**JAVASCRIPT**

JavaScript is a **programming language**used to create dynamic content for websites. It is a **lightweight**, **cross-platform,** and **single-threaded** programming language. JavaScript is an **interpreted**language that executes code line by line providing more flexibility.

* HTML adds Structure to a web page, CSS styles it and JavaScript brings it to life by allowing users to interact with elements on the page, such as actions on clicking buttons, filling out forms, and showing animations.
* JavaScript on the client side is directly executed in the user's browser. Almost all browsers have JavaScript Interpreter and do not need to install any software. There is also a browser console where you can test your JavaScript code.
* JavaScript is also used on the Server side (on Web Servers) to access databases, file handling and security features to send responses**,** to browsers.

Webpage in Browser - JavaScript Tutorial

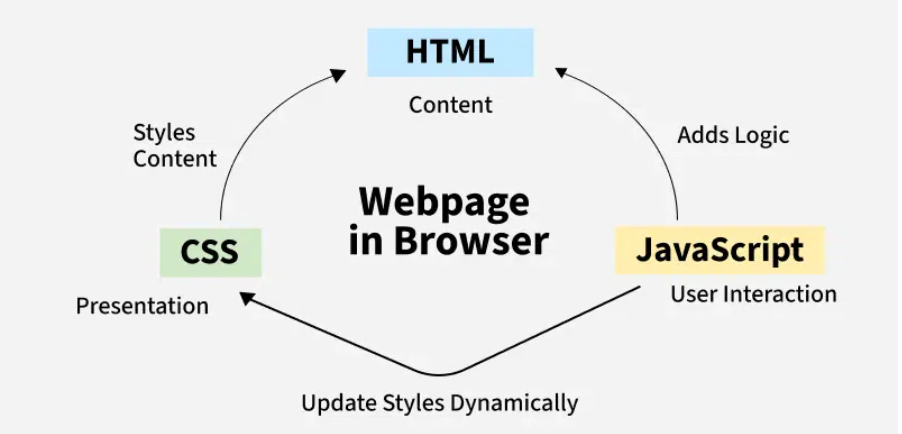
**Hello World Program**

This JavaScript Compiler is completely free and easy to use. Here, you can practice various JS Exercises.

console.log("Hello World!");

**Output**

Hello World!

****

***Webpage in Browser - JavaScript***

**Why to learn JavaScript?**

1. **Versatility**: JavaScript can be used to develop (using [ElectronJS](https://www.geeksforgeeks.org/introduction-to-electronjs/" \t "_blank)) websites, games (Using [Phaser](https://www.geeksforgeeks.org/introduction-to-phaser-js/) and[Three.js](https://www.geeksforgeeks.org/three-js/)), mobile apps (using [React Native](https://www.geeksforgeeks.org/react-native/)), and more.
2. **Client Side:**JavaScript is the main language for client-side logic and is supported by almost all browsers. There is a big list of frameworks and libraries like [React JS](https://www.geeksforgeeks.org/react-tutorial/), [Angular JS](https://www.geeksforgeeks.org/angularjs/), and [Vue JS](https://www.geeksforgeeks.org/vue-js/).
3. **Server-Side**: With runtime environments like [Node.js](https://www.geeksforgeeks.org/nodejs/)and Frameworks like [Express.js](https://www.geeksforgeeks.org/express-js/), JavaScript is now widely used for building server-side applications.
4. **Machine Learning**: With Libraries like [Tensorflow.JS,](https://www.geeksforgeeks.org/tensorflow-js/) JavaScript can be used to develop and train machine learning models. Please refer to [ML in JS](https://www.geeksforgeeks.org/machine-learning-with-javascript/) for details.

