



INTEGRATIVE PROGRAMMING AND TECHNOLOGIES I

P F C - 5 1 0 7

Topic 8 – JavaScript

What is JavaScript?

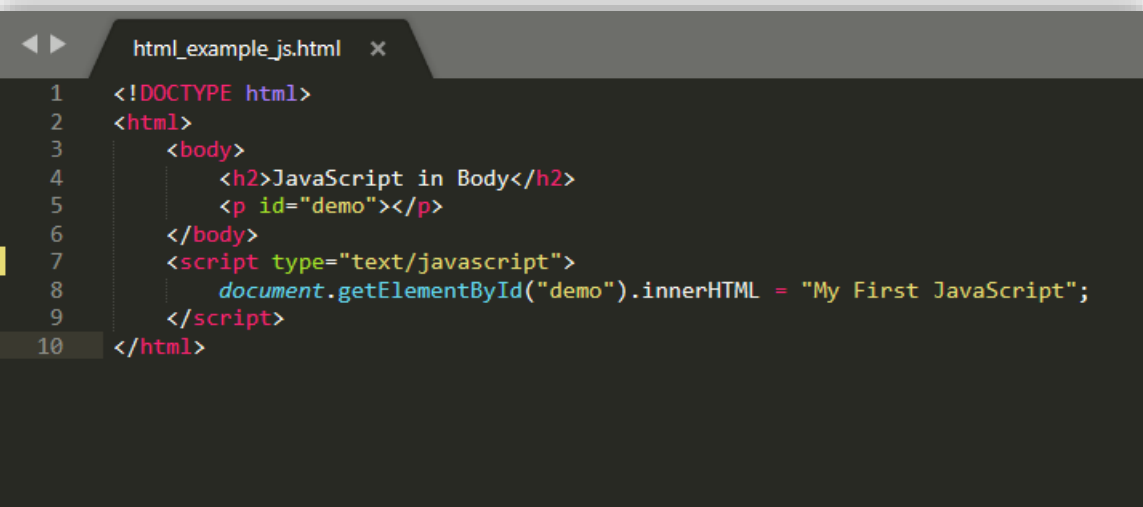
- Commonly used for creating web pages that allow developers to add dynamic behavior to the webpage and add special effects to it
- Also used for validation purposes
- Allows the user to load the content in a document without reloading the webpage
- Uses `.js` file extension when used externally otherwise, it is written within the `<script>` tag in an HTML file

Note: JavaScript and Java are two different languages

The *document.getElementById()* Method

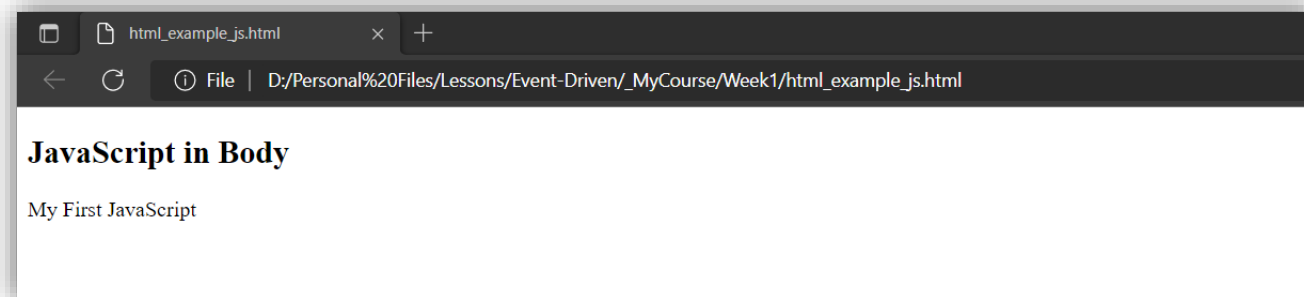
This method is used to manipulate an HTML element.

- Returns an element if the ID exists otherwise, it returns a null value
- If two or more elements with the same ID exists, it returns the first element with the same ID



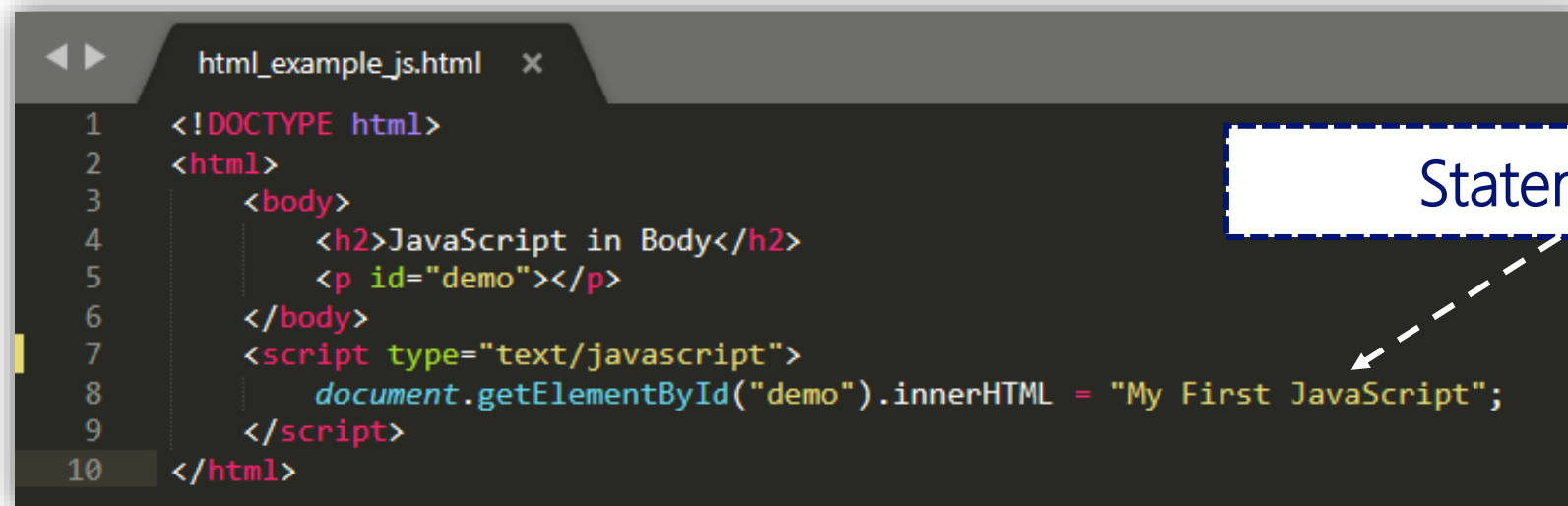
```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <h2>JavaScript in Body</h2>
5     <p id="demo"></p>
6   </body>
7   <script type="text/javascript">
8     document.getElementById("demo").innerHTML = "My First JavaScript";
9   </script>
10 </html>
```

