



JavaScript Tutorial

Last Updated : 18 Jul, 2024

JavaScript is the most powerful and versatile web programming language. It is used for making the websites interactive. JavaScript helps us add features like animations, interactive forms and dynamic content to web pages.

In this **JavaScript Tutorial**, we'll learn all the **basics to advanced** topics and concepts of **JavaScript**. This JavaScript tutorial includes **operators, control flow, functions, objects, Asynchronous JavaScript, Events, DOM manipulation, OOPs, Closures, Event loops** etc.

What is JavaScript?

JavaScript is a **programming language used for creating dynamic content on 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. It is a commonly used programming language to **create dynamic and interactive elements in web applications**. It is easy to learn.

Hello World Program in JavaScript

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

JavaScript



```
console.log("Hello World! I Don't Give a Bug");
```



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

Hello World! I Don't Give a Bug

Table of Content

- [JavaScript Basics](#)
- [JavaScript Control Flow Statements](#)
- [JavaScript Loops](#)
- [JS Expression and Operators](#)
- [JavaScript Objects](#)
- [JavaScript Functions](#)
- [JavaScript Regular Expression](#)
- [JavaScript Events](#)
- [JavaScript OOPs](#)
- [JavaScript Inheritance and Prototype Chain](#)
- [JavaScript Memory Management](#)
- [JavaScript Promises](#)
- [JavaScript Iterators and generators](#)
- [JavaScript Validations](#)
- [JavaScript Exception Handling](#)
- [JavaScript Global Objects](#)
- [JavaScript Miscellaneous](#)
- [JavaScript Practice Quiz](#)
- [JavaScript Interview Questions](#)

Prerequisites to learn JavaScript

HTML: Learn the fundamental of HTML to create structure of web page.



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

JavaScript is an essential programming language for web developers to learn as it grows rapidly. JavaScript is responsible for behaviour of the web pages. By learning it, you will develop skills in web development.

Key Features of JavaScript:

1. **Versatility:** JavaScript can be used to develop websites, games, mobile apps, and more.
2. **Client and Server-Side:** With frameworks like [Node.js](#) and [Express.js](#), JavaScript is now used for building server-side applications.
3. **End-to-End Solutions:** JavaScript enables developers to create complete solutions for various problems.
4. **Constant Evolution:** JavaScript continually evolves with new features and standards.
5. **Vibrant Community:** A large community of users and mentors actively contributes to JavaScript's growth.

JavaScript Basics

Let's Explore **JavaScript's fundamentals**: variables, data types, operators, and control flow for web page dynamism.

- [Introduction to JavaScript](#)
- [History of JavaScript](#)
- [JavaScript Code Execution](#)
- [Console in JavaScript](#)
- [JavaScript Versions](#)
- [JavaScript Syntax](#)
- [JavaScript Comments](#)
- [Where to Put JavaScript in HTML](#)
- [JavaScript First Program Hello world](#)
- [JavaScript Data types](#)
- [JavaScript Variables](#)
 - [let Keyword](#)
 - [var Keyword](#)
 - [const Keyword](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Conditional Statements and Loops in JavaScript (Using if...else structures and loops (like for and while) to control program flow based on conditions and execute code repeatedly).

- [JS return Statement](#)
- [JS break Statement](#)
- [JS continue Statement](#)
- [JS throw Statement](#)
- [JS if...else Statement](#)
- [JS switch Statement](#)
- [JS try...catch Statement](#)

JavaScript Loops

Repetition Made Easy (Navigate for, while, and do...while for Efficient Code).

- [JS for Loop](#)
- [JS do...while Loop](#)
- [JS while Loop](#)
- [JS for...in Loop](#)
- [JS for...of Loop](#)
- [JS labeled Statement](#)
- [JS break Statement](#)
- [JS continue Statement](#)

JS Expression and Operators

You'll learn the syntax and usage of expressions, which are combinations of values, variables, operators, and functions that produce a result.

- [JS Assignment operators](#)
- [JS Comparison operators](#)
- [JS Arithmetic operators](#)
- [JS Bitwise operators](#)
- [JS Logical operators](#)
- [JS BigInt Operators](#)
- [JS String operators](#)
- [JS Ternary operator](#)
- [JS Comma operator](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

JavaScript Objects

Objects in JavaScript are complex data types that allow for the creation of structured entities with properties and methods.

