Why Study JavaScript?

JavaScript is one of the **3 languages** all web developers **must** learn:

1. **HTML** to define the content of web pages
2. **CSS** to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages

JavaScript Can Change HTML Content

One of many JavaScript HTML methods is **getElementById().**

Example:

```
<html>
      <body>
            <h2>What Can JavaScript Do?</h2>
                  <p id="demo">JavaScript can change HTML content.</p>
                  <button type="button" onclick='document.getElementById("demo").innerHTML
                  = "Hello JavaScript! '"> Click Me! </button>
      </body>
</html>
```

JavaScript Can Change HTML Styles (CSS)

Example:

```
<html>
      <body>
      <h2>What Can JavaScript Do?</h2>
      <p id="demo">JavaScript can change the style of an HTML element.</p>
      <button type="button" onclick="document.getElementById('demo').style.fontSize='35px'"> Click
Me! </button>
      </body>
</html>
```

JavaScript Can Hide HTML Elements

Example:

<html>

<body>

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can hide HTML elements.</p>

<button type="button" onclick="document.getElementById('demo').style.display='none'"> Click Me!

</button>

</body>

</html>

JavaScript Where To

The **<script>** Tag

In HTML, JavaScript code is inserted between <script> and </script> tags.

Example:

<html>

<body>

<h2>JavaScript in Body</h2>

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

<script>

        document.getElementById("demo").innerHTML = "My First JavaScript";

</script>

</body>

</html>

JavaScript Functions and Events

A JavaScript function is a block of JavaScript code, that can be executed when "called" for.

For example, a function can be called when an **event** occurs, like when the user clicks a button.

JavaScript in <head> or <body>

You can place any number of scripts in an HTML document.

Scripts can be placed in the <body>, or in the <head> section of an HTML page, or in both.

JavaScript in <head>

In this example, a JavaScript function is placed in the <head> section of an HTML page.

The function is invoked (called) when a button is clicked:

Example:

```html
<html>
<head>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>
<body>
<h2>Demo JavaScript in Head</h2>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>
</body>
</html>
```

JavaScript in <body>

In this example, a JavaScript function is placed in the <body> section of an HTML page.

The function is invoked (called) when a button is clicked:

Example:

```html
<html>
<body>
<h2>Demo JavaScript in Body</h2>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
```

```
</script>
</body>
</html>
```

External JavaScript

Scripts can also be placed in external files:

External file: myScript.js

```javascript
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
```

```html
<html>
<body>

	<h2>Demo External JavaScript</h2>

	<p id="demo">A Paragraph.</p>

	<button type="button" onclick="myFunction()">Try it</button>

	<p>This example links to "myScript.js".</p>

	<p>(myFunction is stored in "myScript.js")</p>

	<script src="myScript.js"></script>
</body>
</html>
```

JavaScript Output

JavaScript Display Possibilities

JavaScript can "display" data in different ways:

- Writing into an HTML element, using innerHTML.

- Writing into the HTML output using document.write().

- Writing into an alert box, using window.alert().

- Writing into the browser console, using console.log().

Using innerHTML

To access an HTML element, JavaScript can use the document.getElementById(id) method.

The id attribute defines the HTML element. The innerHTML property defines the HTML content:

Example:

```
<html>
<body>
<h1>My First Web Page</h1>
<p>My First Paragraph</p>
<p id="demo"></p>
<script>
        document.getElementById("demo").innerHTML = 5 + 6;
</script>
</body>
</html>
```

Using document.write()

For testing purposes, it is convenient to use document.write():

Example:

```
<html>
<body>
<h1>My First Web Page</h1>
<p>My first paragraph.</p>
<script>
        document.write(5 + 6);
</script>
</body>
</html>
```

Using window.alert()

You can use an alert box to display data:

Example

```html
<html>
<body>
<h1>My First Web Page</h1>
<p>My first paragraph.</p>
<script>
window.alert(5 + 6);
</script>
</body>
</html>
```

Using console.log()

For debugging purposes, you can call the console.log() method in the browser to display data.

Example

```html
<html>
<body>
<script>
        console.log(5 + 6);
</script>
</body>
</html>
```