w3schools.com

THE WORLD'S LARGEST WEB DEVELOPER SITE

☰ HTML CSS JAVASCRIPT MORE ▾ REFERENCES ▾

JavaScript Break and Continue

❮ Previous Next ❯

The break statement "jumps out" of a loop.

The continue statement "jumps over" one iteration in the loop.

The Break Statement

You have already seen the break statement used in an earlier chapter of this tutorial. It was used to "jump out" of a switch() statement.

The break statement can also be used to jump out of a loop.

The **break statement** breaks the loop and continues executing the code after the loop (if any):

Example

```
for (i = 0; i < 10; i++) {
    if (i === 3) { break; }
    text += "The number is " + i + "<br>";
}
```

Try it Yourself »

The Continue Statement

The **continue statement** breaks one iteration (in the loop), if a specified condition occurs, and continues with the next iteration in the loop.

This example skips the value of 3:

Example

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

Try it Yourself »

JavaScript Labels

To label JavaScript statements you precede the statements with a label name and a colon:

```
label:
statements
```

The break and the continue statements are the only JavaScript statements that can "jump out of" a code block.

Syntax:

```
break labelname;

continue labelname;
```

The continue statement (with or without a label reference) can only be used to **skip one loop iteration**.

The break statement, without a label reference, can only be used to **jump out of a loop or a switch**.

With a label reference, the break statement can be used to **jump out of any code block**:

Example

```
var cars = ["BMW", "Volvo", "Saab", "Ford"];
list: {
    text += cars[0] + "<br>";
    text += cars[1] + "<br>";
    text += cars[2] + "<br>";
    break list;
    text += cars[3] + "<br>";
    text += cars[4] + "<br>";
    text += cars[5] + "<br>";
}
```

Try it Yourself »

A code block is a block of code between { and }.