Web Programming 281

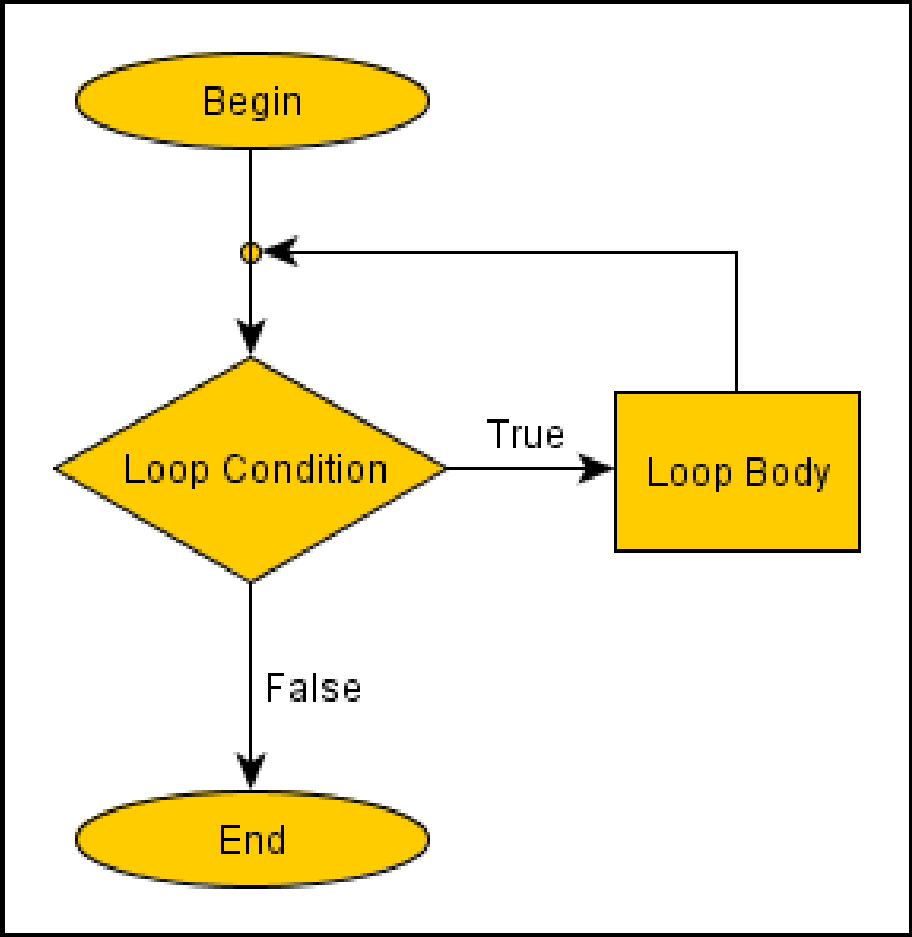
JavaScript Loops and Iteration

Outcomes

Students should understand the following outcomes, upon successful completion of this module:

- Looping
- for, forEach
- while, do while
- Iteration in JavaScript

Loops in JavaScript



Loops offer a quick and easy way to do something repeatedly.

Loops all essentially do the same thing: they repeat an action some number of times.

JavaScript supports all the necessary loops, including:
for, while, do…while, for..in, for..of etc.

Iteration in JavaScript

Iteration is a process where a set of instructions or structures are repeated in a sequence a specified number of times or until a condition is met.

When the first set of instructions is executed again, it is called an *iteration*.

When a sequence of instructions is executed in a repeated manner, it is called a *loop*.

So, to perform iteration in a program, you need to utilize a loop

Iteration in JavaScript

for Loop

Basic Syntax:

```
for ([initialization]; [condition]; [final-expression] ) {
  // statement
}
```

initialization - Runs before the first execution on the loop. This expression is commonly used to create counters. Variables created here are scoped to the loop. Once the loop has finished it's execution they are destroyed.

condition - Expression that is checked prior to the execution of every iteration. If omitted, this expression evaluates to true. If it evaluates to true, the loop's statement is executed. If it evaluates to false, the loop stops.

final-expression - Expression that is run after every iteration. Usually used to increment a counter. But it can be used to decrement a counter too.

statement - Code to be repeated in the loop

Iteration in JavaScript

Examples

Create a JavaScript program that iterates through integers from 0-8

```
for (var i = 0; i < 9; i++) {
  console.log(i);
}
```

Use a *break* statement to exit the loop before the condition expression evaluates to *false*.
Example: Create a program that will print odd numbers only, between 5 and 13, inclusive.

```
for (var num = 5; num < 15; num += 2) {
  if (num === 15) {
      break;
  }
  console.info(`${num} is an odd number`);
}
```

```
for (var num = 5; num < 15; num += 2) {
  if (num === 15) {
      break;
  }
  console.info(`${num} is an odd number`);
}
```

To print prime numbers

Iteration in JavaScript

for...in Loop

The *for...in* statement loops through the properties of an object.

