Javascript

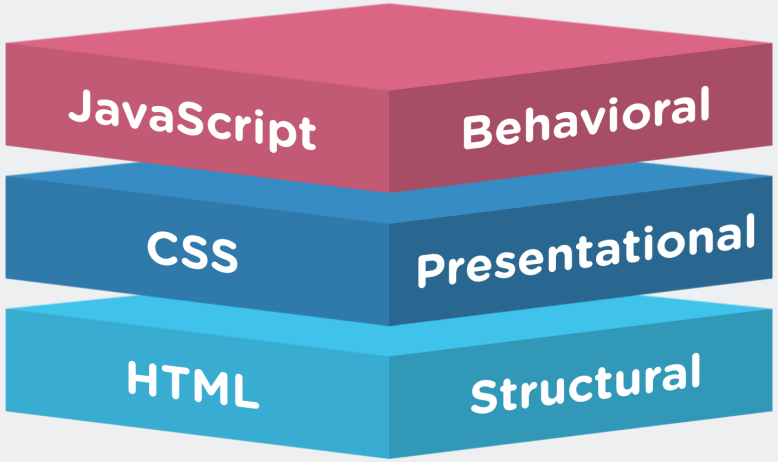
PART 1

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

(https://www.w3schools.com/js/js_intro.asp)

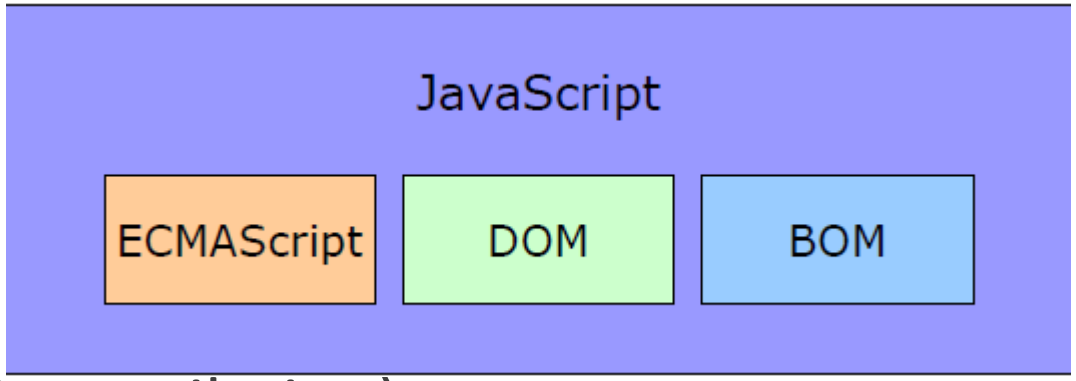


Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries.

All major web browsers have a dedicated JavaScript engine to execute the code on users' devices.

What is Javascript?

JavaScript is a:
High-level
dynamic
untyped
and interpreted
programming language (JIT compilation).
It has been standardized in the ECMAScript language specification



https://en.wikipedia.org/wiki/ECMAScript

Java vs Javascript

The two languages fill very different roles in web development and programming as a whole

Javascript key ideas

- **Load and go delivery**: No .exe, .dll or any abstract class files
- **Loose typing**: End of line semicolons are optional; variables do not have to be declared before using, and can have values of different types during execution…
- **Objects** are used as general containers. Classes are functions.
- **Prototypal inheritance** is implemented by cloning existing objects
- **Use of Lambda functions**: Functions that don't get named, and are passed to another function in order pass a behavior as a value.
- Linkage through **global variables**

Top programming languages (PYPL)

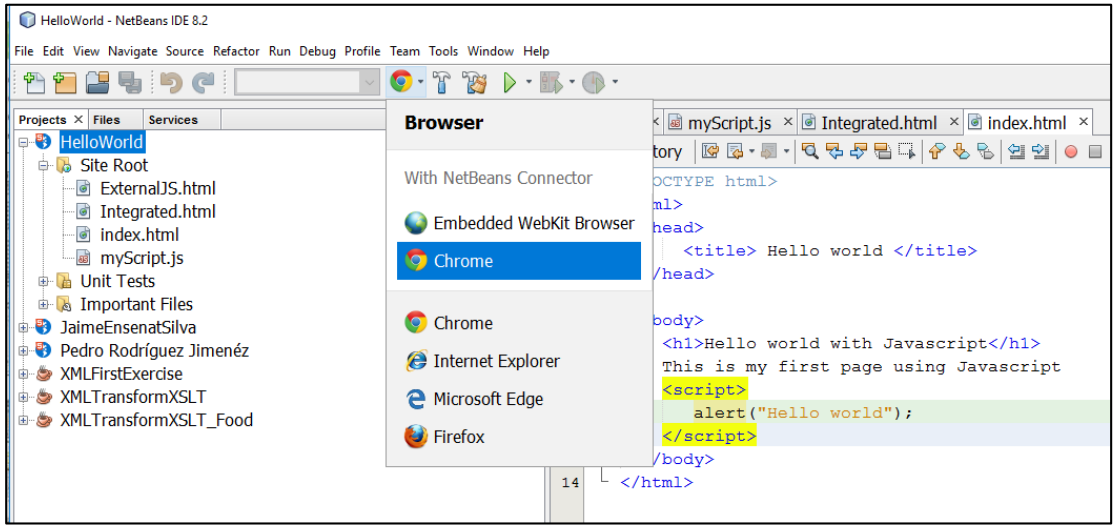
Worldwide, Mar 2022 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	28.27 %	-2.0 %
2		Java	18.03 %	+0.8 %
3		JavaScript	8.86 %	+0.4 %
4		C#	7.51 %	+0.6 %
5		C/C++	7.32 %	+0.6 %
6		PHP	5.71 %	-0.4 %
7		R	4.23 %	+0.5 %
8		Objective-C	2.28 %	-1.2 %