**Fundamentals**

Let's Explore **JavaScript's fundamentals** which will help build you a understanding to start with JavaScript

* [Introduction](https://www.geeksforgeeks.org/introduction-to-javascript/)
* [Using JS in HTML](https://www.geeksforgeeks.org/where-to-put-javascript-in-an-html-document/)
* [Variables and Datatypes](https://www.geeksforgeeks.org/variables-datatypes-javascript/)
* **Quiz:**[Variables & Data Types](https://www.geeksforgeeks.org/quizzes/variables-and-data-types-1/)
* [Operators](https://www.geeksforgeeks.org/javascript-operators/)
* [Type Conversion and Coercion](https://www.geeksforgeeks.org/type-conversion-and-type-coercion-in-javascript/)
* [Control Flow Statements](https://www.geeksforgeeks.org/javascript-control-flow-statements/)
* **Quiz:**[Operators](https://www.geeksforgeeks.org/quizzes/javascript-operators/)**,**[Control Flow](https://www.geeksforgeeks.org/quizzes/control-flow/)
* [Scope](https://www.geeksforgeeks.org/understanding-variable-scopes-in-javascript/)
* [Browser console](https://www.geeksforgeeks.org/console-in-javascript/)

**Functions and Events**

Functions in JavaScript are reusable blocks of code that perform a specific task. Events are actions that happen in the browser, such as mouse clicks, keyboard input, or page loading.

* [Functions](https://www.geeksforgeeks.org/functions-in-javascript/)
* [Function binding](https://www.geeksforgeeks.org/javascript-function-binding/)
* [Hoisting](https://www.geeksforgeeks.org/javascript-hoisting/)
* [Closures](https://www.geeksforgeeks.org/closure-in-javascript/)
* [Higher-Order Functions](https://www.geeksforgeeks.org/javascript-higher-order-functions/)
* [Iterator](https://www.geeksforgeeks.org/javascript-iterator/)
* [Function Generator](https://www.geeksforgeeks.org/javascript-function-generator/)
* **Quiz:**[Functions](https://www.geeksforgeeks.org/quizzes/javascript-functions/)
* [Events](https://www.geeksforgeeks.org/javascript-events/)
* [Event Loop](https://www.geeksforgeeks.org/what-is-an-event-loop-in-javascript/)
* [Event Bubbling](https://www.geeksforgeeks.org/event-bubbling-in-javascript/)
* **Quiz:** [Event Handling](https://www.geeksforgeeks.org/quizzes/javascript-events/)

**JavaScript Beginner Projects**

Now you have a basic understanding of JavaScript. So start with some beginner level projects to clear your concept and to implement in real world applications.

* [Counter Application](https://www.geeksforgeeks.org/design-a-simple-counter-using-html-css-and-javascript/)
* [Prime Number Checker](https://www.geeksforgeeks.org/javascript-application-to-check-prime-and-non-prime-number/)
* [Show and Hide Password](https://www.geeksforgeeks.org/show-hide-password-using-javascript/)
* [Palindrome Checker App](https://www.geeksforgeeks.org/build-a-palindrome-checker-app-using-javascript/)
* [JavaScript Carousel](https://www.geeksforgeeks.org/building-a-carousel-with-vanilla-javascript/)
* [Email Validator App](https://www.geeksforgeeks.org/javascript-application-for-email-validation/)
* [Unicode Character Value](https://www.geeksforgeeks.org/javascript-application-get-unicode-character-value/)
* [Random Number Generator](https://www.geeksforgeeks.org/javascript-application-for-random-number-generator/)
* [Random Password Generator](https://www.geeksforgeeks.org/how-to-generate-a-random-password-using-javascript/)

**JavaScript Data Structure**

JavaScript provides a versatile set of data structures that help in efficient data storage, manipulation, and problem-solving. In this section, we will explore each data structure and algorithm in detail.

* [Numbers](https://www.geeksforgeeks.org/javascript-numbers/)
* [String](https://www.geeksforgeeks.org/string-in-javascript/)
* **Quiz:**[Numbers](https://www.geeksforgeeks.org/quizzes/javascript-numbers/), [String](https://www.geeksforgeeks.org/quizzes/javascript-strings-1/?ref=quiz_lbp)
* [Array](https://www.geeksforgeeks.org/javascript-arrays/)
* [LinkedList](https://www.geeksforgeeks.org/implementation-linkedlist-javascript/)
* **Quiz:**[Arrays](https://www.geeksforgeeks.org/quizzes/javascript-arrays-1/)**,**[Linked List](https://www.geeksforgeeks.org/quizzes/javascript-linked-list/)
* [Map](https://www.geeksforgeeks.org/javascript-map/)
* [Stack](https://www.geeksforgeeks.org/implementation-stack-javascript/)
* [Queue](https://www.geeksforgeeks.org/implementation-queue-javascript/)
* **Quiz:**[Stack](https://www.geeksforgeeks.org/quizzes/javascript-stack/)**,**[Queue](https://www.geeksforgeeks.org/quizzes/javascript-queue/)
* [Sorting Algorithms](https://www.geeksforgeeks.org/sorting-algorithms-in-javascript/)