- [JS Array](#)
- [JS String](#)
- [JS Date](#)
- [JS Number](#)
- [JS Math](#)
- [JS Object](#)
- [JS Boolean](#)
- [JS JSON](#)
- [JS Map](#)
- [JS Set](#)
- [JS Atomics](#)
- [JS BigInt](#)
- [JS Promise](#)
- [JS Proxy](#)
- [JS Reflect](#)
- [JS WeakMap](#)
- [JS WeakSet](#)

JavaScript Functions

Functions in JavaScript are reusable blocks of code that perform a specific task.

- [JS Functions](#)
- [JS Function Definitions](#)
- [JS Function Call](#)
- [JS Function Expression](#)
- [JS Pure Functions](#)
- [JS Function Parameters](#)
- [JS Function Invocation](#)
- [JS Anonymous Functions](#)
- [JS Arrow functions](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

- [JS Async/Await Function](#)
- [Hoisting in JavaScript](#)

JavaScript Regular Expression

Regular expressions, often abbreviated as regex or regexp, are patterns used to match character combinations in strings.

- [JS Regular expressions](#)
- [JS RegExp \[abc\] Expression](#)
- [JS RegExp S Metacharacter](#)
- [JS RegExp m Modifier](#)
- [JS RegExp ?! Quantifier](#)
- [JS RegExp {X,Y} Quantifier](#)
- [JS RegExp test\(\) Method](#)
- [JS RegExp \[^0-9\] Expression](#)

JavaScript Events

Events are actions that happen in the browser, such as mouse clicks, keyboard input, or page loading. This part of the content explores how to work with events in JavaScript, including event handling, event listeners, event propagation, event objects.

- [JS Events](#)
- [JS onclick Event](#)
- [JS dblclick Event](#)
- [JS onload Event](#)
- [JS onresize Event](#)
- [JS onblur Event](#)
- [JS onchange Event](#)
- [JS Focus Event](#)
- [JS Clipboard Event](#)
- [JS onscroll Event](#)

JavaScript OOPs

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

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

- [JS class expression](#)
- [JS Object Constructors](#)
- [JS Static Methods](#)
- [JS Prototype](#)
- [JS Constructor Method](#)
- [JS Encapsulation](#)
- [JS Inheritance](#)
- [JS Polymorphism](#)
- [JS Abstraction](#)

JavaScript Inheritance and Prototype Chain

Inheritance allows objects to inherit properties and methods from other objects, enabling code reuse and creating hierarchical relationships between objects.

- [Prototype Inheritance in JavaScript](#)
- [Prototype Chain in JavaScript](#)
- [JS Object Constructors](#)
- [JS Prototype in JavaScript](#)

JavaScript Memory Management

JavaScript uses automatic memory management, known as garbage collection, to allocate and deallocate memory dynamically.

- [JS Memory Management](#)
- [JS Garbage Collection](#)

JavaScript Promises

JS Promises provide a cleaner and more intuitive way to work with asynchronous code compared to traditional callback-based approaches.

- [JS Promise](#)
- [JS Promise Chaining](#)
- [JS Errors Throw and Try to Catch](#)
- [JS Class compositions in JavaScript](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

pause and resume their execution to produce a sequence of values lazily.

- [JS Iterator](#)
- [JS Function Generator](#)

JavaScript Validations

JavaScript validations help ensure that data entered by users meets specific criteria, such as format, length, or range, before submitting it to the server.

- [JS Form Validation](#)
- [JS Email Validation](#)

JavaScript Exception Handling

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

- [JS Exception Handling](#)
- [JS try-catch Statement](#)
- [JS Promises](#)
- [JS async/await](#)

JavaScript Global Objects

Global objects play a important role in JavaScript development, offering a wide range of functionalities for various tasks

- [JS encodeURIComponent\(\), encodeURIComponent\(\) and decodeURIComponent\(\) Method](#)
- [JS eval\(\) Method](#)
- [JS globalThis Property](#)
- [JS Global property](#)
- [JS Infinity](#)
- [JS Undefined vs Undeclared](#)

JavaScript Miscellaneous

This section serves as a repository for miscellaneous JavaScript topics that don't fit neatly into other categories but are still valuable for JavaScript

developers to understand

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

- [JS setTimeout\(\) and setInterval\(\) Method](#)
- [JS typeof Operator](#)
- [JS Debugging](#)
- [JS Local Storage](#)
- [JS Callback](#)
- [JS Closures](#)
- [JS defer](#)
- [JS scope](#)
- [JS Void](#)

JavaScript Practice Quiz

This section offers a collection of practice Quiz designed to test and support understanding of various concepts in JavaScript.

- [Practice Quiz-1](#)
- [Practice Quiz-2](#)
- [Practice Quiz-3](#)

JavaScript Interview Questions

This section provides a list of interview questions related to JavaScript.

