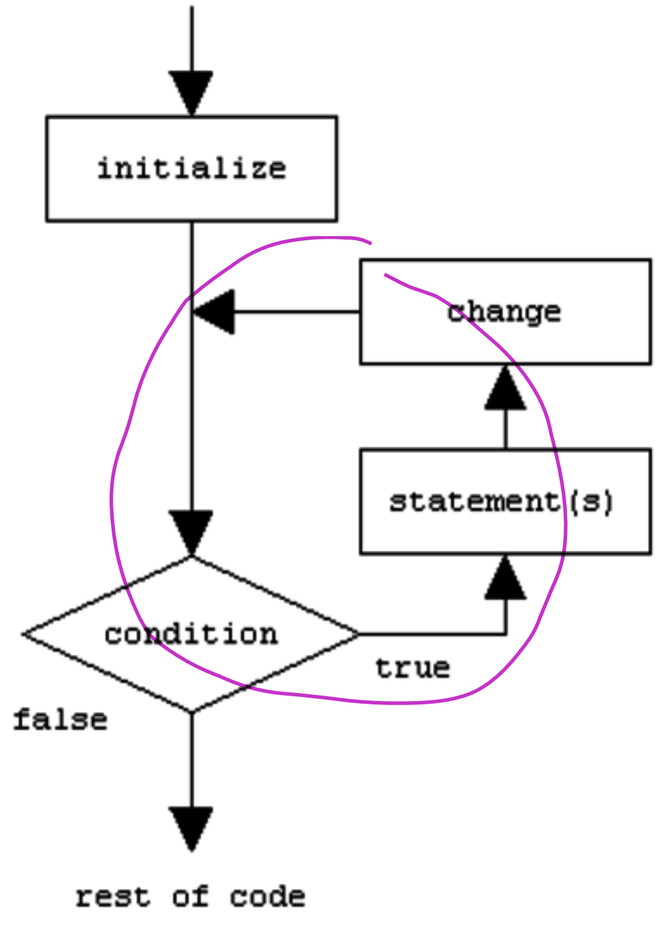
CD Studio: JavaScript
Week 5

Loops & Iteration

09.30.21

Diagram of a loop



Loops allow us to give a set of instructions to the browser which are then repeated automatically. The path of a loop is determined by testing for predefined conditions. When the conditions are no longer met the loop ends.

What do loops do?

```
console.log("loop 0");
console.log("loop 1");
console.log("loop 2");
console.log("loop 3");
```

```
for (var i = 0; i < 4; i++) {
   console.log("loop " + i);
}
```

A loop can be used to compute actions that would be tedious or even impossible to write by hand.

What do loops do?

```
console.log("loop 0");
console.log("loop 1");
console.log("loop 2");
console.log("loop 3");

for (var i = 0; i < 4; i++) {
    console.log("loop " + i);
}
```

*Note:
"i" is a commonly used variable name for counters. It is short for integer. You could replace "i" with your choice of any string.

A loop can be used to compute actions that would be tedious or even impossible to write by hand.

*Note:
"i++" or any var++ is a shorthand for writing out:

```
  i = i + 1;
```

or

```
  i += 1;
```

```
for (var i = 0; i < 4; i++) {
   console.log("loop " + i);
}
```

0");
1");
2");
3");

A loop can be used to compute actions that would be tedious or even impossible to write by hand.

```
// Initiate an infinite loop
while (0==0) {
    browserCrash();
    // execute code forever
}
```

If a condition always equals true, the loop will never terminate. This is called an infinite loop and should be avoided as it is likely to result in the browser and/or entire computer crashing.