JavaScript's built-in utilities and ES6+ enhancements provide additional data structures, including the following

* [WeakMap](https://www.geeksforgeeks.org/javascript-weakmap/)
* [WeakSet](https://www.geeksforgeeks.org/javascript-weakset/)
* [Typed Arrays](https://www.geeksforgeeks.org/typedarray-introduction/)
* [Deque](https://www.geeksforgeeks.org/deque-in-javascript/)
* [Priority Queue (Heap)](https://www.geeksforgeeks.org/implementation-priority-queue-javascript/)
* **Quiz:** [Data Structure](http://www.geeksforgeeks.org/javascript-data-structures-quizzes/)

**Object Oriented Programming**

Object-Oriented Programming (OOP) in JavaScript, a concept that enables the structure of code by modeling real-world entities as objects with properties and behaviors.

* [Introduction to OOP](https://www.geeksforgeeks.org/introduction-object-oriented-programming-javascript/)
* [Objects](https://www.geeksforgeeks.org/javascript-object-constructors/)
* **Quiz:**[Objects](https://www.geeksforgeeks.org/quizzes/javascript-objects/?ref=quiz_lbp)
* [Classes](https://www.geeksforgeeks.org/javascript-classes/)
* [Constructor Method](https://www.geeksforgeeks.org/javascript-object-prototype-constructor-property/)
* [this Keyword](https://www.geeksforgeeks.org/this-in-javascript/)
* [Prototype](https://www.geeksforgeeks.org/prototype-in-javascript/)
* [Static Methods](https://www.geeksforgeeks.org/static-methods-in-javascript/)
* [Inheritance](https://write.geeksforgeeks.org/preview/javascript-inheritance/)
* **Quiz:**[Classes and Inheritance](https://www.geeksforgeeks.org/quizzes/classes-and-inheritance/)
* [Encapsulation](https://www.geeksforgeeks.org/encapsulation-in-javascript/)
* [Abstraction](https://www.geeksforgeeks.org/abstraction-in-javascript/)
* [Polymorphism](https://www.geeksforgeeks.org/polymorphism-in-javascript/)
* [Getters and Setters](https://www.geeksforgeeks.org/javascript-getters-and-setters/)
* **Quiz:**[OOP](https://www.geeksforgeeks.org/quizzes/oop-in-javascript/)

**Browser and Document Object Model**

* [Browser Object Model](https://www.geeksforgeeks.org/browser-object-model/)
* [Document Object Model](https://www.geeksforgeeks.org/javascript-html-dom/)
* [Manipulate DOM Elements](https://www.geeksforgeeks.org/how-to-manipulate-dom-elements-in-javascript/)
* [Event Handling in the DOM](https://www.geeksforgeeks.org/how-to-use-dom-and-events/)
* **Quiz:**[BOM and DOM](https://www.geeksforgeeks.org/quizzes/javascript-dom-and-bom/)

**Asynchronous JavaScript**

* [Callbacks](https://www.geeksforgeeks.org/javascript-callbacks/)
* [Promise](https://www.geeksforgeeks.org/javascript-promise/)
* [Promise Chaining](https://www.geeksforgeeks.org/javascript-promise-chaining/)
* [Async/Await](https://www.geeksforgeeks.org/async-await-function-in-javascript/)
* **Quiz:**[Asynchronous JavaScript](https://www.geeksforgeeks.org/quizzes/asynchronous-javascript-1/)

**JavaScript Intermediate Projects**

Now you have a good understanding of JavaScript. So let's implement all these in some real world applications.

