

JavaScript: Loops



JavaScript performs several types of repetitive operations, called "looping". Loops are set of instructions used to repeat the same block of code till a specified condition returns false or true depending on how you need it. To control the loops you can use counter variable that increments or decrements with each repetition of the loop.

JavaScript supports two loop statements: for and while. The For statements are best used when you want to perform a loop a specific number of times. The While statements are best used to perform a loop an undetermined number of times. In addition, you can use the break and continue statements within loop statements.

- [The For Loop](#)
- [The While Loop](#)
- [Break and Continue Statements](#)

The For Loop

The For loop is executed till a specified condition returns false. It has basically the same syntax then in other languages. It takes 3 arguments and looks as follows:

```
for (initialization; condition; increment)
{
    // statements
}
```

When the For loop executes, the following occurs:

1. The initializing expression is executed. This expression usually initializes one or more loop counters, but the syntax allows an expression of any degree of complexity.



Search

Search

or browse [popular tags](#)



JS Tutorial

- [What is JavaScript?](#)
- [What can JavaScript do?](#)
- [The first script](#)
- [Variables](#)
- [Arrays](#)
- [Functions](#)

- [Loops](#)
- [Conditional Statements](#)
- [Form Validation](#)
- [How to disable the Enter key](#)
- [Set Focus to First Available Form Control](#)
- [Show different content to search engine visitors](#)
- [BBCode editor](#)
- [Try It Online](#)



Subscription

Sign up for the free email newsletter for new tips, tutorials and more. Enter your email address below, and then click the button.

[Privacy Policy](#)

Subscribe



2. The condition expression is evaluated. If the value of condition is true, the loop statements execute. If the value of condition is false, the for loop terminates.
3. The update expression increment executes.
4. The statements execute, and control returns to step 2.

The following example generates a multiplication table 2 through 9. Outer loop is responsible for generating a list of dividends, and inner loop will be responsible for generating lists of dividers for each individual number:

```
<script type="text/javascript">
document.write("<h1>Multiplication table</h1>");
document.write("<table border=2 width=50%");

for (var i = 1; i <= 9; i++ ) { //this is the outer loop
  document.write("<tr>");
  document.write("<td>" + i + "</td>");

  for ( var j = 2; j <= 9; j++ ) { // inner loop
    document.write("<td>" + i * j + "</td>");
  }

  document.write("</tr>");
}

document.write("</table>");
</script>
```

Select Action ▼

Next example creates a function containing the For statement that counts the number of selected options in a list. The For statement declares the variable i and initializes it to zero. It checks that i is less than the number of options in the Select object, performs the succeeding if statement, and increments i by one after each pass through the loop.

```
<script type="text/javascript">
function howMany (selectItem) {
  var numberSelected=0

  for (var i=0; i < selectItem.options.length; i++) {
    if (selectItem.options[i].selected == true)
      numberSelected++;
  }

  return numberSelected
}
```

Select Action ▼

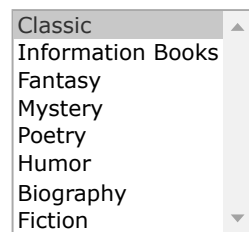
```

</script>

<form name="selectForm">
<p>Choose some book types, then click the button below:</p>
<select multiple name="bookTypes" size="8">
<option selected> Classic </option>
<option> Information Books </option>
<option> Fantasy </option>
<option> Mystery </option>
<option> Poetry </option>
<option> Humor </option>
<option> Biography </option>
<option> Fiction </option>
</select>
<input type="button" value="How many are selected?"
onclick="alert ( 'Number of options selected: ' + howMany(document.selectForm.bookTypes)) ">
</form>

```

Choose some book types and click the button below to see what the script in the preceding example does:



How many are selected?

At last let's consider the example that uses 2 variables. One to add all the numbers from 1 to 10. The other to add only the even numbers.

```

<script type="text/javascript">

var total = 0;
var even = 0;

for ( x = 1, y = 1; x <= 10; x++, y++ ) {
  if ( ( y % 2 ) == 0 ) {
    even = even + y;
  }
  total = total + x;
}

document.write ( "The total sum: " + total + "<br>" );

```

Select Action ▼

```
document.write ( "The sum of even values: " + even );  
</script>
```

[↑ Back to top](#)

The While Loop

The While loop is another commonly used loop after the For loop. The while statement repeats a loop as long as a specified condition evaluates to true. If the condition becomes false, the statements within the loop stop executing and control passes to the statement following the loop. The while statement looks as follows:

```
while (condition)  
{  
    // statements  
}
```

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 10. i will increase by 1 each time the loop runs.

```
<script type="text/javascript">  
  
var i=0;  
  
while (i<=10) //Output the values from 0 to 10  
{  
    document.write(i + "<br>")  
    i++;  
}  
  
</script>
```

Select Action ▼

Now let's consider a more useful example which creates drop-down lists of days, months and years. You can use this code for registration form, for example.

```
<script type="text/javascript">  
  
var month_array = new Array();  
  
month_array[0] = "January";  
month_array[1] = "February";  
month_array[2] = "March";  
month_array[3] = "April";  
month_array[4] = "May";  
month_array[5] = "June";  
month_array[6] = "July";
```

Select Action ▼

```

month_array[7] = "August";
month_array[8] = "September";
month_array[9] = "October";
month_array[10] = "November";
month_array[11] = "December";

document.write('<select name="day">');
var i = 1;
while ( i <= 31 ) {
    document.write('<option value=' + i + '>' + i + '</option>');
    i++;
}
document.write('</select>');

document.write('<select name="month">');
var i = 0;
while ( i <= 11 ) {
    document.write('<option value=' + i + '>' + month_array[i] + '</option>');
    i++;
}
document.write('</select>');

document.write('<select name="year">');
var i = 1900;
while ( i <= 2005 ) {
    document.write('<option value=' + i + '>' + i + '</option>');
    i++;
}
document.write('</select>');

</script>

```

Note: Make sure the condition in a loop eventually becomes false; otherwise, the loop will never terminate.

[↑Back to top](#)

Break and Continue Statements

Sometimes you may want to let the loops start without any condition, and allow the statements inside the brackets to decide when to exit the loop. There are two special statements that can be used inside loops: break and continue. The break statement terminates the current while or for loop and continues executing the code that follows after the loop (if any). A continue statement terminates execution of the block of statements in a while or for loop and continues execution of the loop with the next iteration.

Example below shows how to use these statements:

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

Select Action ▼

```
document.write("<p><b>Example of using the break statement:</b></p>");

var i = 0;
for (i=0; i<=10; i++) {
    if (i==3){break}
    document.write("The number is " + i);
    document.write("<br />");
}

document.write("<p><b>Example of using the continue statement:</b></p>");

var i = 0;
for (i=0; i<=10; i++) {
    if (i==3){continue}
    document.write("The number is " + i);
    document.write("<br />")
}

</script>
```

[⬆️ Back to top](#)

Tags:

Add To:  dzone |  digg |  del.icio.us |  stumbleupon

in select statement its a good concept.....

Posted by irfan | 15 Oct 2009 01:39:15