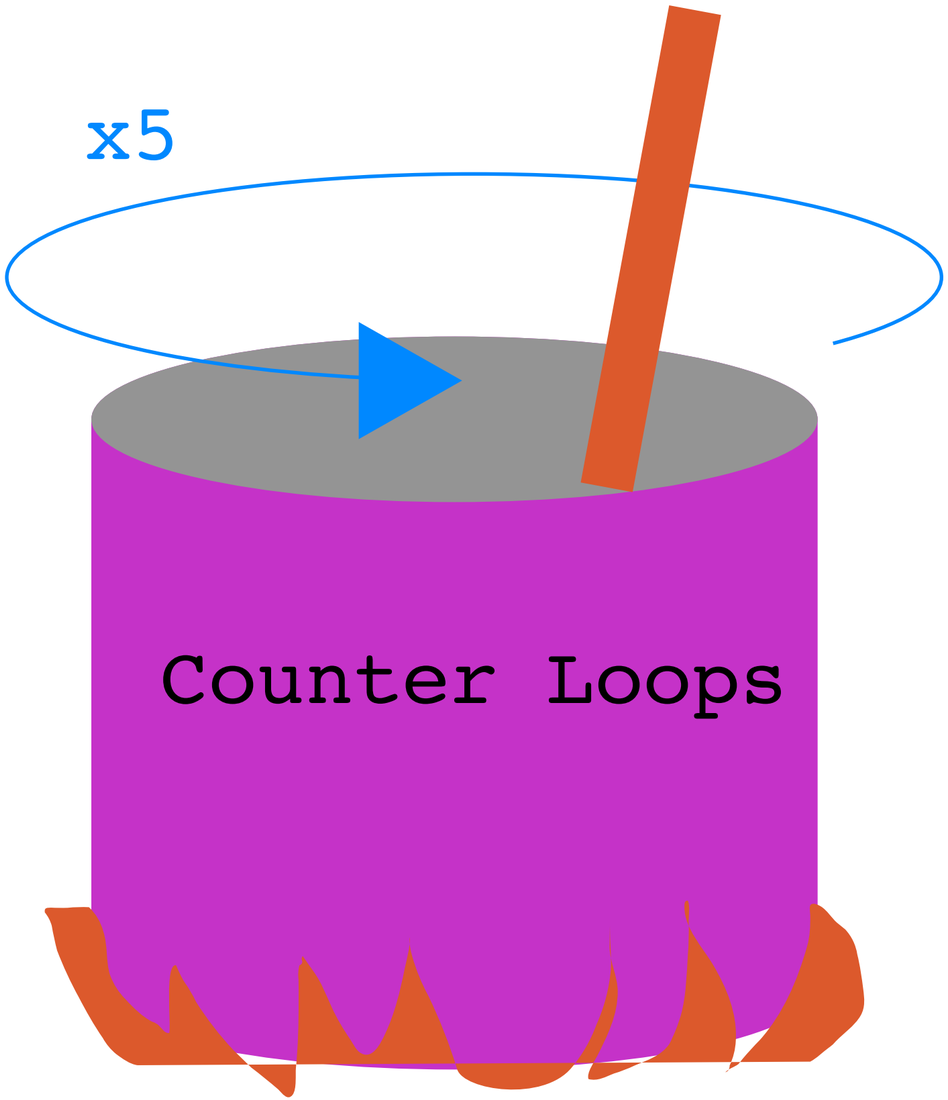
Counter Loops

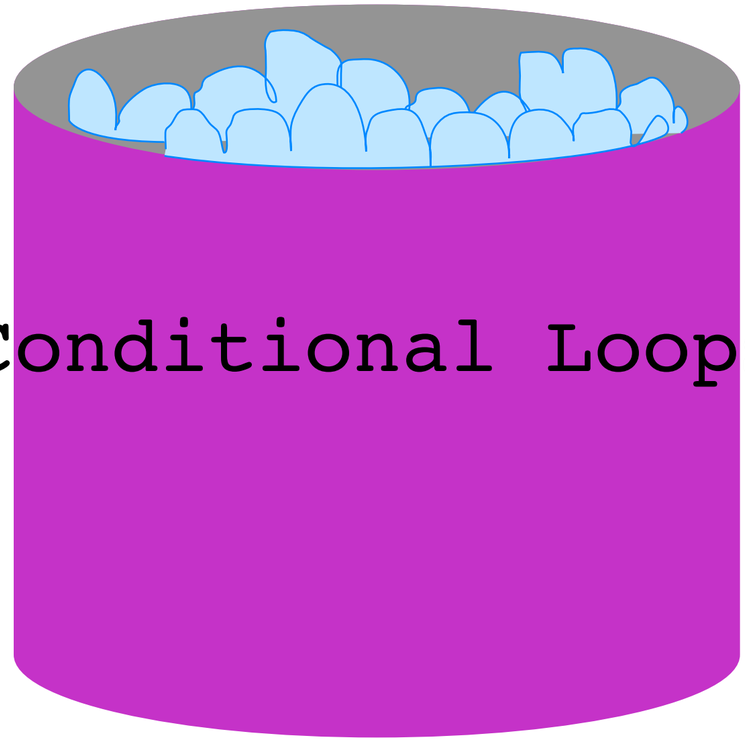
Counter loops repeat the instructions until the counter reaches the set limit.

Conditional Loops

Conditional loops repeat the instructions until a defined condition is met.

Cooking Examples





For 5 times stir pot.

While the water is not boiling, keep the heat function on high.

For Statements

```
for ([initialExpression]; [conditionExpression]; [incrementExpression]){
    // execute code
}
```

```
for (var i = 0; i < 4; i++){
    // execute code
}
```

Count based for loops require a loop counter, a condition to be met, and a method of updating the loop counter value.

```
for (var i = 0; i < 10; i++) {
    var newDiv = document.createElement("div");
    newDiv.innerHTML = "New div number " + i;
    document.body.appendChild(newDiv);
}
```

This counter FOR loop script will add 10 new div's (numbered 0-9) to the HTML <body>.

```
var i = 0
while (i < 10) {
    var newDiv = document.createElement("div");
    newDiv.innerHTML = "New div number " + i;
    document.body.appendChild(newDiv);
    i++
}
```

This conditional WHILE loop script will add 10 new div's (numbered 0-9) to the HTML <body>.

```
var randomResult = 0;
var loopCount = 0;

while (randomResult !== 8) {
    randomResult = Math.floor(Math.random() * 10);
    console.log("Random result: " + randomResult);
    loopCount++;
}

console.log("Final random result: " + randomResult);
console.log("Total number of loops: " + loopCount);
```

This conditional WHILE loop script will continue drawing random numbers until it picks the number 8.

```
var randomResult = 0;
var loopCount = 0;

while (randomResult !== 8) {
    randomResult = Math.floor(Math.random() * 10);
    console.log("Random result: " + randomResult);
    if (randomResult == 6) {
        break;
    }
    loopCount++;
}

console.log("Final random result: " + randomResult);
console.log("Total number of loops: " + loopCount);
```

This conditional WHILE loop script will continue drawing random numbers until it picks the number 8 or 6.

```
var fruits = ["apple", "peach", "banana"];
```

```
for (var i = 0; i < fruits.length; i++)
{
   console.log(fruits[i]);
}
```

```
//for (variable of object)
for (var fruit of fruits) {
   console.log(fruit);
}
```

The for...of method allows a shorthand method of iterating over each item in an array or object. It functions similarly to a counter loop, where the condition to be met is "i < array.length".