.innerHTML is used for *elements*
.value is used for *form elements*



Statements

JavaScript Statements are just lines of code. It is composed of values, operators, expressions, keywords, and comments. Statements are set of instructions executed by a machine one at a time in the same order they were written.



The screenshot shows a code editor window titled 'html_example.js.html'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <h2>JavaScript in Body</h2>
5     <p id="demo"></p>
6   </body>
7   <script type="text/javascript">
8     document.getElementById("demo").innerHTML = "My First JavaScript";
9   </script>
10 </html>
```

A dashed arrow points from a box labeled 'Statement' to the JavaScript code on lines 7-9.

Statement

To comment a statement in JavaScript, use the `//` symbol.

Variables

Three types of JavaScript variables:

- var - globally or locally scoped variables. This type of variable can be declared and used outside of a function. It can also be re-declared and updated

```
[ var text = "This is a JavaScript variable." ]
```

- let – block-scoped variables. It cannot be re-declared but it can be updated

```
[ let text = "This is a JavaScript variable." ]
```

- const – block-scoped variables. It cannot be re-declared nor updated as const variables are supposed to be constant

```
[ const text = "This is a JavaScript variable." ]
```

Note: JavaScript variables can be declared using variable name only

Operators

JavaScript uses assignment and arithmetic operators. The assignment operator `=` is used to assign a specific value to a variable while arithmetic operators `+` `-` `*` `/` are used to compute values.

```

10  <script type="text/javascript">
11      var i = 2;
12      var j = 5;
13      document.getElementById("sum").innerHTML = i + j;
14      document.getElementById("difference").innerHTML = j - i;
15      document.getElementById("product").innerHTML = i * j;
16      document.getElementById("quotient").innerHTML = j / i;
17  </script>

```

Assignment Operator

Arithmetic Operators

Conditional Statements and Comparison Operators

JavaScript uses conditional statements for executing a code in various conditions. Comparison Operators `==` `===` `!=` `!==` `>` `<` `>=` `<=` makes use of Conditional Statements to compare the equality or difference between two variables or values.

```
23     if(i < j)
24     {
25         document.getElementById("lessThan").innerHTML = i + " is less than " + j;
26     }
27     if(j > i)
28     {
29         document.getElementById("greaterThan").innerHTML = j + " is greater than " + i;
30     }
31     if(j >= k)
32     {
33         document.getElementById("greaterThanOrEqualTo").innerHTML = j + " is greater OR EQUAL TO " + k;
34     }
```

The operator `==` checks if both values matched while the `===` checks if the values as well as the data type matched.

Logical Operators

JavaScript uses logical operators `&&` `||` `!` to determine logic between values. `&&` stands for **and**, `||` stands for **or** and `!` stands for **not**.

e.g. For example, given that we have variables `x = 5`, `y = 4` and `z = 5`, instead of writing it like this:

```
if (x != y) {  
    if(x == z) {  
        document.getElementById("id").innerHTML = "X is equal to Z and  
not equal to Y";  
    }  
}
```

You can write it like this instead;

```
if (x != y && x == z)  
    document.getElementById("id").innerHTML = "X is equal to Z and not  
equal to Y";
```


Functions

A function is a block of code designed to perform a specific task when called. The syntax for JS function is *functionName(parameters)*.

e.g. We have a function that adds two separate numbers and returns the sum

```
function sum(firstNumber, secondNumber)  
{  
    return firstNumber + secondNumber;  
}
```

sum is the function name and firstNumber and secondNumber are the parameters

```
12      function sum(first_number, second_number)  
13      {  
14          return first_number + second_number;  
15      }  
16  
17      var x = 5;  
18      var y = 10;  
19  
20      var z = sum(x, y);
```

The *onclick()* HTML Event

The *onclick()* is an HTML event that calls a function when an element is clicked.

```
<div onclick="divClick()">
  This is a div
</div>
<br>
<button onclick="buttonClick()">
  This is a button
</button>
<script type="text/javascript">
  function divClick(){
    alert('You clicked a DIV')
  }

  function buttonClick(){
    alert('You clicked a BUTTON')
  }
</script>
```

END OF TOPIC 8