Getting started

What do you need:

- **Text editor**: Notepad++, Sublime, Netbeans (with Chrome connector), …
- **Web browser**: Google Chrome, Firefox, Microsoft Edge, Opera, …
- **Debugger**: Modern web browsers debuggers, IDE debuggers



The <script> tag

- You can place the script blocks anywhere on the page, but the results are going to be slightly different

```
<html>
    <head>
        <title>Javascript1</title>
        <script>
            document.getElementById('donetag').innerHTML = "DONE";
        </script>
    </head>
    <body>
        <h1> My first Javascript</h1>
        <div> This is my Hello world with Javascript </div>
        <div id="donetag">not yet...</div>
    </body>
</html>
```

```
<html>
    <head>
        <title>Javascript1</title>
    </head>
    <body>
        <h1> My first Javascript</h1>
        <div> This is my Hello world with Javascript </div>
        <div id="donetag">not yet...</div>
        <script>
            document.getElementById('donetag').innerHTML = "DONE";
        </script>
    </body>
</html>
```

The <script> tag

¿How to use Javascript in an HTML document?
There are three ways:

- In-Line Javascript
- External Javascript
- Integrated in HTML elements

The <script> tag

- In-Line Javascript:

```
<html>
    <head>
        <title>Javascript1</title>
    </head>
    <body>
        <h1> My first Javascript</h1>
        <div> This is my Hello world with Javascript </div>
        <div id="donetag">not yet...</div>
        <script>
            document.getElementById('donetag').innerHTML = "DONE";
        </script>
    </body>
</html>
```

The <script> tag

- External Javascript

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

myScript.js

Source History

```
1  document.getElementById('donetag').innerHTML = "DONE";
2  alert("Hello world");
```

- Placing scripts in external files has some advantages:
  - It separates HTML and code
  - It makes HTML and JavaScript easier to read and maintain
  - Cached JavaScript files can speed up page loads
- To add several script files to one page - use several script tags:

```
<script src="myScript1.js"></script>
<script src="myScript2.js"></script>
```

The <script> tag

- Integrated in HTML elements:

```
<html>
  <head>
      <title> Hello world </title>
  </head>

  <body>
    <h1>Hello world with Javascript</h1>
    <p onclick="alert('Hello world');">
    This is my first page using Javascript
    </p>
  </body>
</html>
```

Basic syntax rules

- JavaScript is Case Sensitive:
  - For example, The variables **lastName** and **lastname**, are two different variables
  - Recommendations:
    - Objects, variables, functions start with a lowercase letter
    - Use **Lower Camel Case:**

      JavaScript programmers tend to use camel case that starts with a lowercase letter:
      firstName, lastName, masterCard, interCity.

- In JavaScript, the first character must be a letter, an underscore (_), or a dollar sign ($).
  - Subsequent characters may be letters, digits, underscores, or dollar signs.
- JavaScript statements are separated by **semicolons**
- Code after double slashes ***//*** or between ***/**** and ****/*** is treated as a **comment**.

Coding examples

Poor legibility:

```
001 <script type="text/javascript">var i,j=0;
002 for (i=0;i<5;i++) { alert("Variable i: "+i);
003 for (j=0;j<5;j++) { if (i%2==0) {
004 document.write
005 (i + "-" + j + "<br>");}}}</script>
```

Easy legibility:

```
<script type="text/javascript">
        var i,j;
        for (i=0;i<5;i++){
                alert("Variable i: "+i);
                for (j=0;j<5;j++){
                        //if (i%2==0) {
                            document.write(i + "-" + j + "<br>");
                        //}
                }
        }
    </script>
```