



Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Μάθημα: Τεχνικές Προγραμματισμού Υπολογιστών. (Εργαστηριακό μάθημα)

Καθηγητής: Πεφάνης Ευάγγελος

3) Εργαστηριακές σημειώσεις στην γλώσσα προγραμματισμού JavaScript.

Βρόχοι

Η δήλωση (For)

Το JavaScript έχει δύο μεθόδους για την εκτέλεση του ίδιου κώδικα αρκετές φορές. Χρησιμοποιείται κυρίως για iterating πάνω σε συστοιχίες ή αντικείμενα. Ας δούμε ένα παράδειγμα:

```
var i;  
for (i = 0; i < 3; i = i + 1)  
{  
    console.log(i);  
}
```

Αυτό θα εκτυπώσει τα εξής:

```
0  
1  
2
```

Η **for** δήλωση στο JavaScript έχει την ίδια σύνταξη με την Java και C. Διαθέτει τρία μέρη:

1. **Initialization** (Αρχικοποίηση) - Αρχικοποιεί τη μεταβλητή iterator *i*. Σε αυτό το παράδειγμα, αρχικοποιούμε την *i* στο 0.
2. **Condition** (Συνθήκη) - Όσο πληρούται η συνθήκη, ο βρόχος συνεχίζει να εκτελείται. Σε αυτό το παράδειγμα, ελέγχουμε ότι *i* είναι μικρότερο από 3.
3. **Increment** (Αύξηση) - Μια οδηγία που αυξάνει το iterator *i*. Στην περίπτωσή μας, το αυξάνουμε κατά 1 σε κάθε βρόχο.

Μπορούμε επίσης να γράψουμε μια συντομότερη αναφορά για τη δήλωση εισάγοντας τον ορισμό της μεταβλητής μέσα στο **for** βρόχο και αυξάνοντας χρησιμοποιώντας τον **++** χειριστή.

```
for (var i = 0; i < 3; i++)
{
    console.log(i);
}
```

Για να διατρέξουμε έναν πίνακα και να εκτυπώσουμε όλα τα μέλη του, συνήθως χρησιμοποιούμε τη `for` δήλωση. Ακολουθεί ένα παράδειγμα:

```
var myArray = ["A", "B", "C"];
for (var i = 0; i < myArray.length; i++)
{
    console.log("The member of myArray in index " + i + " is " + myArray[i]);
}
```

Αυτό εκτυπώνει τα περιεχόμενα της συστοιχίας:

```
The member of myArray in index 0 is A
The member of myArray in index 1 is B
The member of myArray in index 2 is C
```

Παρατηρήστε ότι χρησιμοποιήσαμε την `length` ιδιότητα ενός πίνακα, το οποίο επιστρέφει τον αριθμό των μελών στη συστοιχία, οπότε γνωρίζουμε πότε πρέπει να σταματήσουμε την αλλαγή.

Η δήλωση (*While*) ενώ

Η `while` δήλωση είναι μια πιο απλή έκδοση της `for` δήλωσης που ελέγχει εάν μια έκφραση αξιολογείται `true` και τρέχει όσο είναι `true`.

Για παράδειγμα:

```
var i = 99;
while (i > 0)
{
    console.log(i + " πακέτα ανταλλακτικών ");
    i -= 1;
}
```

break and continue δηλώσεις

Η **break** δήλωση επιτρέπει να σταματήσει η εκτέλεση ενός βρόχου. Το **break** μπορεί να μπει σε μια συνθήκη για να σπάσει τον βρόχο ελέγχοντας ότι μια συγκεκριμένη προϋπόθεση πληρούται.

```
var i = 99;
while (true)
{
    console.log(i + " πακέτα ανταλλακτικών ");
    i -= 1;
    if (i == 0)
    {
        break;
    }
}
```

Η **continue** εντολή παραλείπει τον υπόλοιπο βρόχο και μεταβαίνει στην αρχή του βρόχου. Για παράδειγμα, εάν θέλουμε να εκτυπώσουμε μόνο τους μονούς αριθμούς χρησιμοποιώντας μια **for** δήλωση, μπορούμε να κάνουμε τα εξής:

```
for (var i = 0; i < 100; i++)
{
    // ελέγχει εάν είναι ζυγός αριθμός
    if (i % 2 == 0)
    {
        continue;
    }
    // if we got here, then i is odd.
    console.log(i + " είναι μονός αριθμός.");
}
```