* [Price Range Slider with Min-Max Input](https://www.geeksforgeeks.org/price-range-slider-with-min-max-input-using-html-css-and-javascript/)
* [GitHub Profile Search](https://www.geeksforgeeks.org/how-to-create-a-github-profile-search-using-html-css-and-javascript/)
* [Toast Notification](https://www.geeksforgeeks.org/how-to-make-a-toast-notification-in-html-css-and-javascript/)
* [Multi-Step Progress Bar](https://www.geeksforgeeks.org/how-to-create-multi-step-progress-bar-using-bootstrap/)
* [Quiz App with Timer](https://www.geeksforgeeks.org/create-a-quiz-app-with-timer-using-html-css-and-javascript/)
* [Expense Tracker](https://www.geeksforgeeks.org/build-an-expense-tracker-with-html-css-and-javascript/)
* [Sortable and Filterable Table](https://www.geeksforgeeks.org/create-a-sortable-and-filterable-table-using-javascript/)
* [Dynamic Resume Builder](https://www.geeksforgeeks.org/dynamic-resume-creator-using-html-css-and-javascript/)
* [OTP Input Field](https://www.geeksforgeeks.org/create-otp-input-field-using-html-css-and-javascript/)
* [Student Grade Calculator](https://www.geeksforgeeks.org/design-a-student-grade-calculator-using-javascript/)

**JavaScript JSON**

It is a lightweight data format for storing and exchanging data widely used to send data between a server and a client.

* [JSON Tutorial](https://www.geeksforgeeks.org/json/)
* [JSON vs JavaScript Object](https://www.geeksforgeeks.org/json-vs-javascript-object/)
* [Read JSON File Using JS](https://www.geeksforgeeks.org/read-json-file-using-javascript/)
* [Parse JSON Data in JS](https://www.geeksforgeeks.org/how-to-parse-json-data-in-javascript/)
* [JavaScript JSON Parser](https://www.geeksforgeeks.org/javascript-json-parser/)
* [JavaScript JSON Complete Reference](https://www.geeksforgeeks.org/javascript-json-complete-reference/)
* **Quiz:**[JSON](https://www.geeksforgeeks.org/quizzes/javascript-json/)

**Regular Expression and Validation**

Regular expressions plays important role for validation. Validations help ensure that data entered by users meets specific criteria.

* [Regular expressions](https://www.geeksforgeeks.org/javascript-regexpregular-expression/)
* [Form Validation](https://www.geeksforgeeks.org/form-validation-using-html-javascript/)
* [Email Validation](https://www.geeksforgeeks.org/how-to-validate-email-address-using-regexp-in-javascript/)
* [Number Validation](https://www.geeksforgeeks.org/number-validation-in-javascript/)
* [Password Validation](https://www.geeksforgeeks.org/javascript-program-to-validate-password-using-regular-expressions/)
* [URL Validation](https://www.geeksforgeeks.org/how-to-validate-url-using-regular-expression-in-javascript/)
* [UserName Validation](https://www.geeksforgeeks.org/username-validation-in-js-regex/)
* **Quiz:**[Regular Expressions](https://www.geeksforgeeks.org/quizzes/regular-expressions-and-json/)

**Exception and Error Handling**

Exception and Error handling is crucial for ensuring the reliability and stability of JavaScript applications by handling errors effectively

* [Exception Handling](https://www.geeksforgeeks.org/javascript-error-and-exceptional-handling-with-examples/)
* [throw Statement](https://www.geeksforgeeks.org/javascript-errors-throw-and-try-to-catch/)
* [try-catch Statement](https://www.geeksforgeeks.org/javascript-errors-throw-and-try-to-catch/)
* [Debugging](https://www.geeksforgeeks.org/debugging-in-javascript/)
* **Quiz:** [Error Handling & Debugging](https://www.geeksforgeeks.org/quizzes/error-handling-and-debugging/)

**Testing and Performance Optimization**

* [Unit testing with Jest](https://www.geeksforgeeks.org/testing-with-jest/)
* [Memory Management](https://www.geeksforgeeks.org/memory-management-in-javascript/)
* [Garbage Collection](https://www.geeksforgeeks.org/garbage-collection-in-javascript/)
* [Lazy Loading](https://www.geeksforgeeks.org/what-is-lazy-loading/)
* [Debouncing](https://www.geeksforgeeks.org/debouncing-in-javascript/)
* [Throttling](https://www.geeksforgeeks.org/javascript-throttling/)
* **Quiz:**[Testing & Optimization](https://www.geeksforgeeks.org/quizzes/javascript-testing-and-optimization/)

**Interesting Facts**

This section covers all the interesting facts and features which made JavaScript so popular and will easy if you are switching from other programming languages also.

