JAVASCRIPT DOCUMENTATION

**History of JavaScript**

**1. Origins (1995)**

* **Creator:** Brendan Eich, while working at Netscape Communications.
* **Initial Name:** Mocha, later renamed to LiveScript, and finally JavaScript.
* **Purpose:** Designed to add interactivity to web pages. It was a lightweight, interpreted language that could run in the browser.

**2. Early Development**

* **First Release:** Netscape Navigator 2.0 in 1995.
* **Standardization:** JavaScript was submitted to the European Computer Manufacturers Association (ECMA) for standardization. The first edition of the ECMAScript standard was published in 1997 (ECMAScript 1).

**3. Evolution**

* **ECMAScript 2 (1998):** Minor updates and corrections.
* **ECMAScript 3 (1999):** Major update introducing regular expressions, better string handling, new control statements, and more.
* **ECMAScript 4 (2008):** A significant revision was proposed but was abandoned due to complexity and disagreements.
* **ECMAScript 5 (2009):** Introduced 'strict mode,' JSON support, and new methods for objects and arrays.
* **ECMAScript 6 (2015):** Also known as ES6 or ECMAScript 2015, it brought major features like classes, modules, arrow functions, promises, and let/const for block-scoped variables.
* **Subsequent Editions:** ES7 (2016), ES8 (2017), ES9 (2018), ES10 (2019), ES11 (2020), ES12 (2021), ES13 (2022), and ES14 (2023) have introduced new features and improvements annually.

**JavaScript Engine**

**1. Definition:**

* A JavaScript engine is a program that interprets and executes JavaScript code. It converts JavaScript code into machine code that a computer’s CPU can execute.

**2. Major Engines:**

* **V8:** Developed by Google, used in Chrome and Node.js. Known for its performance and Just-In-Time (JIT) compilation.
* **SpiderMonkey:** Developed by Mozilla, used in Firefox. It was the first JavaScript engine and includes JIT compilation.
* **JavaScriptCore (Nitro):** Developed by Apple for Safari, featuring a highly optimized JIT compiler.
* **Chakra:** Developed by Microsoft, used in Internet Explorer and older versions of Edge. It also includes JIT compilation.

**What is JavaScript?**

**1. Definition:**

* JavaScript is a high-level, interpreted scripting language that conforms to the ECMAScript specification. It is used primarily to create dynamic and interactive effects within web browsers.

**2. Key Features:**

* **Dynamically Typed:** Types are associated with values rather than variables.
* **Prototype-Based:** Objects can inherit properties and methods from other objects.
* **Event-Driven:** Allows handling of events such as user actions or changes in state.
* **Asynchronous Programming:** Supports callbacks, promises, and async/await for handling asynchronous operations.

**JavaScript Terms**

\*\*1. **Variables**: Storage for data (e.g., let, const, var).

* **Functions:** Blocks of code designed to perform a particular task (e.g., function name() {}).
* **Objects:** Collections of properties and methods (e.g., let obj = { key: 'value' }).
* **Arrays:** Ordered collections of values (e.g., let arr = [1, 2, 3]).
* **Closures:** Functions that capture the lexical scope in which they were defined.
* **Promises:** Objects representing the eventual completion (or failure) of an asynchronous operation.
* **Async/Await:** Syntax for handling asynchronous code in a more synchronous style.

**Uses of JavaScript**

\*\*1. **Web Development:**

* **Client-Side Scripting:** Enhancing user interfaces and experiences on web pages.
* **Server-Side Scripting:** Node.js allows JavaScript to be used for backend development.

\*\*2. **Applications:**

* **Single-Page Applications (SPAs):** Frameworks like React, Angular, and Vue.js.
* **Mobile Development:** Tools like React Native and Ionic.
* **Desktop Applications:** Frameworks like Electron.
* **Game Development:** Libraries such as Phaser.js.

**JavaScript Runtime Environment**

\*\*1. **Browser Environment:**

* **Components:** Includes the JavaScript engine (like V8 in Chrome) and APIs provided by the browser (DOM, CSSOM, Web APIs, etc.).
* **Execution:** JavaScript runs within the browser's environment, interacting with the web page and handling events.