JavaScript is a main technology used in the World Wide Web with HTML and CSS. Also, it used in client side as well as the server side. It is a single threaded programming language. It does not wait for IO operations such as network calls (HTTP requests), DB operations and file read. Because of that it is asynchronous. Also, it is dynamically typed, and it support OOP as well as functional programming. It runs on the V8 engine on Google Chrome and spider monkey engine on the Firefox. ECMAScript is a Standard for JavaScript. Also, some popular frameworks have been created using JavaScript. For examples, react JS, node JS, express JS and Vue JS. We can facilitate web development by using this JavaScript based frameworks. By the end of 2021, react JS is the most popular framework among the developers.

We can see some major differences between the Java and the JavaScript.

JAVA

* Compiled based
* Runs on the virtual machine
* Server-side language
* Multithreaded language
* Variables define before the runtime

JS

* Interpreter based
* Runs on the browser
* Not a server-side language
* single threaded language
* variables define at the runtime

**Fundamentals in JavaScript**

Variable declaration keywords

As variable declaration keywords, we can use var, let and const in JavaScript. Var Keyword has scope of global or function. The two keywords let and const have a block scope. they can't be accessible outside the declared block.

Classes and objects

In older versions of JavaScript, classes had to be created using the new keyword with a function. Newer versions of JavaScript have introduced the class keyword. This allows us to create a class very easily. We can create objects using new keyword. We can also create objects using object literals as well.

Prototypes

JavaScript functions have a reference to an object called prototype. This object behaves like a class. Prototype used to create objects and for the inheritance. Keywords such as class and extend, which we use to create and inherit classes in JavaScript, have been developed using prototypes.

‘this’ keyword in JavaScript

‘this’ keyword is used in JavaScript to refer a variable context. Inside the object ‘this’ refers to object itself. In global context this refers to window object. And inside a function ‘this’ refers to a global context.

Closures

closure is a function which returns a function. These two functions are called inner function and outer function. It used to encapsulate variables into functions.

Callbacks and promises

JavaScript does not wait for IO operations, because of that it is asynchronous. But sometimes we have to execute several asynchronized functions in the form of synchronize. In such cases we use callbacks and promises. A callback is a function passed as an argument to another function. In this case it assigns to do async tasks. Promise is an object that being returned from an async task. It has properties like then, catch, finally to deal with the acid task. Sometimes lot of nested callbacks can cause callback hell. Solution to that is promise.

Some ECMAScript – ES6 Features

* Default parameters.
* Arrow function expressions.
* Spread operator
* Template literals
* DE structuring operator

**Version controlling**

Managing changes to our source code in programming can simply be called version control. Developers benefit greatly from using version control systems. This makes easier for developers to collaborate their works on projects. As a result, developers now have the ability to work remotely. It allows us to easily control the access to the source code. It also allows us to resolve conflicts. And we can easily get backups with this.

The most popular version control system is GIT. It is distributed version control system, and it is free and open source. It's supports multiple protocols such as HTTP SSH. It is works on Linux kernel and written in C because of that it is fast.

Here we must make the changes to the source code first in a separate branch and then merge those changes to the master branch. This way we can minimize the errors in the source code as well as easily troubleshoot the errors. And we should always make sure to have the latest version of the file.

**NoSQL databases**

Mostly the no SQL databases mean non-relational databases. Lot of non-relational databases are free open source. These database instances can run on different areas(servers) because of that these are highly distributed. The main advantage of using this NoSQL databases is We can choose a correct database type that match with our system requirements. NoSQL databases are running efficiently on clusters. In the cap theorem these databases ensure the availability, consistency, and partition tolerance. These databases can store data in four ways. We can store data in these databases as key value pairs documents, column families, and graphs. For examples mongo DB, Cassandra, neo4j, Redis and Raik.

* Mongo DB

Mongo DB is NoSQL document database. In this database we store data in BSON data type. it has strong query capabilities with aggregations such as query pipelines. It runs on the spider monkey engine. According to the cap theorem Mongo DB ensures the consistency and partition tolerance. It has a built file storage called Grid file system. In this database almost all the requests have responses because of that mongo DB is highly available database. It has achieved this by using replica sets.