body>
</html>
```

# Τα αντικείμενα της JavaScript



Όταν φορτώνουμε ένα κείμενο στο browser, τότε αυτός δημιουργεί ένα πλήθος από JavaScript αντικείμενα τα οποία έχουν σχέση με την δομή και τα στοιχεία του κειμένου καθώς και με άλλες πληροφορίες και στοιχεία.

**navigator** : έχει τις ιδιότητες και τα χαρακτηριστικά του προγράμματος πλοήγησης καθώς και για τα Mimetypes και τα Plugins που είναι εγκατεστημένα. (π.χ. navigator.userAgent, navigator.userLanguage)

**window** : είναι το υψηλότερο στην ιεραρχία αντικείμενο και περιγράφει τις ιδιότητες, τα χαρακτηριστικά καθώς και τα γεγονότα του παραθύρου μέσα στο οποίο εμφανίζεται η σελίδα ή των frames που υπάρχουν στην σελίδα. (π.χ. window.open, window.status)

**document** : περιέχει τις ιδιότητες του κειμένου που εμφανίζεται μέσα στο παράθυρο. Για παράδειγμα τα χρώματα, τους συνδέσμους, τις φόρμες κ.λπ. (document.backgroundColor, document.write)

**location**: έχει ιδιότητες για την τρέχουσα URL. (location.reload)

**history**: έχει ιδιότητες για τις διευθύνσεις που επισκέφτηκε ο χρήστης σε προηγούμενες πλοηγήσεις του. (π.χ. history.go(-1)).

## Παράδειγμα χρήσης πεδίων με την βοήθεια της ιεραρχίας του κειμένου

```
<HTML> <HEAD><TITLE>A Simple Menu</TITLE>
  <script type="text/javascript">
function doSomething()    {
  document.title = document.good.guest.value
  alert("Name: " + document.forms[0].elements[0].name)
  alert("Value: " + document.forms[0].elements[0].value)
  alert(navigator.userAgent)

if(window.confirm("Do you like a redirection to EKDDA web page?")) {
  window.location = "http://www.ekdd.gr"
}
}
</script>
</HEAD>
<BODY>
<form name="good">
  <input type="text" name="guest" />
  <input type="button" value="Press me..." onclick = "doSomething()" />
</form> </body> </html>
```

Διαφορετικός  
τρόπος  
πρόσβασης στην  
ίδια πληροφορία

## Η εντολή `window.open` και `window.close`

Προσθέστε μια ακόμη συνάρτηση για την δημιουργία ενός νέου παραθύρου. Η σύνταξη της εντολής είναι :

```
window.open('διεύθυνση','όνομα του παραθύρου','ιδιότητα1,ιδιότητα2,.....')
```

Οι ιδιότητες είναι ανά ζεύγος ονομα=τιμή

```
function doSomethingElse()
{
    var x =
window.open("", 'mywindow', 'width=400,height=200,toolbar=yes,
location=yes,
directories=no,status=yes,menubar=yes,scrollbars=yes,copyhistory=yes,
resizable=yes,left=0,top=100,screenX=0,screenY=100')
x.document.write("A new window")
x.document.write("<FORM><INPUT type='button' value='Close Window'
onClick='window.close()'></FORM>")
}
```

Η σύνταξη της εντολής είναι διαφορετική ανά browser για την τοποθέτηση του νέου παραθύρου στην οθόνη IE (left, top), NetScape( ScreenX, ScreenY)

## Πρόγραμμα ανίχνευσης του προγράμματος πλοήγησης

```
<html>
<head>
<script type="text/javascript">
function showBrowserInfo(){
var browser=navigator.appName
var version=parseFloat(navigator.appVersion)

if ((browser == "Microsoft Internet Explorer") && (version >= 4)) {
    alert("Your browser is good enough!")
}
else {
    alert("It's time to upgrade your browser!")
}
}
</script></head><body onload="ShowBrowserInfo()">
</body></html>
```

## Το αντικείμενο String

Η πιο συνηθισμένη αναπαράσταση μιας μεταβλητής ή το περιεχόμενο ενός Html στοιχείου είναι η αλφαριθμητική String. Το αντικείμενο String έχει ένα πλήθος από έτοιμες μεθόδους για την διαχείριση των χαρακτήρων. Ενδεικτικά αναφέρουμε :

