Language Map for JavaScript

<table>
<tr>
<td>Variable Declaration
Is this language strongly typed or dynamically typed? Provide at least three examples (with different data types or keywords) of how variables are declared in this language.</td>
<td>JavaScript is dynamically typed, you do not have to declare the variable type when declaring the value of the variable, JavaScript infers that for you.

var item = “toothbrush”; var is a globally accessible variable in JavaScript
var itemTotal = “30’;

let x = “4’; let is a locally accessible variable in JavaScript, that can be reassigned
let grade = “A”;
let completed = true;

const name = “Chris”; is a locally accessible variable that cannot be reassigned
const password = “01234”;</td>
</tr>
<tr>
<td>Data Types
List all of the data types (and ranges) supported by this language.</td>
<td>
<table>
<tr><th>Data Types</th><th>Description</th><th>Example</th></tr>
<tr><td><code>String</code></td><td>represents textual data</td><td><code>'hello'</code>
,
<code>"hello world!"</code>
etc</td></tr>
<tr><td><code>Number</code></td><td>an integer or a floating-point number</td><td><code>3</code>
,
<code>3.234</code>
,
<code>3e-2</code>
etc.</td></tr>
<tr><td><code>BigInt</code></td><td>an integer with arbitrary precision</td><td><code>900719925124740999n</code>
,
<code>1n</code>
etc.</td></tr>
<tr><td><code>Boolean</code></td><td>Any of two values: true or false</td><td><code>true</code>
and
<code>false</code></td></tr>
<tr><td><code>undefined</code></td><td>a data type whose variable is not initialized</td><td><code>let a;</code></td></tr>
</table>
</td>
</tr>
</table>

	`null`	denotes a `null` value	`let a = null;`
	`Symbol`	data type whose instances are unique and immutable	`let value = Symbol('hello');`
	`Object`	key-value pairs of collection of data	`let student = { };`

Selection Structures
Provide examples of all selection structures supported by this language (if, if else, etc.) ***Don't just list them, show code samples of how each would look in a real program.***

JavaScript supports the if, if…else, and switch selection structures.

If example:

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Using if Statement</title>
</head>
<body>
    <h1>
        Using the if Statement in the script
    </h1>
     <script type="text/javascript">
                                                    var Number = 45;
         if ((Number % 2) != 0) {
             document.write(Number + " is an odd number");
         }
         document.write("<BR/>Thank you!");
     </script>
 </body>
 </html>
```

If…else example:

```
  <!DOCTYPE HTML>
<html>
<head>
```

```
    <title>if...else Statement</title>
</head>
<body>
    <h1>
        Using the if...else Statement in the Script</h1>
    <script type="text/javascript">
                                                    var Number = 44;
        if ((Number % 2) != 0) {
            document.write(Number + " is an odd number");
        }
        else {
            document.write(Number + " is an even  number");
        }
        document.write("<BR/>Thank you!");
     </script>
</body>
</html>
```

Switch example:

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Using switch Statement</title>
</head>
<body>
    <h1>
        Using switch Statement in the script</h1>
    <script type="text/javascript">
                                                    var letter = "I";
        switch (letter) {
            default: document.write("consonant");
                break;
```

```
                case "A": document.write("A is a vowel");
                    break;
                case "E": document.write("E is a vowel");
                    break;
                case "I": document.write("I is a vowel");
                    break;
                case "O": document.write("O is a vowel");
                    break;
                case "U": document.write("U is a vowel");
                    break;
            }
            document.write("<BR/>Thank You!");
        </script>
    </body>
    </html>
```

Repetition Structures
Provide examples of all repetition structures supported by this language (loops, etc.) ***Don't just list them, show code samples of how each would look in a real program.***

JavaScript uses while and for loops for their repetition structures

While loop example:

```
var number = 1;
var sum = 0;


while (number < 11) {
    sum += number;
    ++number;
    console.log(number);
}
console.log(sum);
```

<table>
<tr><td></td><td>

For loop example:

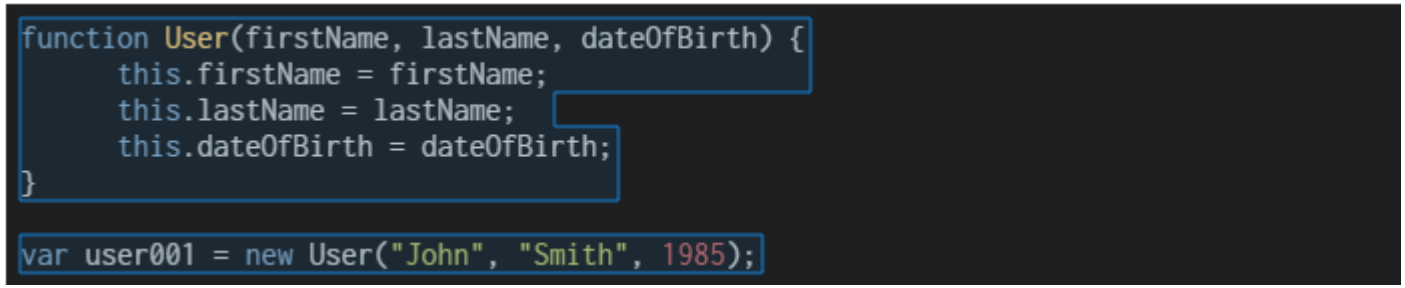
```
var number = 1;
var sum = 0;


for (var number = 1; number < 11; number++) {
    sum += number;
    console.log(sum);
}
console.log(sum);
```