* [Data Types](https://www.geeksforgeeks.org/interesting-facts-about-javascript-data-types/)
* [Strings](https://www.geeksforgeeks.org/interesting-facts-about-javascript-strings/)
* [Functions](https://www.geeksforgeeks.org/interesting-facts-about-javascript-functions/)
* ['this' keyword](https://www.geeksforgeeks.org/javascript-interesting-facts-about-this-keyword/)
* [Set](https://www.geeksforgeeks.org/interesting-facts-about-javascript-set/)
* [Map](https://www.geeksforgeeks.org/interesting-facts-about-map-in-javascript/)
* [Arrays](https://www.geeksforgeeks.org/interesting-facts-about-javascript-arrays/)
* [Object](https://www.geeksforgeeks.org/interesting-facts-about-object-in-javascript/)
* [Complete JavaScript](https://www.geeksforgeeks.org/interesting-facts-about-javascript/)

**JavaScript Advanced Projects**

Now you have covered almost all the important concepts of JavaScript. These projects will improve and revise your JavaScript Knowledge.

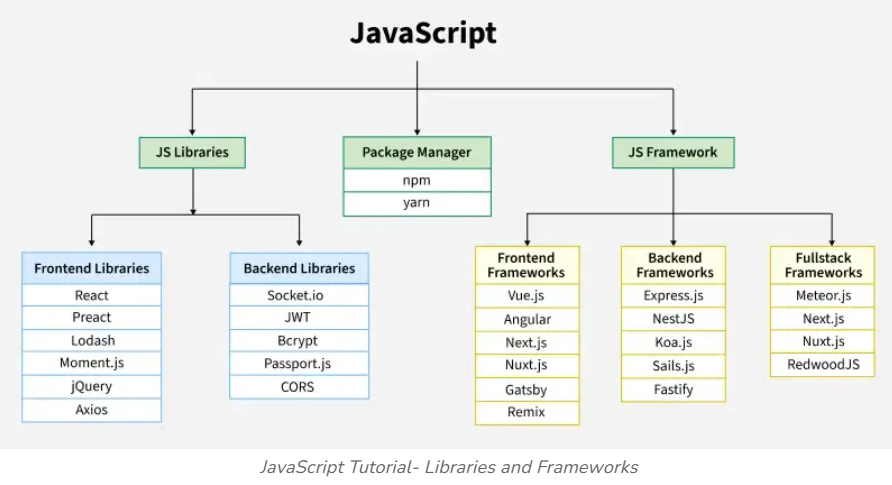
* [Employee Database Management System](https://www.geeksforgeeks.org/employee-database-management-system-using-html-css-and-javascript/)
* [Nested Chat Comments](https://www.geeksforgeeks.org/design-a-nested-chat-comments-using-html-css-and-javascript/)
* [Responsive Admin Dashboard](https://www.geeksforgeeks.org/how-to-create-responsive-admin-dashboard-using-html-css-javascript/)
* [Task Scheduler](https://www.geeksforgeeks.org/task-scheduler-using-html-css-and-js/)
* [Dragon’s World Game](https://www.geeksforgeeks.org/design-dragons-world-game-using-html-css-and-javascript/)
* [Tic-Tac-Toe Game](https://www.geeksforgeeks.org/simple-tic-tac-toe-game-using-javascript/)
* [QR Code Generator](https://www.geeksforgeeks.org/create-an-qr-code-generator-project-using-html-css-javascript/)
* [Resize and Compress Images](https://www.geeksforgeeks.org/create-a-resize-and-compress-images-in-html-css-javascript/)
* [QR Code Scanner or Reader](https://www.geeksforgeeks.org/create-a-qr-code-scanner-or-reader-in-html-css-javascript/)

**Libraries and Frameworks**

JavaScript libraries and frameworks play a important role in modern web development. They offer built-in functions and methods that enhance web pages, making them more dynamic and interactive. They handle repetitive tasks, allowing developers to focus on core functionality.

