

# The While Loop

Moving forward, let's discuss how to write loops in JavaScript. This lesson will cover all the basic concepts related to while loops.

## WE'LL COVER THE FOLLOWING ^

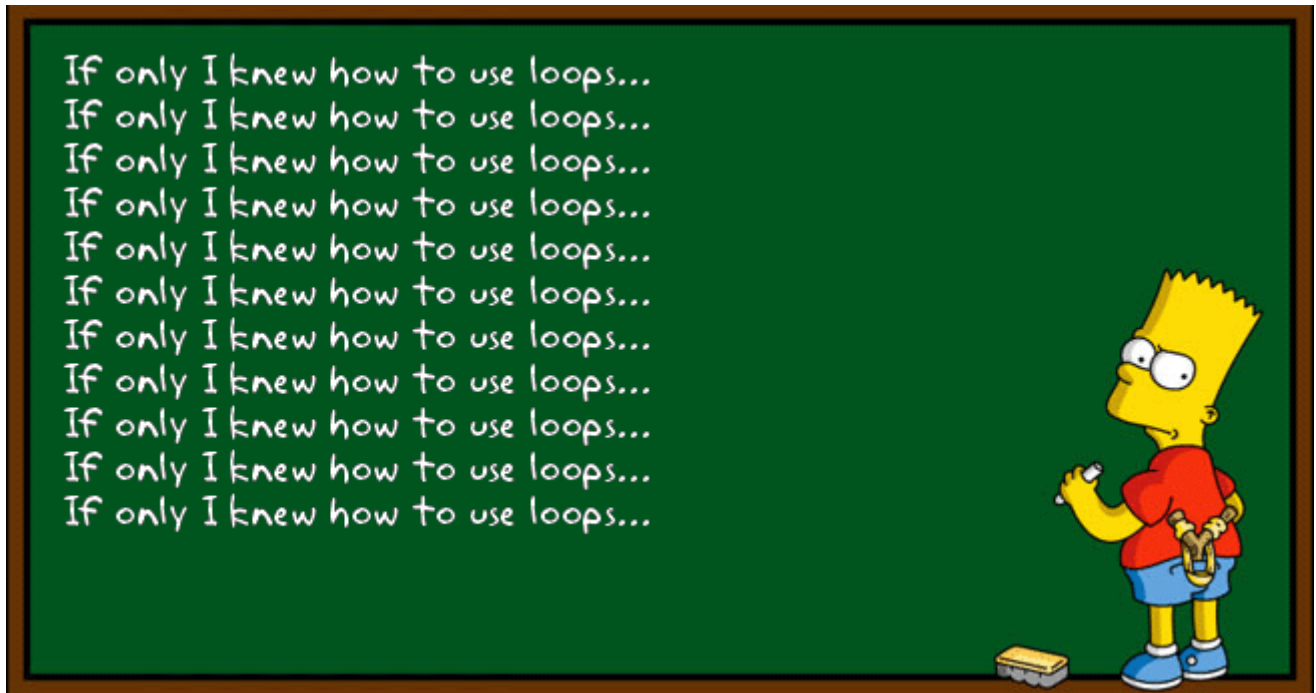
- The **while** Loop
  - Example
- How it works

If you wanted to write code that displayed numbers between 1 and 5, you could do it with what you've already learned:

```
console.log(1);  
console.log(2);  
console.log(3);  
console.log(4);  
console.log(5);
```



This is pretty tiresome though and would be much more complex for lists of numbers between 1 and 1000, for example. How can you accomplish the same thing more simply? JavaScript lets you write code inside a loop that executes repeatedly until it's told to stop. Each time the code runs, it's called an iteration.



## The **while** Loop #

A **while** loop lets you repeat code while a certain condition is true.

Example #

Here's a sample program written with a **while** loop.

```
let number = 1;
while (number <= 5) {
  console.log(number);
  number++;
}
```



Just like the previous one, this code block shows all integer numbers between 1 and 5.

## How it works #

You'll use the following syntax to write a while loop.

```
while (condition) {
  // Code to run while the condition is true
}
```



Before each loop iteration, the condition in parentheses is evaluated to determine whether it's true or not. The code associated with a loop is called its *body*.

- If the condition's value is `true`, the code in the `while` loop's body runs. Afterwards, the condition is re-evaluated to see if it's still true or not. The cycle continues!
- If the condition's value is `false`, the code in the loop stops running or doesn't run.

The loop body must be placed within curly braces, except if it's only one statement. For now, always use curly braces for your loops.