```
<html><head></head> <body>
<script type="text/javascript">
var str="Hello"
document.write(str.length)           5
document.write(str.toUpperCase())    HELLO
document.write(str.toLowerCase())    hello
document.write(str.fontcolor("Red")) Hello
document.write(str.charAt(1))        e
document.write(str.indexOf('e'))     1
document.write(str.indexOf('el'))    1
document.write(str.search('ll'))     2
document.write(str.search(/EL/i))    1
document.write(str.substr(1,2))      el
str = str.concat(" World")
document.write(str)                   Hello World
document.write(str.split(""))        H,e,l,l,o, ,W,o,r,l,d
document.write(str.split(" "))       Hello,World
</script></body></html>
```

# JavaScript Cookies

Το cookie είναι μια πληροφορία που αποστέλλει ο εξυπηρετητής μαζί με την απόκριση του, στην αίτηση ενός πελάτη. Αυτήν την πληροφορία, ο browser του πελάτη την αποθηκεύει και την αποστέλλει σε κάθε νέα αίτηση προς τον ίδιο εξυπηρετητή. Με την JavaScript, μπορούμε να δημιουργήσουμε και να διαβάσουμε τις τιμές από τα cookies.

```
<html>
<head>
<title>Cookies</title>
</head>
<script type = "text/javascript" >
```

```
function setCookie(name, value, expireDays, path) {
var expireDate = new Date
    expireDate.setDate(expireDate.getDate()+expireDays)
    document.cookie= name + "=" + escape(value) + ";" +
        " expires=" + expireDate.toGMTString() + ";" +
        " path=" + path
}
```

```
function getCookie(cname) {  
  if (document.cookie.length>0) {  
    start=document.cookie.indexOf(cname + "=")  
    if (start > -1) {  
      start= start + cname.length+1  
      end=document.cookie.indexOf(";",start)  
      if (end== -1) {  
        end = document.cookie.length  
      }  
      return unescape(document.cookie.substring(start,end))  
    }  
  }  
  return ""  
}
```

```
function checkIT() {  
    var x = getCookie("user")  
    if (x== null || x == "") {  
        alert("It is your first visit")  
        setCookie("user","1","1","/")  
    }  
    else {  
        alert("you have visit : " + x)  
        setCookie("user",String(Number(x) + 1),"1","/")  
    }  
}
```

</script>

<body onload="checkIT()">

</body>

</html>

## Επικύρωση στοιχείων φόρμας (Form Validation)

Η JavaScript μπορεί να χρησιμοποιηθεί για την επικύρωση των δεδομένων μιας φόρμας πριν ο browser αποστείλει τα στοιχεία της φόρμας στον εξυπηρετητή.

```
<html>
<head><title>Form Validation</title>
</head>
<body>
<form name="data" method="post" action="test.asp" onsubmit="return
isFormValid()">
<table>
<tr><td>Όνομα</td><td><input type="text" name="uname" /></td></tr>
<tr><td>Επώνυμο</td><td><input type="text" name="username" /></td></tr>
<tr><td>Ηλικία</td><td><input type="text" name="uage" /></td></tr>
<tr><td>Email</td><td><input type="text" name="uemail" /></td></tr>
<tr><td colspan="2"><input type="submit" value="Αποστολή" /></td></tr>
</table>
</form>
</body>
</html>
```

Στο τμήμα του head προσθέτουμε τις παρακάτω εντολές :

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

function isValid(field, message) {
  if(field.value == "") {
    alert(message)
    return (false)
  }
  return (true)
}

function isNumeric(field, message) {
  if((field.value == "") || isNaN(Number(field.value))) {
    alert(message)
    return (false)
  }
  return (true)
}
```

```
function isEmail(field, message) {  
    if((field.value == "") || field.value.indexOf("@") < 0) {  
        alert(message)  
        return (false)  
    }  
    return (true)  
}
```

```
function isValidFormValid() {  
    if (isValid(document.data.uname, "You must fill the Name field.") &&  
        isValid(document.data.usurname, "You must fill the Surname field.)) {  
        if( isNumeric(document.data.uage, "The age must be a numeric  
value.") && isEmail(document.data.uemail, "The email is not valid.))  
            return true;  
    }  
  
    return false;  
}  
</script>
```