Frontend Programming

Javascript

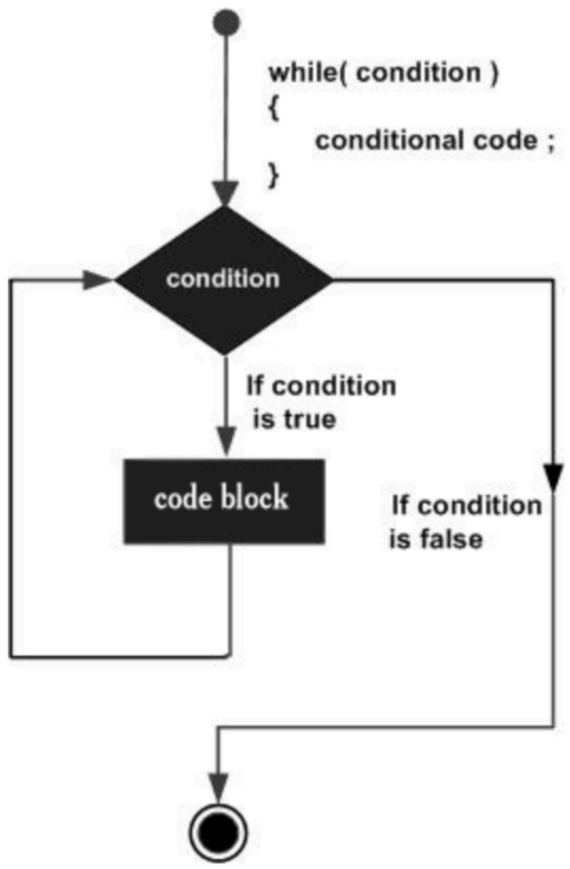
- DAY 2-

MARTHA SUTOPO

While Loop

G2 | Academy

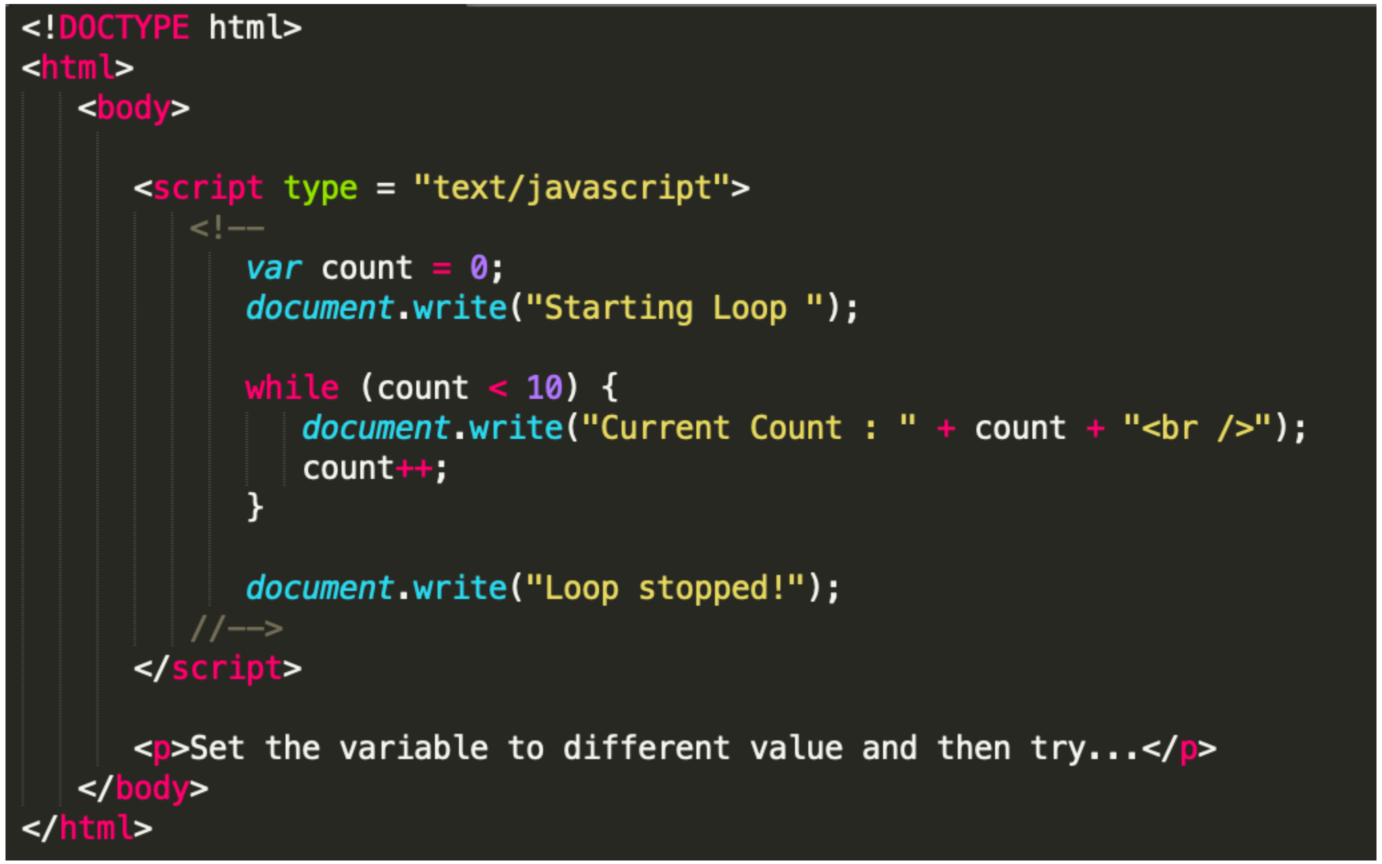
JavaScript - While Loop



```
while (expression) {
    Statement(s) to be executed if expression is true
}
```

G2 | Academy

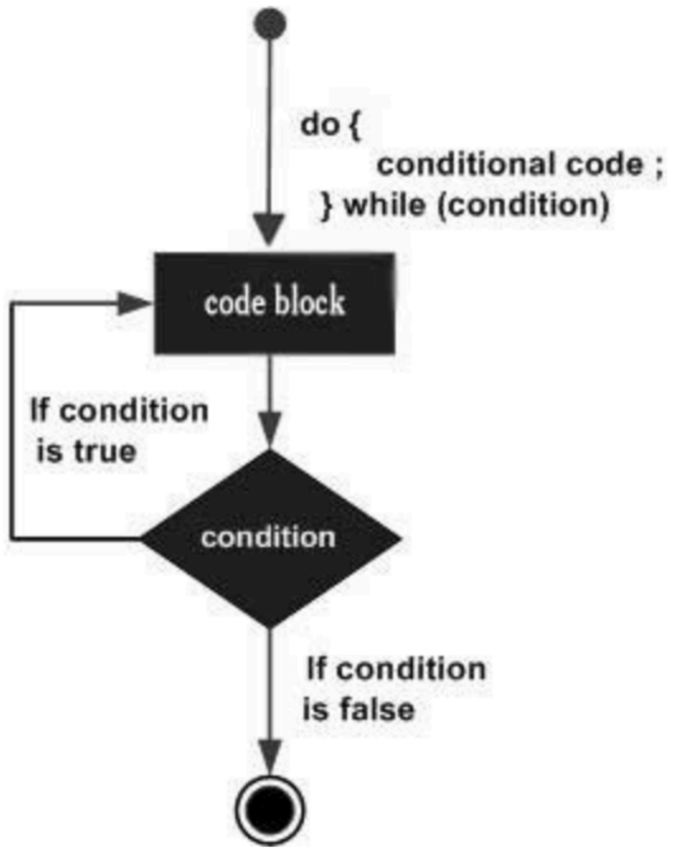
JavaScript - While Loop

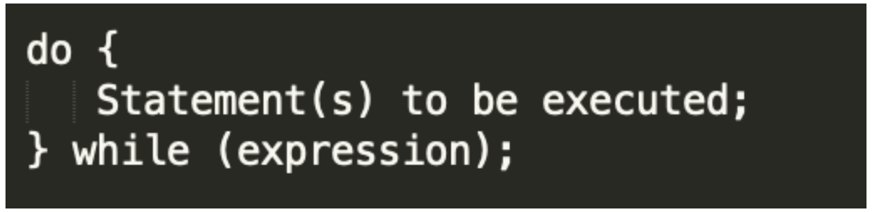


```
<!DOCTYPE html>
<html>
   <body>

      <script type = "text/javascript">
         <!--
            var count = 0;
            document.write("Starting Loop ");

            while (count < 10) {
               document.write("Current Count : " + count + "<br />");
               count++;
            }

            document.write("Loop stopped!");
         //-->
      </script>

      <p>Set the variable to different value and then try...</p>
   </body>
</html>
```