Also, they provide project structure and data flow structure that helps to create

fast and more reliable applications.



JavaScript Tutorial- Libraries and Frameworks

**Libraries**

Libraries provide pre-built solutions for common tasks. Developers can use these functions instead of writing code from scratch, saving valuable time. Here are a few popular libraries of JavaScript.

* **Frontend Libraries**[**:** React](https://www.geeksforgeeks.org/react/), [Preact](https://www.geeksforgeeks.org/difference-between-preact-and-angular/" \t "_blank), [Lodash](https://www.geeksforgeeks.org/lodash/" \t "_blank),[Moment.js](https://www.geeksforgeeks.org/moment-js/),[jQuery](https://www.geeksforgeeks.org/moment-js/),[Axios](https://www.geeksforgeeks.org/axios-in-react-a-guide-for-beginners/)
* **Backend Libraries:** [Socket.io](https://www.geeksforgeeks.org/introduction-to-sockets-io-in-node-js/),[JWT](https://www.geeksforgeeks.org/json-web-token-jwt/), [Bcrypt](https://www.geeksforgeeks.org/npm-bcrypt/" \t "_blank), [Passport.js](https://www.geeksforgeeks.org/explain-passport-in-node-js/), [CORS](https://www.geeksforgeeks.org/npm-cors/)

**Frameworks**

Frameworks, offer a comprehensive structure for building applications. Here are a few popular frameworks of JavaScript.

* **Frontend Frameworks:** [Vue.js](https://www.geeksforgeeks.org/vue-js/), [Angular](https://www.geeksforgeeks.org/angular-tutorial/), [Next.js](https://www.geeksforgeeks.org/nextjs/), [Nuxt.js](https://www.geeksforgeeks.org/nuxtjs/), [Gatsby](https://www.geeksforgeeks.org/gatsby-introduction/),[Remix](https://www.geeksforgeeks.org/getting-started-with-remix/)
* **Backend Frameworks:** [Express.js](https://www.geeksforgeeks.org/express-js/), [NestJS](https://www.geeksforgeeks.org/nestjs/" \t "_blank), [Koa.js](https://www.geeksforgeeks.org/express-js-vs-koajs-in-node-js/), Sails.js, [Fastify](https://www.geeksforgeeks.org/difference-between-express-and-fastify-web-app-frameworks/" \t "_blank)
* **FullStack FrameWorks:** [Meteor.js](https://www.geeksforgeeks.org/meteor-introduction-to-meteor/), Next.js, Nuxt.js, RedwoodJS

**Features of JavaScript**

JavaScript is one of the most widely used programming languages, known for its flexibility and versatility in web development. Here are some key features or characteristics:

* **Easy to Learn and Use**: JavaScript's syntax is simple and straightforward, making it easy for both beginners and experienced developers to pick up.
* **Interpreted Language**: JavaScript is an interpreted language, means it is executed directly by the browser without the need for a compiler.
* **Event-Driven and Asynchronous**: JavaScript is designed to handle events, making it ideal for interactive web pages. It supports asynchronous programming, enabling non-blocking operations.
* **Object-Oriented and Functional**: JavaScript supports both object-oriented and functional programming. This gives developers the flexibility to structure their code according to their need.
* **Dynamically Typed**: In JavaScript, you don’t need to declare data types when declaring variables. The language automatically determines the data type at runtime, making it more flexible.
* **Extensive Libraries and Frameworks**: JavaScript has a rich ecosystem of libraries and frameworks such as React, Angular, and Vue.js, which simplify and speed up development for both front-end and back-end projects.
* **Cross-Platform**: JavaScript is supported across all modern browsers, allowing the same code to run seamlessly on different operating systems and platforms.
* **Large Community Support**: JavaScript benefits from a large, active developer community that contributes to its growth. With numerous resources available, finding help or solutions is always easy.

**Applications of JavaScript**

JavaScript is a versatile language that powers various applications, from web development to mobile apps, making it an essential tool for modern programming.