It *iterates* over the *enumerable* properties of an object, in arbitrary order.

For each *distinct property*, statements can be executed.

Basic Syntax:

```
for (var in object) {
  //statements
}
```

An enumerable property is one that can be included in and visited during for..in loops (or a similar iteration of properties, like *Object.keys()*).

If a property isn't identified as enumerable, the loop will ignore that it's within the object.

Iteration in JavaScript

Examples

Create a JavaScript program that iterates through the following object:

cities = { "a": "Athens", "b": "Belgrade", "c": "Cairo" }, and log in the key value pairs one by one.

```
//object
let cities = { "a": "Athens", "b": "Belgrade", "c": "Cairo" }

// Iterate over the properties.
for (let value in cities) {
    console.log(`${value} : ${cities[value]}` );
}
```

Iteration in JavaScript

for...of Loop

The for...of statement creates a loop iterating over *iterable* objects (including *Array*, *Map*, *Set*, *strings* etc.), invoking a custom iteration hook with statements to be executed for the value of each distinct property.

The code inside the loop is executed for each element of the iterable object

Basic Syntax:

```
for (var of object) {
  //statement
}
```

Iteration in JavaScript

Examples

Create a JavaScript program that iterates through the following array:
names = ["fred", "tom", "bob", "Charlie"]; and log in the names in the array, one by one.

```
// Iterating over array
let names = [ "fred", "tom", "bob", "Charlie"];

for (let i of names) {
    console.log(i);
}
```

Create a JavaScript program that iterates through the following string:
I *greet = "Hello World!";*
and log in the characters in the string, one by one.

```
// Iterating over a string
let greet = "Hello World!";

for(let character of greet) {
    console.log(character);
}
```

Iteration in JavaScript

while Loop

The *while* loop starts by evaluating the *condition*. If the condition is *true*, the statement(s) is/are executed.

If the condition is *false*, the statement(s) is/are not executed.

After that, while loop ends.

Basic Syntax:

```
while (condition)
{
  //statement(s);
}
```

Iteration in JavaScript

Examples

Create a JavaScript program that uses a while loop to print numbers on the browser. The program should ask the user to type in the end value, and then loops printing values from number 1 until the end value set by the user.

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <script src="app.js"></script>
   <title>Document</title>
</head>
<body>

</body>
</html>
```

app.js

```
let endvalue = prompt("Enter end value: ");

var i = 1;
while(i <= endvalue) {
    document.write("<p>The number is " + i + "</p>");
    i++;
}
```

Iteration in JavaScript

do...while Loop

The *do...while* loop is closely related to *while* loop.

In the *do...while* loop, the *condition* is checked at the end of the loop.

With a *do-while* loop the block of code executed once, and then the condition is evaluated, if the condition is true, the statement is repeated as long as the specified condition evaluated to is true.

Basic Syntax:

```
do {
  // Code to be executed
}
while(condition);
```

Iteration in JavaScript

Examples

Create a JavaScript program that uses a do…while loop to print numbers on the browser. The program should ask the user to type in the end value, and then loops printing values from number 1 until the end value set by the user.

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <script src="app.js"></script>
   <title>Document</title>
</head>
<body>

</body>
</html>
```

app.js

```
let endValue = prompt("Enter end value: ");
var i = 1;
do {
    document.write("<p>The number is " + i + "</p>");
    i++;
}
while(i <= endValue);
```

Iteration in JavaScript

Question

Create a JavaScript program that uses a do…while loop to insert values from 1 to 10 in an empty array. Use a for…of loop to iterates and display the values in the array.

More exercise questions are on Moodle

BELGIUM CAMPUS iTversity
It's the way we're wired

Thank You!

THE END

info@belgiumcampus.ac.za +27 10 593 53 68

/belgiumcampusSA
#Belgium Campus
/belgiumcampus