G2 | Academy

JavaScript - Do...While Loop

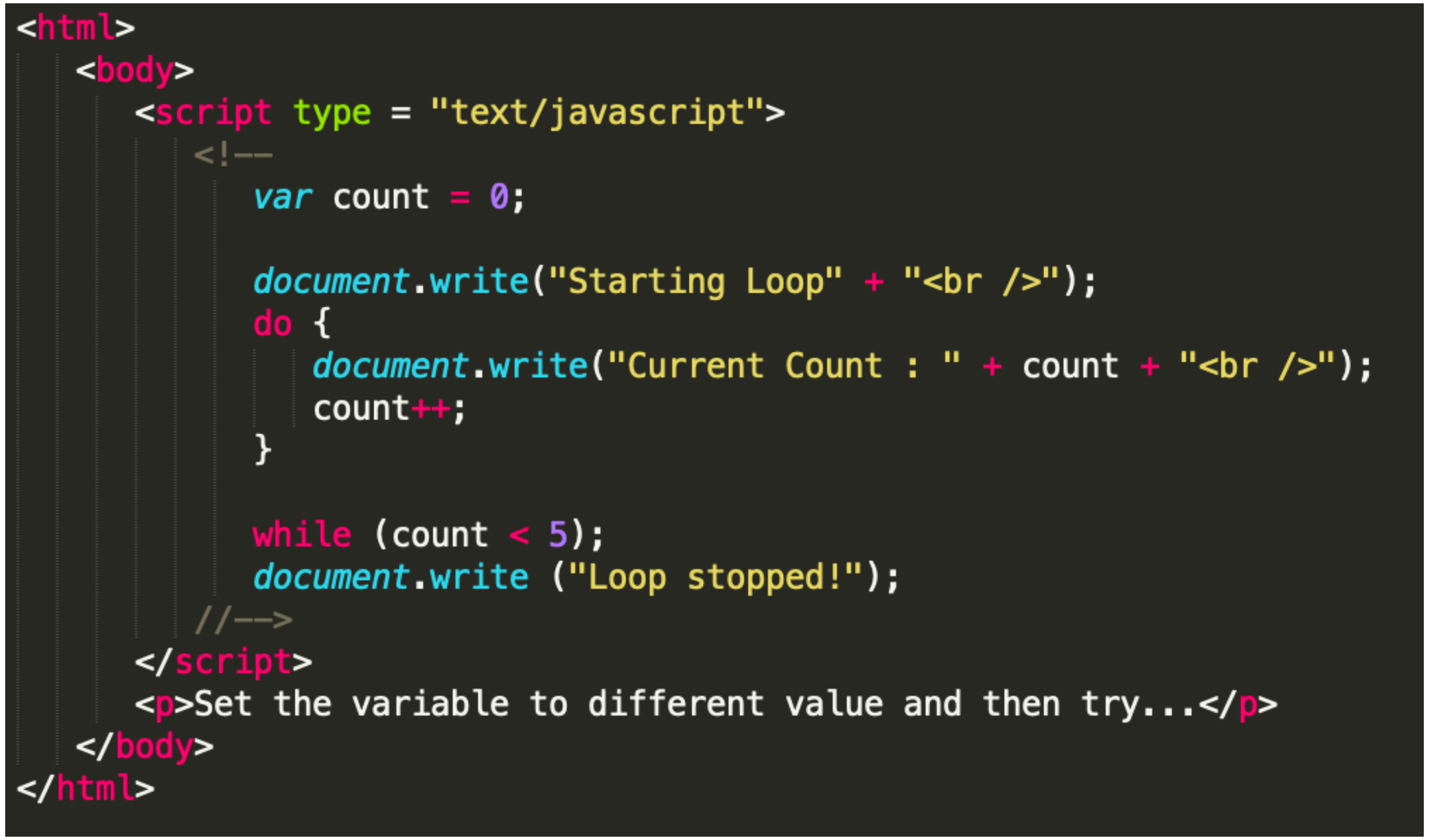




```
do {
    Statement(s) to be executed;
} while (expression);
```

G2 | Academy

JavaScript - Do...While Loop



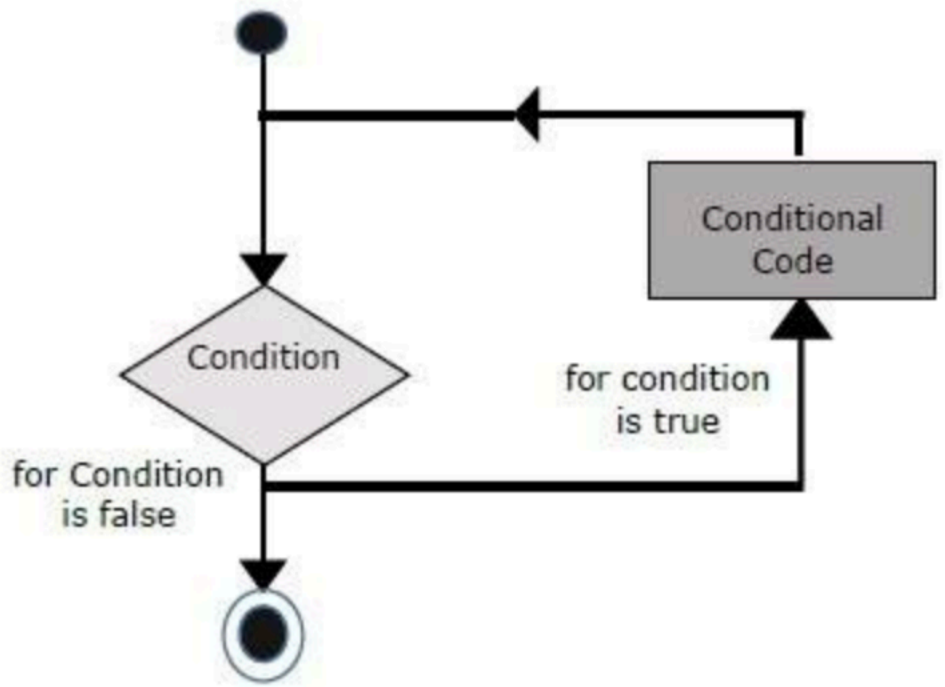
```html
<html>
    <body>
        <script type = "text/javascript">
            <!--
                var count = 0;

                document.write("Starting Loop" + "<br />");
                do {
                    document.write("Current Count : " + count + "<br />");
                    count++;
                }

                while (count < 5);
                document.write ("Loop stopped!");
            //-->
        </script>
        <p>Set the variable to different value and then try...</p>
    </body>
</html>
```

