**Intro to Javascript**

[*https://www.studytonight.com/javascript*](https://www.studytonight.com/javascript/javascript-data-types)

**Uses of Javascript**

* Websites
* Web Applications
* Server-side Application using NodeJS
* Mobile Applications (Hybrid Apps, React Native)
* Desktop Application using Electron JS
* Games, Smartwatch applications, etc.,

**Javascript as a Programming Language that is:**

* ***High Level***, it provides abstractions that allow you to ignore the details of the machine where it is running. It manages memory automatically with a garbage collector so you can focus on codes instead of memory management.
* ***Dynamic***, a dynamic language executes the codes at runtime. This has pros and cons, and it gives us powerful features like dynamic typing, late binding, reflection, functional programming, object runtime alteration, closures and much more.
* ***Dynamically-typed***, a variable does not enforce a type. You can reassign 5 any type to a variable, for example, assigning an integer to a variable that holds a string.
* ***Loosely typed,*** as opposed to strong typing, loosely (or weakly) typed languages do not enforce the type of an object, allowing more flexibility but denying us type safety and type checking (something that TypeScript - which builds on top of JavaScript - provides).
* **Interpreted**, Javascript codes do not need a compilation stage or are not compiled before being run. In practice, browsers do compile JavaScript before executing it, for performance reasons, but this is transparent to you: there is no additional step involved.
* ***Multi-paradigm,*** Javascript does not enforce any particular programming paradigm unlike Java for example which uses the Object Oriented programming paradigm. You can write JavaScript using an object-oriented paradigm, using prototypes and the new (as of ES6) classes syntax. You can write JavaScript in a functional programming style, with its first-class functions, or even in an imperative style (C-like).

**More about Javascript**

**Plain Javascript** or **Vanilla Javascript** or **Javascript (JS)** mean the same. It means, there is no framework used, third-party plugins or library used. Just pure javascript codes.

A **JavaScript framework** is an application framework written in JavaScript. It differs from a **JavaScript library** in its control flow: A library offers functions to be called by its parent code, whereas a framework defines the entire application design.

|  |  |
| --- | --- |
| **Examples of Javascript Framework** | **Examples of Javascript Library** |
| * Vue JS * Angular JS * D3 JS | * JQuery * Ember JS * React JS |
| [*https://learntocodewith.me/posts/javascript-libraries-frameworks*](https://learntocodewith.me/posts/javascript-libraries-frameworks) | |

**Javascript Syntax**

* **Whitespace,** unlike Python where spaces and indentation to codes matters in Javascript, whitespace has no meaning. Developers prefer to use plugins on their code editor for code indentation, etc.
* **JavaScript is case sensitive**. A variable named ***something*** is different from ***Something.*** Same for Javascript variable and function names.

***Note***: I will be showing samples of Javascript syntaxes using Javascript codes.

**Document Object Model (DOM)**

* Is an application programming interface (API) for valid HTML and well-formed XML documents. It defines the logical structure of documents and the way a document is accessed and manipulated
* Represents that same document so it can be manipulated. The DOM is an object-oriented representation of the web page, which can be modified with a scripting language such as JavaScript. The W3C DOM and WHATWG DOM standards are implemented in most modern browsers

**How does JavaScript access Dom?**

Refer to the link below to know more on accessing HTML elements via DOM

[*https://www.digitalocean.com/community/tutorials/how-to-access-elements-in-the-dom*](https://www.digitalocean.com/community/tutorials/how-to-access-elements-in-the-dom)

**JSON**

* The JSON object contains methods for parsing JavaScript Object Notation (JSON) and converting values to JSON. It can't be called or constructed, and aside from its two method properties, it has no interesting functionality of its own.
* JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications (e.g., sending some data from the server to the client, so it can be displayed on a web page, or vice versa).
* Ref: <https://www.w3schools.com/js/js_json_intro.asp>

**JavaScript and JSON differences**

JSON is a syntax for serializing objects, arrays, numbers, strings, booleans, and null. It is based upon JavaScript syntax but is distinct from it: some JavaScript is not JSON.