</td></tr>
<tr><td>Arrays
If this language supports arrays, provide at least two examples of creating an array with a primitive or String data types (e.g. float, int, String, etc.)</td><td>Const littleDebbies = [“nutterbutter”, ”Oatmeal Creampie”, “Star Crunch”, “Swiss Cake Roll”, “Zebra Cake”];

Const gradeAvg = [95.6, 86.7, 77.3, 91.4, 79.1, 68.9, 84.5, 72.3];</td></tr>
<tr><td>Data Structures
If this language provides a standard set of data structures, provide a list of the data structures and their Big-Oh complexity.</td><td>
<table>
<tr><th>Data structure</th><th>Access</th><th>Search</th><th>Insertion</th><th>Deletion</th></tr>
<tr><td>Array</td><td>O(1)</td><td>O(N)</td><td>O(N)</td><td>O(N)</td></tr>
<tr><td>Stack</td><td>O(N)</td><td>O(N)</td><td>O(1)</td><td>O(1)</td></tr>
<tr><td>Queue</td><td>O(N)</td><td>O(N)</td><td>O(1)</td><td>O(1)</td></tr>
</table>
</td></tr>
</table>

	Singly Linked list O(N) O(N) O(1) O(1) **Doubly Linked List** O(N) O(N) O(1) O(1) **Objects (Hash Table)** O(1) O(1) O(1) O(1) **Binary Search Tree** O(log N) O(log N) O(log N) O(log N) **Heaps (Max and Min)** O(log N) O(log N) O(log N) O(log N) **Graphs**

Objects
If this language support object-orientation, provide an example of how you would write a simple object with a default constructor and then how you would instantiate it.
	JavaScript does provide object-oriented programming though not as straight forward and easy to complete some aspects of object oriented programming such as encapsulation and inheritance. But you can use a constructor to create an object which can be instantiated. Example below:
	 
	`function User(firstName, lastName, dateOfBirth) {`
	`  this.firstName = firstName;`
	`  this.lastName = lastName;`
	`  this.dateOfBirth = dateOfBirth;`
	`}`
	
	`var user001 = new User("John", "Smith", 1985);`

Runtime Environment *What runtime environment does this language compile to? For example, Java compiles to the Java Virtual Machine.*	JavaScript uses Node.js as it's runtime environment to compile to. Due to its ability to build mobile, web and IoT applications Node.js is becoming increasingly popular as a runtime environment for programming languages. Other languages that use Node.js include Python, Java, PHP, .Net, C++, Go, C, Ruby and Swift

Do other languages also compile to this runtime?	
Libraries/Frameworks *What are the popular libraries or frameworks used by programmers for this language? List at least three (3) and describe what they are used for..*	The most popular framework for JavaScript is **React.js,** it was created by Facebook. It makes creating interactive user interfaces easier. It's handy for making single page applications, small business applications and applications that cross platforms. **AngularJS** is Google's JavaScript framework. Its primary focus is on single page applications. It dynamically loads from the web server instead of from the web browser, as a result the SPAs behave more like a mobile application. **Gatsby** also known as GatsbyJS is based on a React framework, it helps build static website applications, which are an interconnected collection of HTML pages that display the same content to all viewers in the same order.
Domains *What industries or domains use this programming language? Provide specific examples of companies that use this language and what they use it for**. E.g. Company X uses JavaScript for its line of business applications.***	The ecommerce industry relies heavily on JavaScript to create and maintain their websites where the commerce happens. For examples **Groupon** (an online marketplace that offers discounts to it's customers) uses JavaScript for it's website. By using JavaScript, they are able to split their services into several web application for more efficient use. They built and developed their mobile application through JavaScript. **eBay**, a customer-to-customer ecommerce website use JavaScript for all front end and back end development/programming purposes. They used to rely on Java for all user facing stacks but have migrated to JavaScript and it's Node.js runtime. And **LinkedIn** which is a commerce site for connecting employers and employees uses JavaScript for all its website and mobile applications.