G2 | Academy

For Loop

G2 | Academy

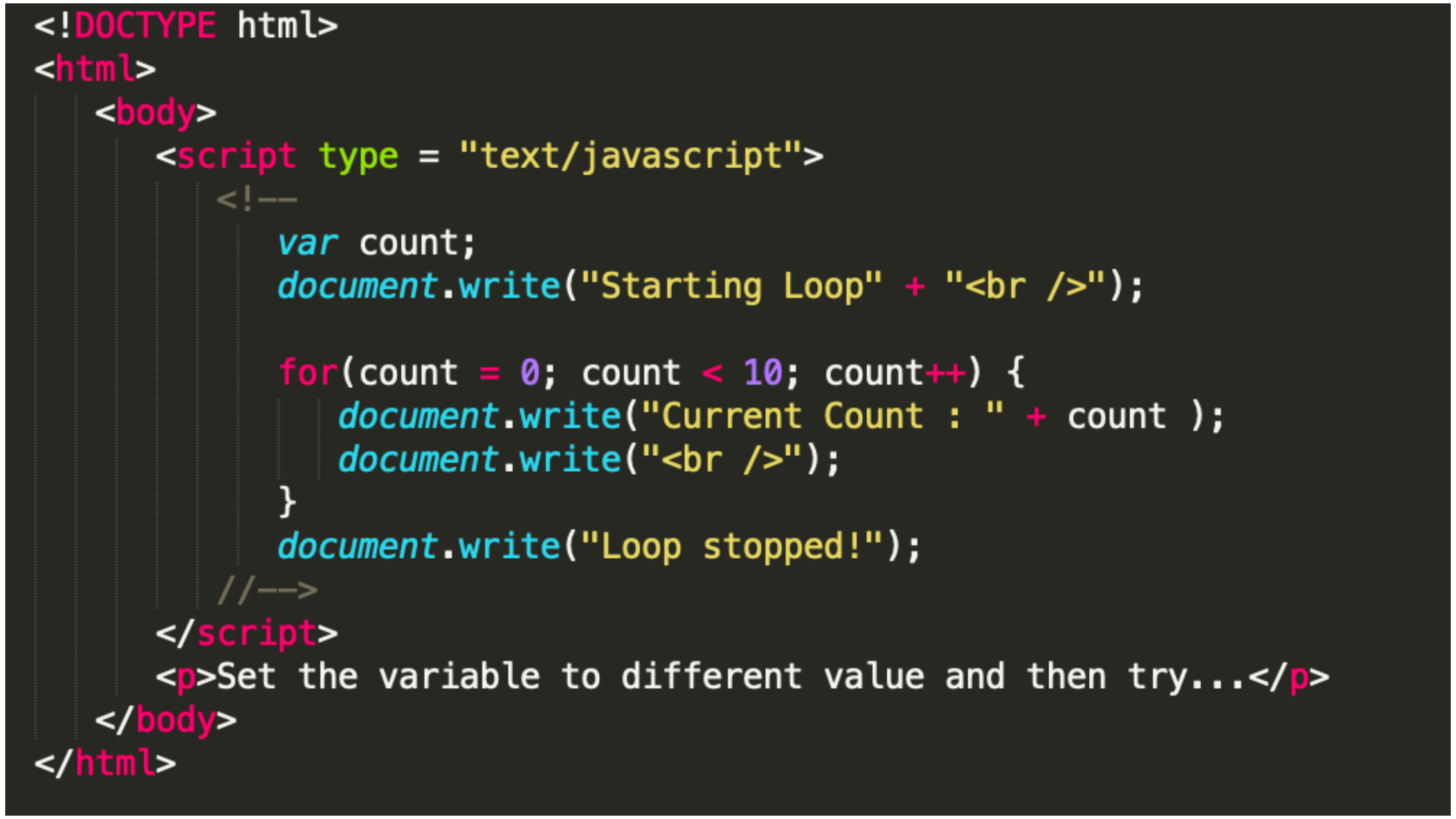
JavaScript - For Loop



```
for (initialization; test condition; iteration statement) {
    Statement(s) to be executed if test condition is true
}
```

G2 | Academy

JavaScript - For Loop



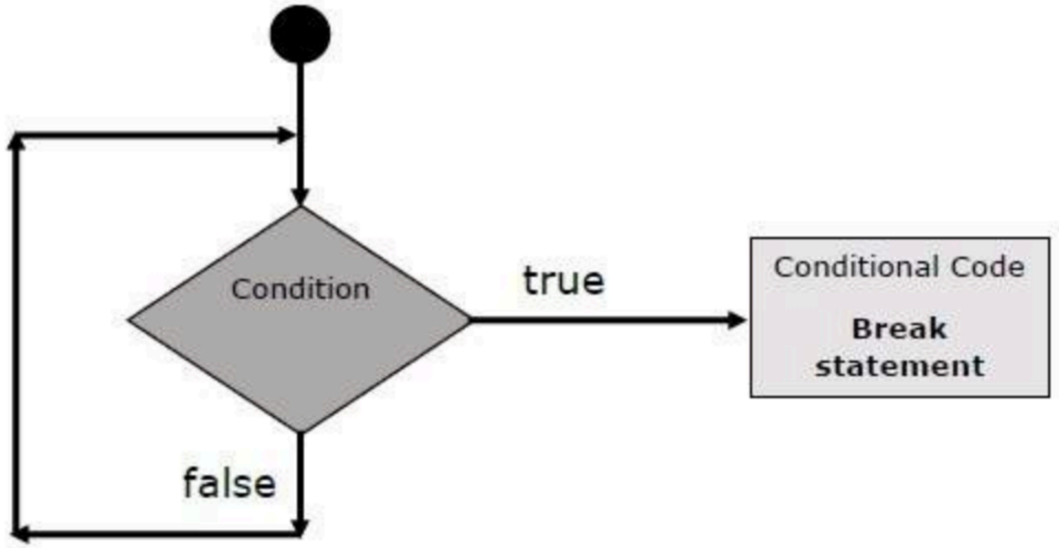
```
<!DOCTYPE html>
<html>
   <body>
      <script type = "text/javascript">
         <!--
            var count;
            document.write("Starting Loop" + "<br />");

            for(count = 0; count < 10; count++) {
               document.write("Current Count : " + count );
               document.write("<br />");
            }
            document.write("Loop stopped!");
         //-->
      </script>
      <p>Set the variable to different value and then try...</p>
   </body>
</html>
```

