**Key Features of JavaScript**

* **Client-Side Scripting:**[JavaScrip](https://www.geeksforgeeks.org/javascript/)t runs on the user’s[browser,](https://www.geeksforgeeks.org/web-browser/) so has a faster response time without needing to communicate with the [server](https://www.geeksforgeeks.org/client-server-model/).
* **Versatile:** JavaScript can be used for a wide range of tasks, from simple calculations to complex server-side applications.
* **Event-Driven:** JavaScript can respond to user actions (clicks, keystrokes) in real-time.
* **Asynchronous:** JavaScript can handle tasks like [fetching data](https://www.geeksforgeeks.org/how-to-use-the-javascript-fetch-api-to-get-data/)from[servers](https://www.geeksforgeeks.org/what-is-server/) without freezing the user interface.
* **Rich Ecosystem:**There are numerous [libraries and frameworks](https://www.geeksforgeeks.org/javascript-libraries-and-frameworks/)built on JavaScript, such as[React](https://www.geeksforgeeks.org/react/), [Angular](https://www.geeksforgeeks.org/angular-tutorial/), and[Vue.js](https://www.geeksforgeeks.org/vue-js/), which make development faster and more efficient.
* **Imperative Programming**: Focuses on how to perform tasks, controlling the flow of computation. It includes approaches like procedural and object-oriented programming, often using constructs like async/await to handle actions.
* **Declarative Programming**: Focuses on what should be done rather than how it’s done. It emphasizes describing the desired result, like with arrow functions, without detailing the steps to achieve it.

**Why JavaScript is known as a lightweight programming language ?**

[**JavaScript**](https://www.geeksforgeeks.org/javascript/) is considered a lightweight language due to its low CPU usage, minimalist syntax, and ease of implementation. With no explicit data types and a syntax similar to [C++](https://www.geeksforgeeks.org/c-plus-plus/)and [Java,](https://www.geeksforgeeks.org/java/) it’s easy to learn and runs efficiently in browsers. Unlike heavier languages like Dart or [Java](https://www.geeksforgeeks.org/java/), [JavaScript](https://www.geeksforgeeks.org/javascript/), especially with [Node.js](https://www.geeksforgeeks.org/nodejs/), performs faster and uses fewer resources. While it has fewer built-in libraries, this makes it more flexible, though external libraries are often needed for advanced functionality.[JavaScript’s](https://www.geeksforgeeks.org/javascript/)efficiency and simplicity make it a top choice for web development.

**Is JavaScript Compiled or Interpreted or both ?**

JavaScript is both compiled and interpreted. The V8 engine improves performance by first interpreting code and then compiling frequently used functions for speed. This makes JavaScript efficient for modern web apps

**Just-In-Time (JIT)**compilation is a technique used by JavaScript engines (like V8) to improve performance.

**Interpretation**: Initially, the code is interpreted line-by-line by the engine.

* **Hot Code Detection**: The engine identifies frequently executed code, such as often-called functions.
* **Compilation**: The “hot” code is compiled into optimized machine code for faster execution.
* **Execution**: The compiled machine code is then executed directly, improving performance compared to repeated interpretation.
* **JIT** compilation balances between interpretation (for quick startup) and compilation (for faster execution).

**How do you include JavaScript in an HTML document?**

*JavaScript can be included in an HTML document in three ways:*

* *Inline, by placing it directly within HTML elements.*
* *Internal, by placing it within a <script> tag in the HTML document.*
* *External, by linking to a separate .js file using the <script src=”filename.js”></script> tag.*

Variables and datatypes: [Variables and Datatypes in JavaScript - GeeksforGeeks](https://www.geeksforgeeks.org/variables-datatypes-javascript/)

<https://www.geeksforgeeks.org/javascript-variables/>

**Variable Shadowing in JavaScript**

**Variable shadowing** occurs when a variable declared within a certain scope (e.g., a function or block) has the **same name** as a variable in an outer scope. The inner variable **overrides** the outer variable within its scope.

**let** n = 10; *// Global scope*

**function** gfg() {

**let** n = 20; *// Shadows the global 'n' inside this function*

console.log(n); *// Output: 20*

}gfg();

console.log(n);

4.**A character is also a string.** There is no separate type for characters. A single character is also a string.

[Pure Functions in JavaScript - GeeksforGeeks](https://www.geeksforgeeks.org/pure-functions-in-javascript/)

[Difference between Array and Array of Objects in JavaScript - GeeksforGeeks](https://www.geeksforgeeks.org/difference-between-array-and-array-of-objects-in-javascript/)

**How does the == (Equality) operator work?**

*The == operator compares two values for equality after converting both values to a common type (type coercion). It returns true if the values are equal; otherwise, it returns false.*

**How does the === (Strict equality) operator work?**

*The === operator compares two values for equality without converting their types. It returns true if the values are equal and of the same type; otherwise, it returns false.*

**JavaScript in Operator**

The [in-operator](https://www.geeksforgeeks.org/javascript-in-operator/) in JavaScript checks if a specified property exists in an object or if an element exists in an array. It returns a Boolean value.

**let** languages = ["HTML", "CSS", "JavaScript"];

*// true (index 1 exists in the array)*

console.log(1 **in** languages);

*// false (index 3 doesn't exist in the array)*

console.log(3 **in** languages);

**JavaScript instanceof Operator**

The [instanceof operator](https://www.geeksforgeeks.org/instanceof-operator-in-javascript/" \t "_blank) in JavaScript tests if an object is an instance of a particular class or constructor, returning a Boolean value.

**let** languages = ["HTML", "CSS", "JavaScript"];

console.log(languages **instanceof** Array);T

console.log(languages **instanceof** Object);T

console.log(languages **instanceof** String);F

console.log(languages **instanceof** Number);F

[JavaScript String Operators - GeeksforGeeks](https://www.geeksforgeeks.org/javascript-string-operators/?ref=lbp)

TheJavaScript**for…in loop** iterates over the properties of an [object](https://www.geeksforgeeks.org/objects-in-javascript/). It allows you to access each key or property name of an object.

**const** car = {

make: "Toyota",

model: "Corolla",

year: 2020

};

**for** (**let** key **in** car) {

console.log(`**${**key**}**: **${**car[key]**}**`);

}

**Output**

make: Toyota

model: Corolla

year: 2020

For arrays, we should use below loops.

* [For of Loop](https://www.geeksforgeeks.org/javascript-for-of-loop/) if we need to put continue or break in the loop
* [forEach()](https://www.geeksforgeeks.org/javascript-array-foreach-method/)if we need execute something for all elements without any condition