* [**Web Development**](https://www.geeksforgeeks.org/web-development/): JavaScript is widely used in web development to create interactive and dynamic websites. Frameworks like React and Angular make front-end development faster, while Node.js is used for building server-side applications.
* [**Mobile App Development**](https://www.geeksforgeeks.org/top-javascript-frameworks-for-mobile-apps-development/): JavaScript helps in developing mobile apps using frameworks like React Native, allowing developers to build cross-platform apps for both iOS and Android.
* [**Game Development**](https://www.geeksforgeeks.org/top-javascript-frameworks-libraries-for-building-games/): JavaScript is also used for creating browser-based games with libraries like Phaser, making it easy to develop 2D games that run directly in the browser.
* [**Server-Side Development**](https://www.geeksforgeeks.org/javascript-backend-basics/): With Node.js, JavaScript is used for server-side programming, enabling developers to build scalable applications and APIs, especially for real-time features like chat systems.
* [**Scripting & Automation**](https://www.geeksforgeeks.org/what-is-scripting/): JavaScript is ideal for automating web-related tasks like form validation and data manipulation, improving efficiency and reducing manual work.
* [**Web Scraping**](https://www.geeksforgeeks.org/what-is-web-scraping-in-node-js/): JavaScript, along with libraries like [Puppeteer](https://www.geeksforgeeks.org/npm-puppeteer/), is used to extract information from websites for data analysis or research, making it useful for web scraping tasks.
* [**IoT (Internet of Things)**](https://www.geeksforgeeks.org/javascript-for-iot/): JavaScript is used to control devices and sensors in IoT projects, allowing developers to build smart systems with frameworks like Johnny-Five.
* [**Real-Time Applications**](https://www.geeksforgeeks.org/how-to-implement-real-time-features-in-your-web-app-with-example/): JavaScript powers real-time applications, such as live chats or notifications, using technologies like WebSockets and Socket.io for instant communication between users and servers.

**JavaScript vs. Other Programming Languages**

Below is a comparison of JavaScript with Python, C++, and Java:

| **Feature** | **JavaScript** | **Python** | **C++** | **Java** |
| --- | --- | --- | --- | --- |
| **Type** | Interpreted | Interpreted | Compiled | Compiled and Interpreted |
| **Paradigm** | Multi-paradigm (event-driven, functional, object-oriented) | Multi-paradigm (object-oriented, procedural, functional) | Multi-paradigm (procedural, object-oriented, generic) | Object-oriented, structured |
| **Memory Management** | Automatic (Garbage collection) | Automatic (Garbage collection) | Manual | Automatic (Garbage collection) |
| **Syntax** | Simple | Simple | Complex | Complex |
| **Use Cases** | Web development, mobile apps, real-time applications | Web development, data analysis, machine learning | System programming, game development, high-performance applications | Large-scale applications, enterprise software |
| **Notable Frameworks/Libraries** | React, Angular, Node.js | Django, Flask | Standard Library, Boost | Spring, Hibernate |
| **Community Support** | Strong | Strong | Strong | Strong |
| **Job Market** | Abundant | Abundant | Abundant | Abundant |

**List of Companies Using JavaScript**

These are some popular companies that use JavaScript in their workflow

| **Company** | **Description** |
| --- | --- |
| **Google** | Google uses JavaScript extensively in its web services, including Gmail and Google Maps. |
| **Facebook** | Facebook’s front-end is built with React, a JavaScript library, for a dynamic user interface. |
| **Netflix** | Netflix uses JavaScript for their interactive user interface and in their backend services. |
| **Airbnb** | Airbnb relies on JavaScript, using Node.js for scalable back-end development and React for the front end. |
| **LinkedIn** | LinkedIn uses JavaScript for both front-end and back-end development, enabling real-time interaction. |
| **Uber** | JavaScript is used by Uber for real-time tracking and routing, employing both front-end and back-end technologies. |
| **Twitter** | Twitter uses JavaScript for handling its dynamic feeds and providing a seamless user experience. |
| **PayPal** | PayPal’s front-end is powered by JavaScript, offering smooth, interactive user experiences. |
| **eBay** | eBay uses JavaScript to build their responsive, dynamic web applications and enhance user interactions. |
| **Slack** | Slack's web client and several of its real-time communication features are powered by JavaScript. |