G2 | Academy

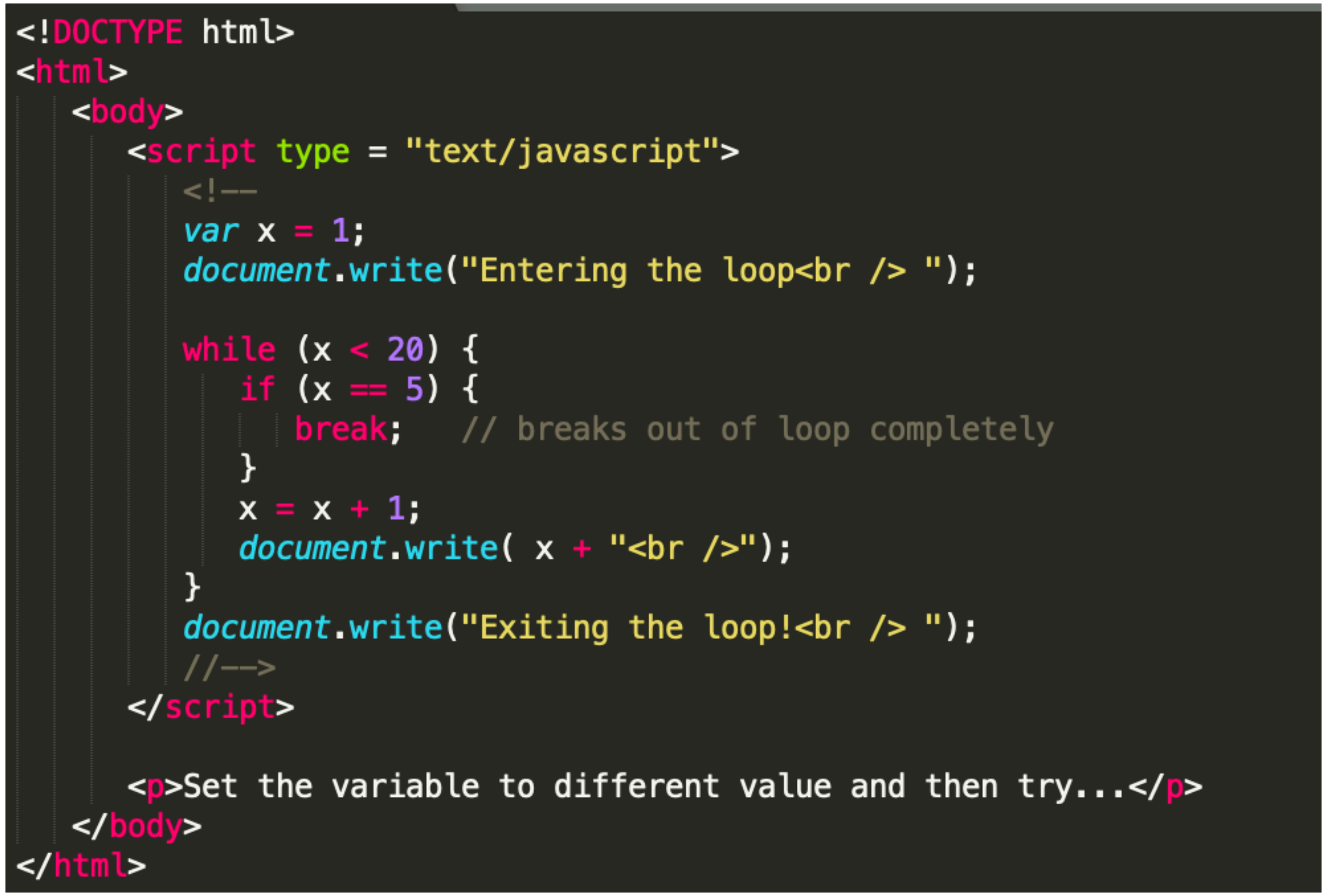
Break & Continue

JavaScript - Break

- The break is used to **exit a loop early**, breaking out of the enclosing curly braces



JavaScript - Break



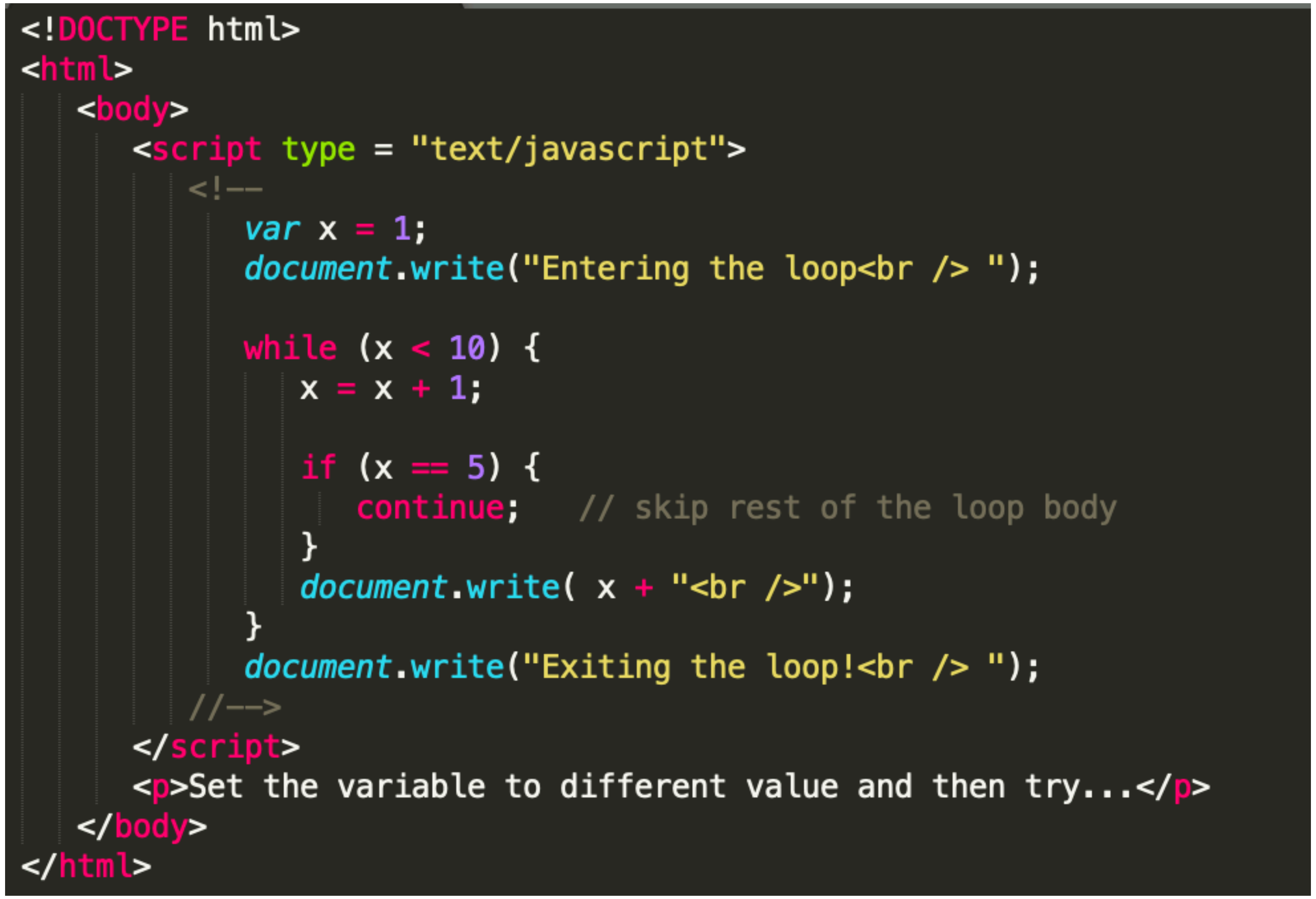
```
<!DOCTYPE html>
<html>
    <body>
        <script type = "text/javascript">
            <!--
            var x = 1;
            document.write("Entering the loop<br /> ");

            while (x < 20) {
                if (x == 5) {
                    break;   // breaks out of loop completely
                }
                x = x + 1;
                document.write( x + "<br />");
            }
            document.write("Exiting the loop!<br /> ");
            //-->
        </script>

        <p>Set the variable to different value and then try...</p>
    </body>
</html>
```

G2 | Academy

JavaScript - Continue

- The **continue** statement tells the interpreter to immediately **start the next iteration** of the loop and **skip the remaining** code block

JavaScript - Continue



```
<!DOCTYPE html>
<html>
    <body>
        <script type = "text/javascript">
            <!--
                var x = 1;
                document.write("Entering the loop<br /> ");

                while (x < 10) {
                    x = x + 1;

                    if (x == 5) {
                        continue;   // skip rest of the loop body
                    }
                    document.write( x + "<br />");
                }
                document.write("Exiting the loop!<br /> ");
            //-->
        </script>
        <p>Set the variable to different value and then try...</p>
    </body>
</html>
```

