

Coding in JavaScript – repetition structures or loops

Loops execute a block of code a specified number of times, or while a specified condition is true. Often when you write code, you want the same block of code to run over and over again in a row. Instead of adding several almost equal lines in a script we can use loops to perform a task like this.

In JavaScript, there are two different kind of loops:

- **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

The for Loop

The for loop is used when you know in advance how many times the script should run.

Syntax

```
for (variable=startvalue;variable<=endvalue;variable=variable+increment)
{
  code to be executed
}
```

Example

The example below defines a loop that starts with i=0. The loop will continue to run as long as i is less than, or equal to 5. i will increase by 1 each time the loop runs.

Note: The increment parameter could also be negative, and the <= could be any comparing statement.

```
var i=0;
for (i=0;i<=5;i++)
{
  println("The number is " + i);
}
```

Loops execute a block of code a specified number of times, or while a specified condition is true.

The while Loop

The while loop loops through a block of code while a specified condition is true.

Syntax

```
while (variable <= endvalue)
{
  code to be executed
}
```

Note: The <= could be any comparing operator.

Example

The example below defines a loop that starts with `i=0`. The loop will continue to run as long as `i` is less than, or equal to 5. `i` will increase by 1 each time the loop runs:

```
var i=0;
while (i <= 5)
{
    println("The number is " + i);
    i++;
}
```

The do...while Loop

The do...while loop is a variant of the while loop. This loop will execute the block of code ONCE, and then it will repeat the loop as long as the specified condition is true.

Syntax

```
do
{
    code to be executed
}
while (variable<=endvalue);
```

Example

The example below uses a do...while loop. The do...while loop will always be executed at least once, even if the condition is false, because the statements are executed before the condition is tested:

```
var i=0;
do
{
    println("The number is " + i);
    i++;
}
while (i<=5);
```

The break Statement

The break statement will break the loop and continue executing the code that follows after the loop (if any).

Example

```
var i=0;
for (i=0;i<=10;i++)
{
    if (i==3)
    {
        break;
    }
}
```

```
    }  
    println("The number is " + i);  
}
```

The continue Statement

The continue statement will break the current loop and continue with the next value.

Example

```
var i=0  
for (i=0;i<=10;i++)  
{  
    if (i==3)  
    {  
        continue;  
    }  
    println("The number is " + i);  
}
```

JavaScript For...In Statement

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

Syntax

```
for (variable in object)  
{  
    code to be executed  
}
```

Note: The code in the body of the for...in loop is executed once for each property.

Example

Looping through the properties of an object:

```
var person={fname:"John",lname:"Doe",age:25};  
var x;  
  
for (x in person)  
{  
    println(person[x] + " ");  
}
```