

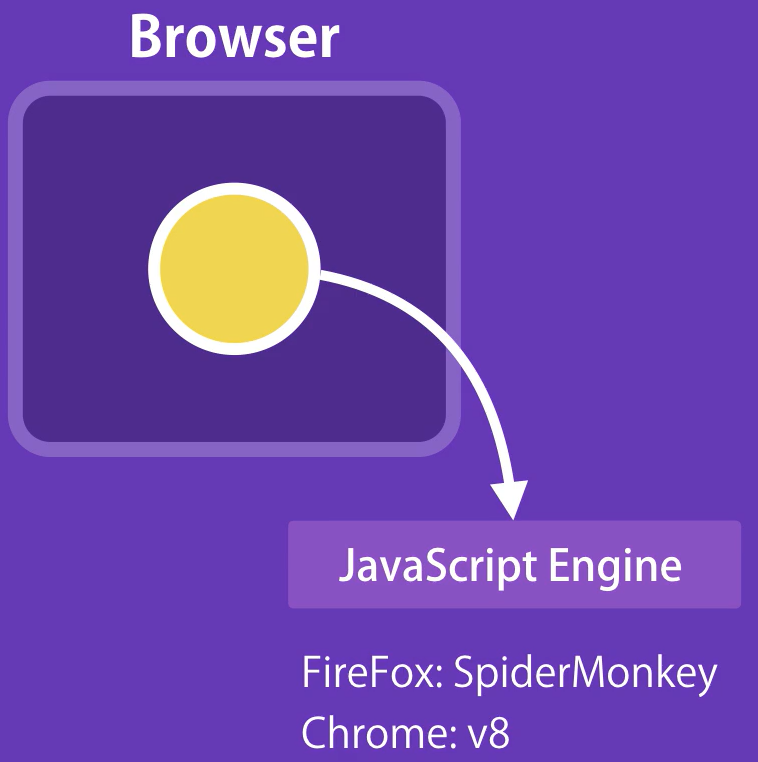
**What is JavaScript**:

Today we can use JavaScript to build



*Where does JavaScript code run*:

JavaScript was originally designed to run only in browsers. So every browser has a *JavaScript engine* that can execute JavaScript code.



In 2009 an engineer took the open source JavaScript engine in chrome and embedded it inside a C++ program called *Node*.

So node is a C++ program that includes Google’s V8 JavaScript engine.

“*With this implementation we can run our JavaScript code outside of a browser and pass to node for execution. With this we can develop backend of our application*”.

Summary: JavaScript can be run inside a browser or in Node.

*Difference between ECMAScript and JavaScript*:



ECMAScript is just a specification while JavaScript is a programming language that confirms to this specification.

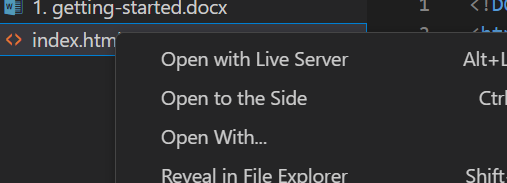
We have this organization called ECMA which is responsible for defining standards.

**Setting up development environment**:

For creating JavaScript based applications, you need

1. Code editor 🡪VS code
2. Node
3. Index.html file as host for JS code.
4. Live server extension

🡪 Right click on index.html file and select open with live server



It will open the browser with this address

<http://127.0.0.1:5500/index.html>

🡪 Add some simple HTML like this,

  <body>

    <h1>Hello World</h1>

  </body>

You will see changes in browser as soon as you save the file.

**JavaScript in Browsers**:

To start writing JavaScript code, first we need a script element. There are two places we can add a script element.

It can be either inside <head> </head> section or in the <body> </body> section.

Best practice is to add script element at the end of body section after all the existing elements.

Note: There are exceptions where we need to add script inside head section when we are using some third party code.

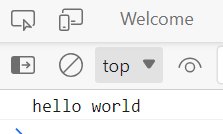
Let us write a simple console.log statement inside script.

    <script>

      console.log("hello world");

    </script>

We see the same reflected in browser console.



**Separation of Concerns**:

We need to separate our HTML which defines the content of our application with JavaScript which defines the behavior.

🡪 So create a new file called index.js

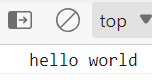
🡪 Put JavaScript code in this file



🡪 Reference this index.js file in to our index.html file by defining its name in src attribute of script tag.

<script src="index.js"></script>

And that is it, We still get the same result

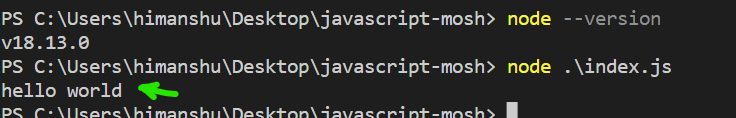


**JavaScript in Node:**

In last lecture, we executed this piece of code inside a browser and we will do the same in Node.

Inside the terminal of the current folder write

node .\index.js



We see the same output as we observed in browser console.