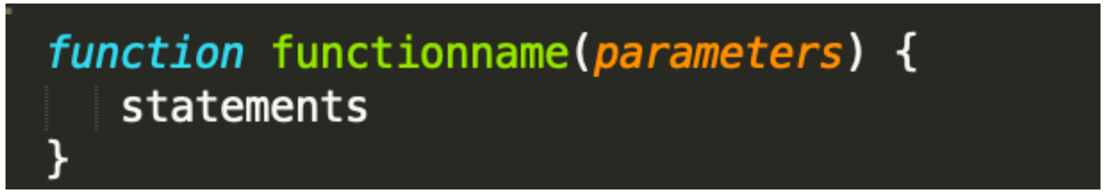
G2 | Academy

Function

G2 | Academy

JavaScript - Function

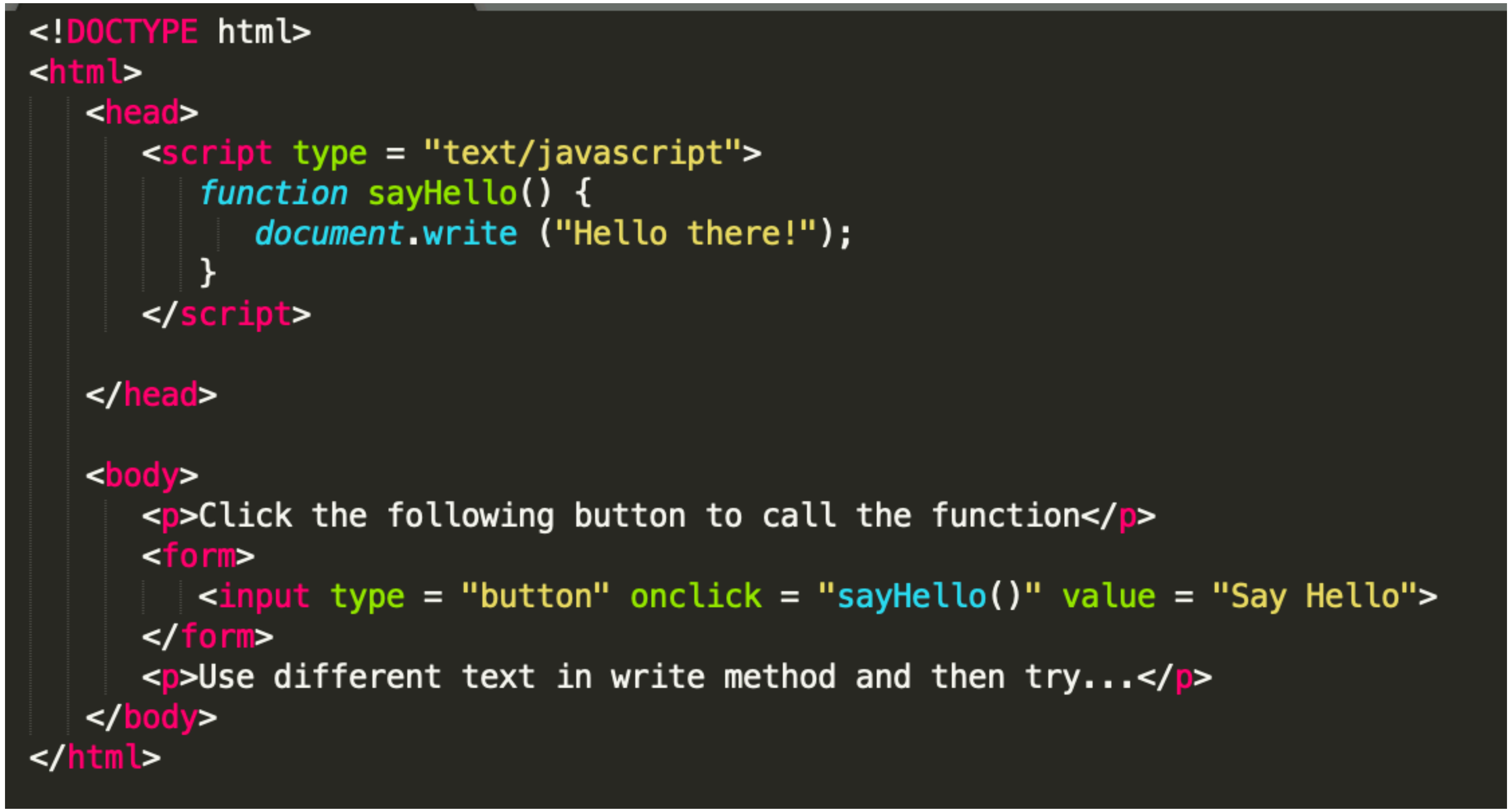
- JavaScript functions are **defined** with the function keyword



```
function functionname(parameters) {
    statements
}
```

G2 | Academy

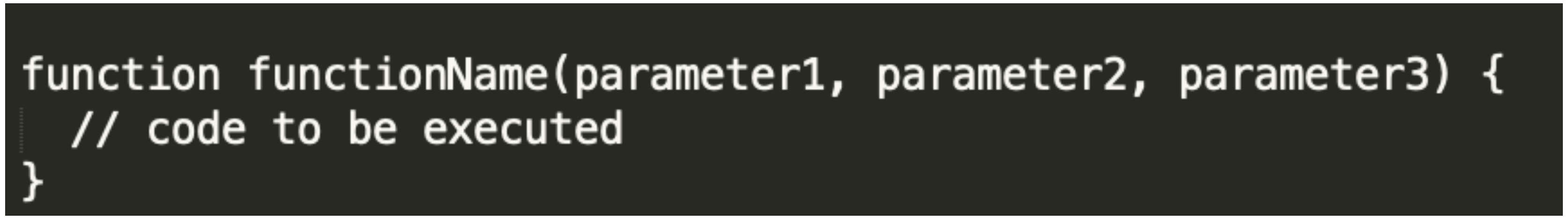
JavaScript - Function



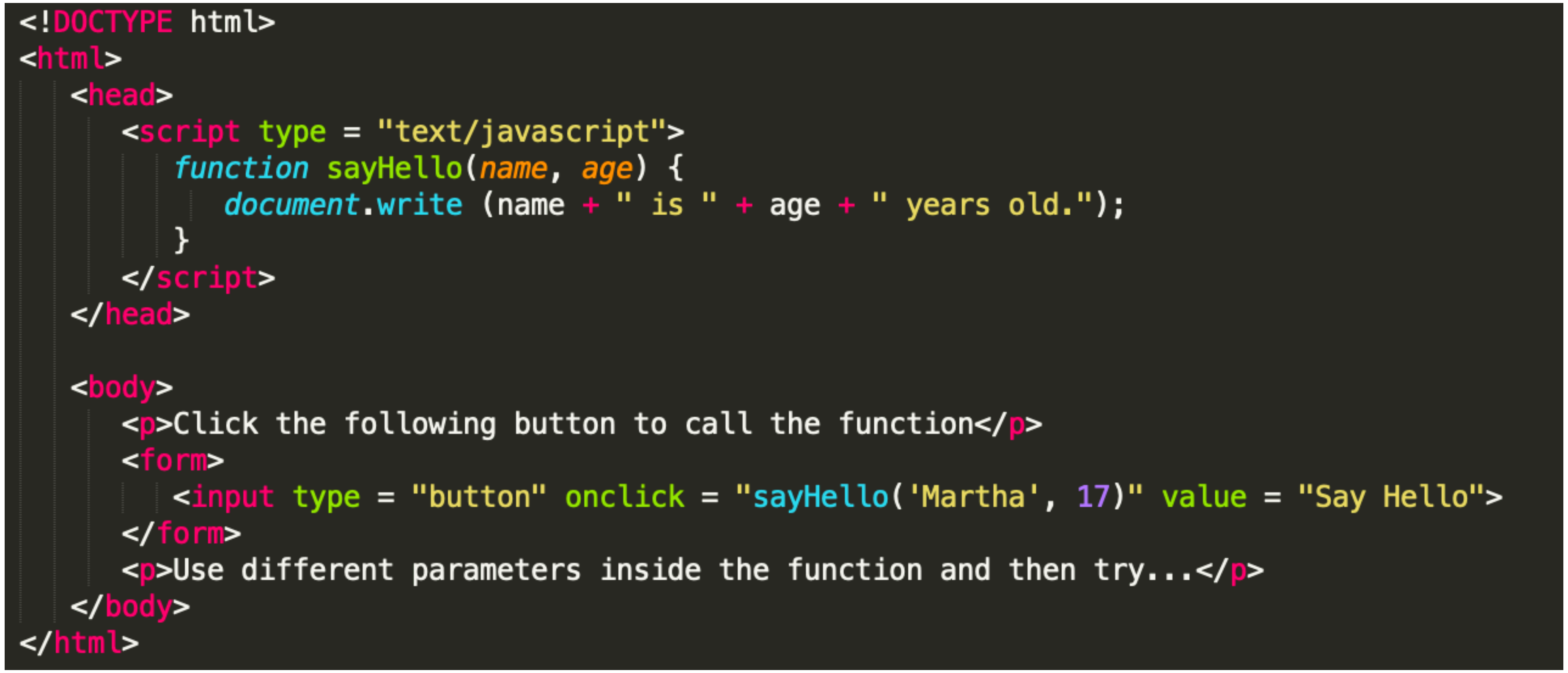
```
<!DOCTYPE html>
<html>
    <head>
        <script type = "text/javascript">
            function sayHello() {
                document.write ("Hello there!");
            }
        </script>

    </head>

    <body>
        <p>Click the following button to call the function</p>
        <form>
            <input type = "button" onclick = "sayHello()" value = "Say Hello">
        </form>
        <p>Use different text in write method and then try...</p>
    </body>
</html>
```

G2 | Academy

JavaScript - Function Parameter


```
function functionName(parameter1, parameter2, parameter3) {
  // code to be executed
}
```

JavaScript - Function Parameter



```html
<!DOCTYPE html>
<html>
    <head>
        <script type = "text/javascript">
            function sayHello(name, age) {
                document.write (name + " is " + age + " years old.");
            }
        </script>
    </head>

    <body>
        <p>Click the following button to call the function</p>
        <form>
            <input type = "button" onclick = "sayHello('Martha', 17)" value = "Say Hello">
        </form>
        <p>Use different parameters inside the function and then try...</p>
    </body>
</html>
```

G2 | Academy

Object

JavaScript - Object

- A car is an **object**
- A car has **properties** like weight and color, and **methods** like start and stop

Object	Properties	Methods

	car.name = Fiat
	car.model = 500
	car.weight = 850kg
	car.color = white
		car.start()
		car.drive()
		car.brake()
		car.stop()

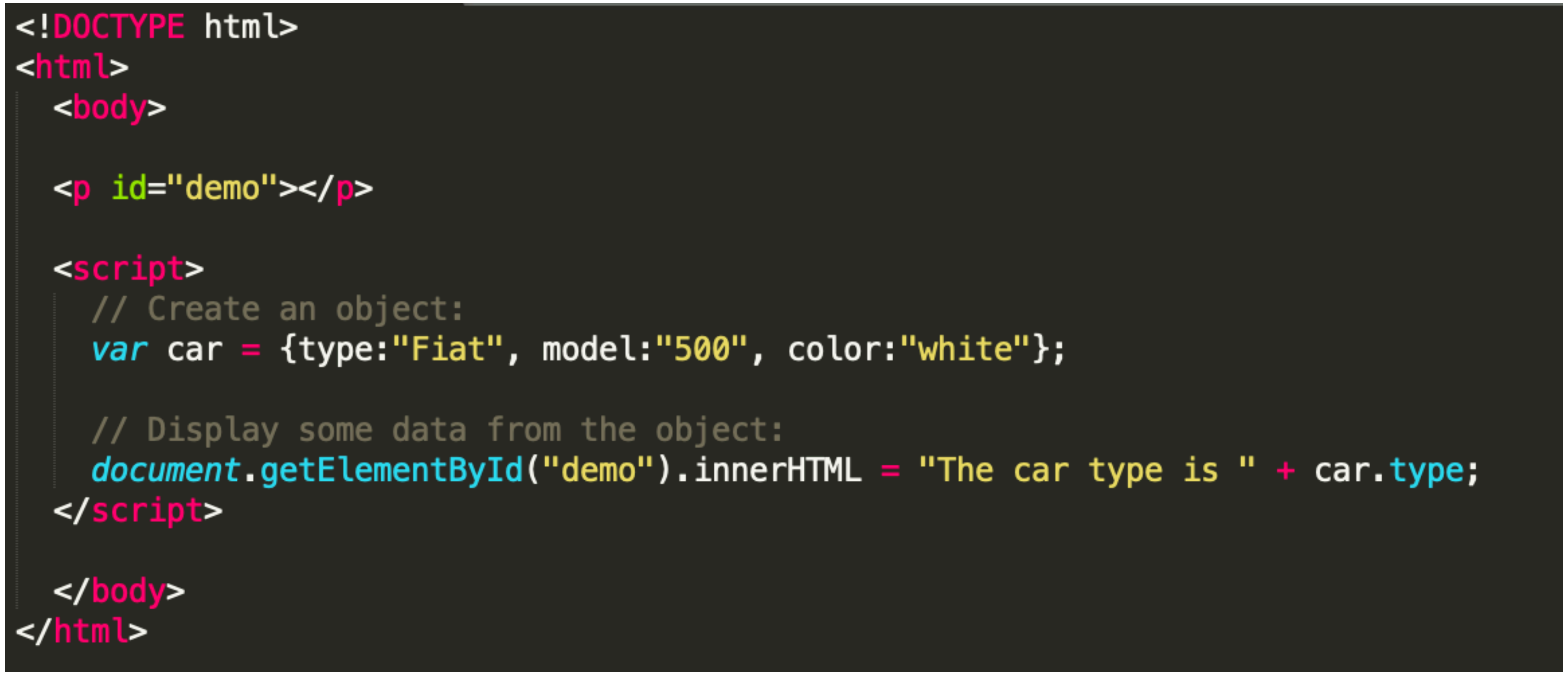
G2 | Academy

JavaScript - Object

- Simple Value
  `var car = "Fiat";`

- Many Values
  `var car = {type:"Fiat", model:"500", color:"white"};`

JavaScript - Object



```
<!DOCTYPE html>
<html>
  <body>

  <p id="demo"></p>

  <script>
    // Create an object:
    var car = {type:"Fiat", model:"500", color:"white"};

    // Display some data from the object:
    document.getElementById("demo").innerHTML = "The car type is " + car.type;
  </script>

  </body>
</html>
```

G2 | Academy

JavaScript – Object Properties

- The **name:values** pairs in JavaScript objects are called **properties**
  For example:
  - One line:

```
var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};
```

  - Multiple line:

```
var person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};
```

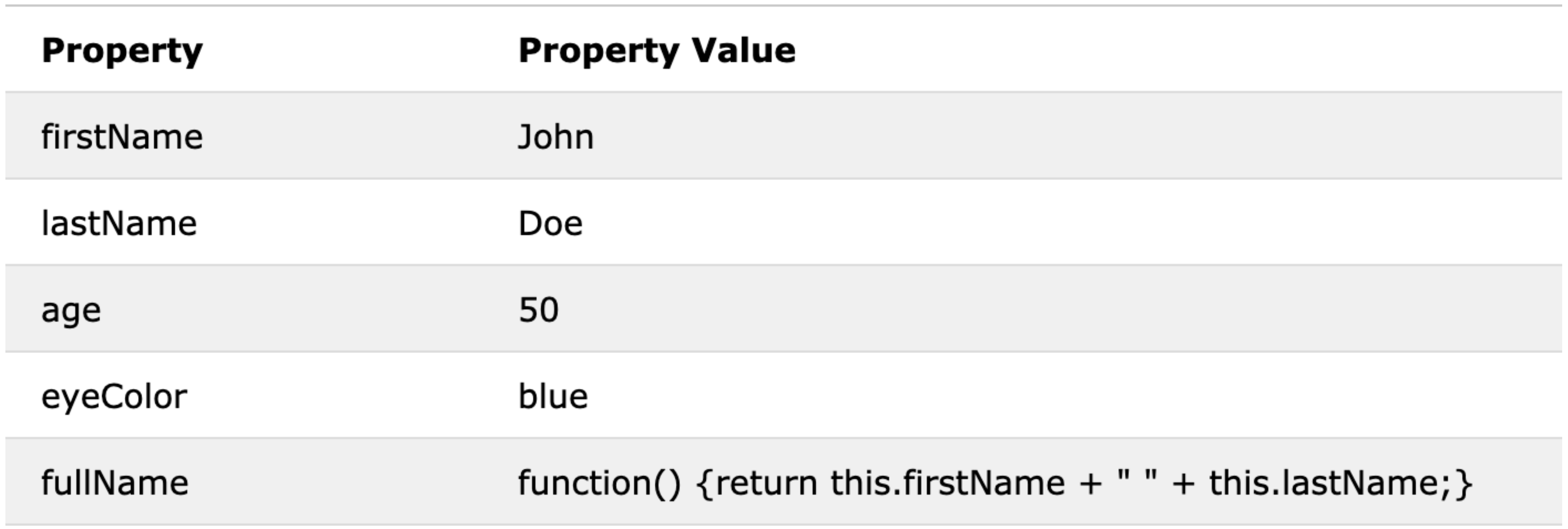
G2 | Academy

JavaScript - Object Properties

- Accessing Object Properties:
  objectName.propertyName
  or
  objectName["propertyName"]

JavaScript – Object Method

- Methods are **actions** that can be performed on objects
- Methods are stored in properties as **function definitions**



Property	**Property Value**
firstName	John
lastName	Doe
age	50
eyeColor	blue
fullName	function() {return this.firstName + " " + this.lastName;}

JavaScript - Object Method

```
var person = {
  firstName: "John",
  lastName : "Doe",
  id       : 5566,
  fullName : function() {
    return this.firstName + " " + this.lastName;
  }
};
```

G2 | Academy

JavaScript - Object Method

- Accessing Object Methods:

```
objectName.methodName()
```

```
name = person.fullName();
```

```
name = person.fullName;
```

JavaScript - Object Method

- When a JavaScript variable is declared with the keyword **"new"**, the variable is created as an **object**

```
var x = new String();        // Declares x as a String object
var y = new Number();        // Declares y as a Number object
var z = new Boolean();       // Declares z as a Boolean object
```

- **Avoid String, Number, and Boolean objects**

G2 | Academy

Array

JavaScript – Array Assign a Value

- The **Array** object lets you store **multiple values** in a **single variable**

```
var array_name = [item1, item2, ...];
```

- Single line

```
var cars = ["Saab", "Volvo", "BMW"];
```

- Multiple line

```
var cars = [
  "Saab",
  "Volvo",
  "BMW"
];
```

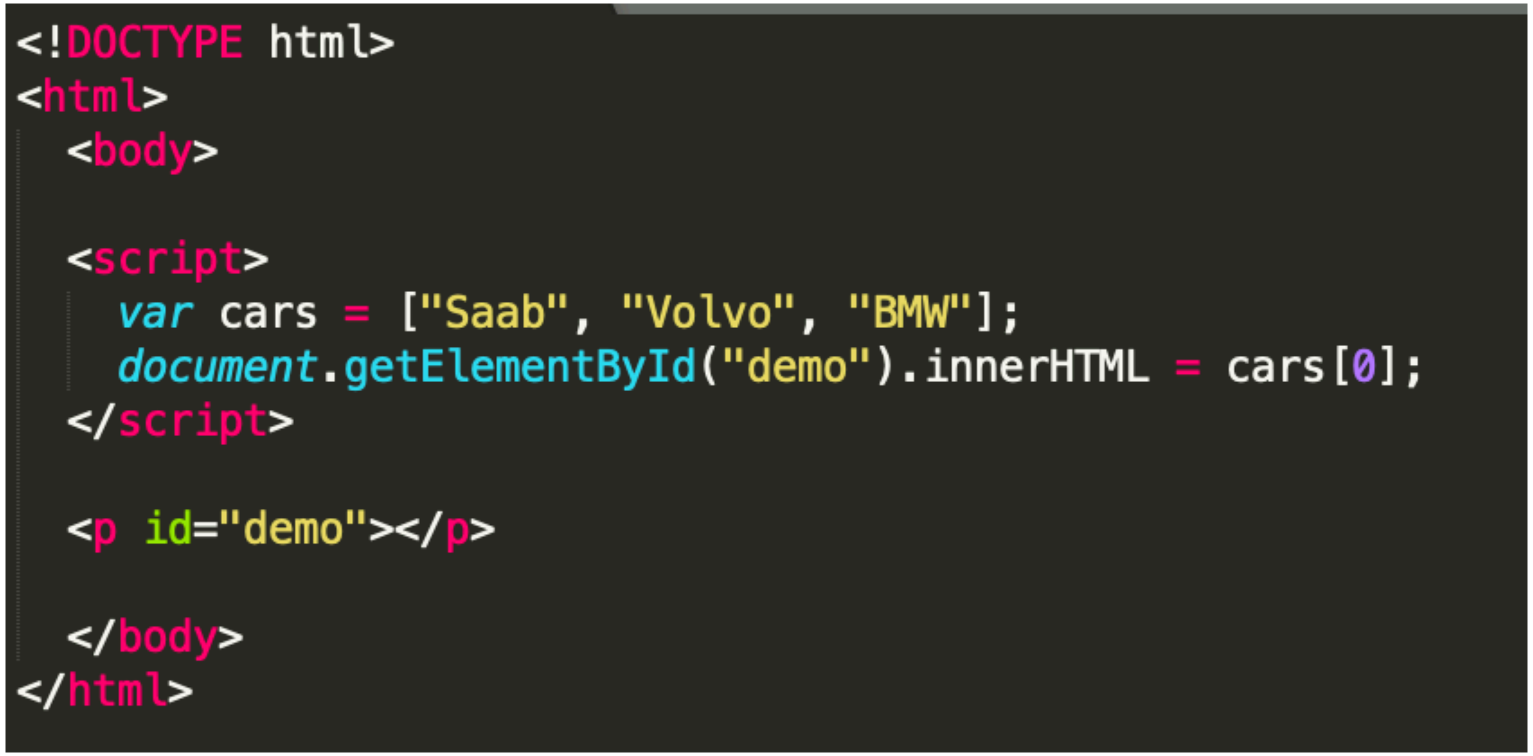
JavaScript - Array Object

```
var cars = new Array("Saab", "Volvo", "BMW");
```

- Access and set a value:

```
var cars = ["Saab", "Volvo", "BMW"];
document.getElementById("demo").innerHTML = cars[0];
```

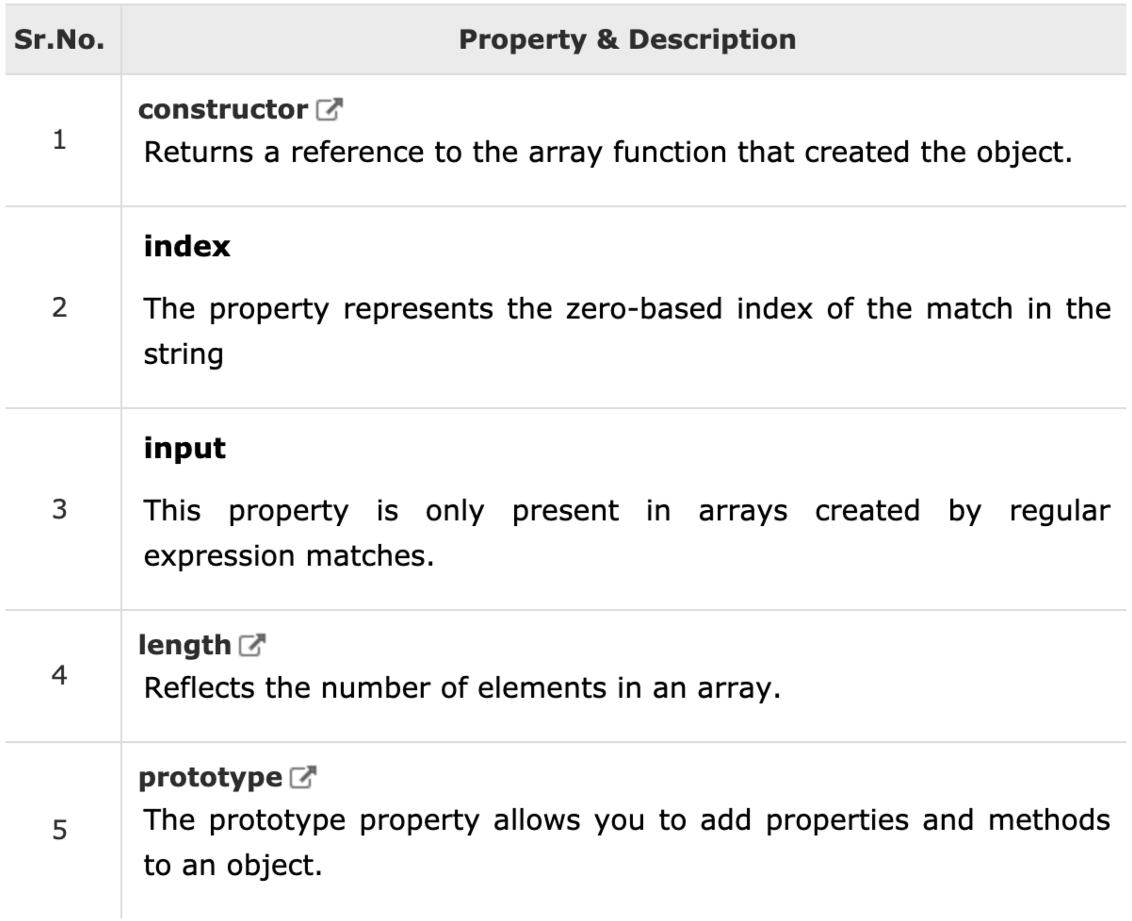
JavaScript - Array Object



```
<!DOCTYPE html>
<html>
  <body>

  <script>
    var cars = ["Saab", "Volvo", "BMW"];
    document.getElementById("demo").innerHTML = cars[0];
  </script>

  <p id="demo"></p>

  </body>
</html>
```

G2 | Academy

JavaScript - Array Properties



Sr.No.	Property & Description
1	**constructor** Returns a reference to the array function that created the object.
2	**index** The property represents the zero-based index of the match in the string
3	**input** This property is only present in arrays created by regular expression matches.
4	**length** Reflects the number of elements in an array.
5	**prototype** The prototype property allows you to add properties and methods to an object.

G2 | Academy

JavaScript - Avoid to do in Array

- There is **no need to use** the JavaScript's built-in array constructor **new Array()**

```
var points = new Array();       // Bad
var points = [];                // Good
```

```
var points = new Array(40, 100, 1, 5, 25, 10); // Bad
var points = [40, 100, 1, 5, 25, 10];          // Good
```

Event

G2 | Academy

JavaScript - Event

- HTML events are **"things"** that happen to HTML elements
- JavaScript is used in HTML pages, JavaScript can **"react"** on these events
- Single Quotes:

```
<element event='some JavaScript'>
```

- Double Quotes:

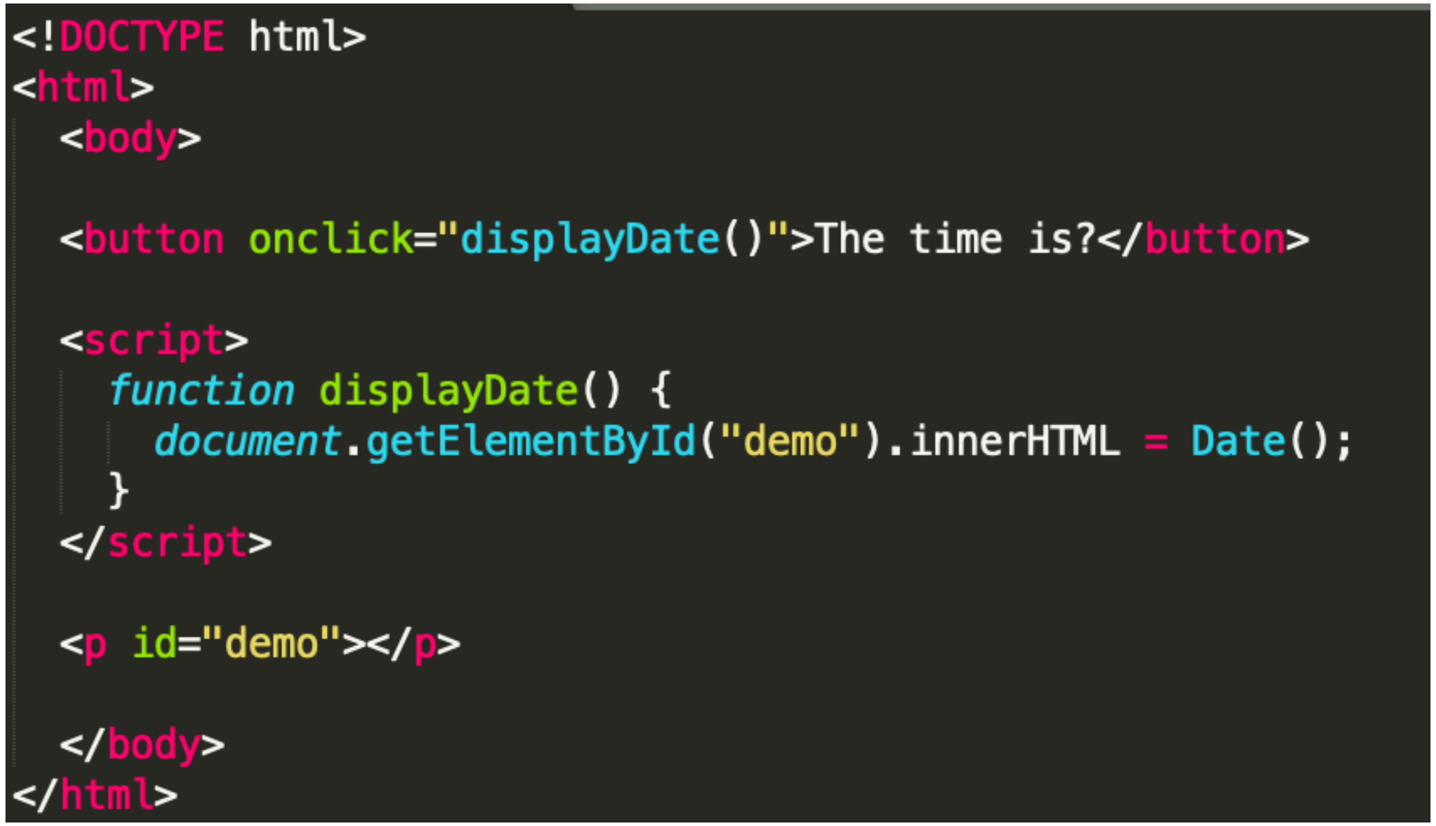
```
<element event="some JavaScript">
```

G2 | Academy

JavaScript - Event

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

JavaScript - Event



```
<!DOCTYPE html>
<html>
  <body>

  <button onclick="displayDate()">The time is?</button>

  <script>
    function displayDate() {
      document.getElementById("demo").innerHTML = Date();
    }
  </script>

  <p id="demo"></p>

  </body>
</html>
```

G2 | Academy

Summary

- While Loop
- For Loop
- Break & Continue
- Function
- Event
- Object
- Array

G2 | Academy