- [JavaScript Interview Questions and Answers \(2024\) For Beginners](#)
- [Intermediate JavaScript Interview Questions and Answers \(2024\)](#)
- [Advance JS Interview Questions and Answers \(2024\) For Experienced](#)

History of javascript

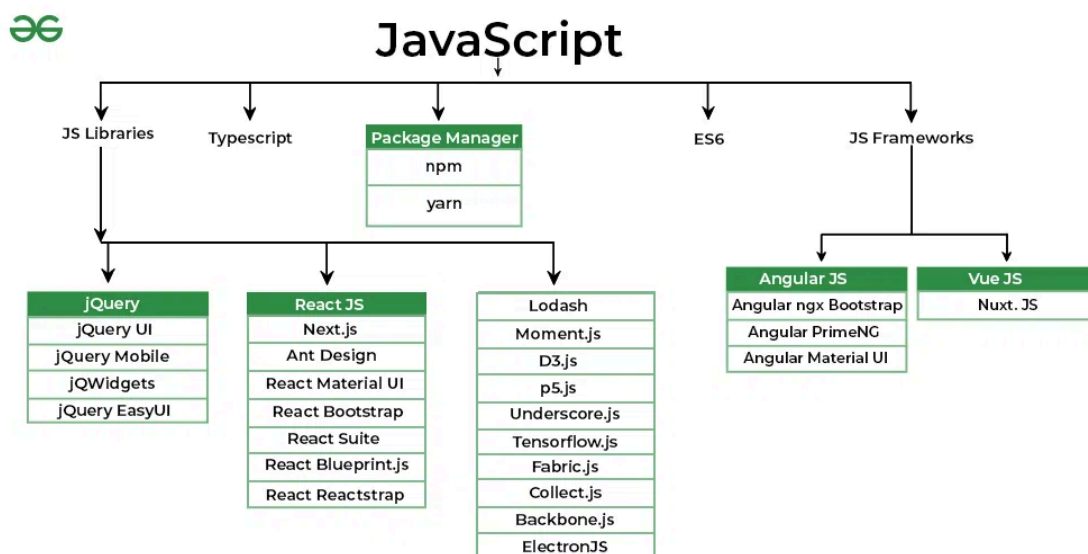
- JavaScript was developed by **Brendan Eich** in **May 1995**.
- **First name of the JavaScript was Mocha**, it later underwent a name change to **LiveScript** before finally settling on **JavaScript**.
- In **1997**, JavaScript was recognized as an **ECMA standard** (ECMA-262).
- **ECMAScript 6 (ES6)**, released in **2015**.

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

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Also, they provide project structure and data flow structure that helps to create fast and more reliable applications.



JavaScript Libraries

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

- [ReactJS](#)
- [jQuery](#)
- [p5.js](#)
- [D3.js](#)
- [Collect.js](#)
- [Underscore.js](#)
- [Lodash](#)
- [Tensorflow.js](#)

JavaScript Frameworks

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

- [AngularJS](#)
- [Vue.js](#)
- [NuxtJS](#)

If you are a person, who is more into study from Courses. [JavaScript](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

What versions of JavaScript does the tutorial cover?

*This **JS tutorial** covers a wide range of JavaScript versions, from the basics to more advanced concepts. It encompasses both older versions like [ES5 and ES6](#), as well as more recent versions and features introduced in ES7, ES8, ES9, and beyond.*

What is JavaScript used for?

JavaScript is primarily used for enhancing web pages by adding interactivity, dynamic content, and user-friendly features. It runs directly in web browsers and allows developers to create responsive and engaging user interfaces.

How do I learn JavaScript from scratch?

To learn JavaScript from scratch, start with the basics:

- *Understand variables, data types, and operators.*
- *Learn about functions, loops, and conditional statements.*
- *Practice by building small projects and gradually move to more complex ones.*

What are JavaScript frameworks?

JavaScript frameworks (such as [Angular](#), or [Vue.js](#)) provide a structured way to build web applications. They offer pre-built components, routing, state management, and other tools to streamline development.

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

this javascript tutorial covers all important [advanced JavaScript concepts](#) such as functional programming, object-oriented programming, asynchronous programming, and more. Beginners can start with the basics and gradually delve into more advanced concepts as they progress through the tutorial.

Three 90 Challenge is back on popular demand! After processing refunds worth INR 1CR+, we are back with the offer if you missed it the first time. Get 90% course fee refund in 90 days. [Avail now!](#)

Want to be a Software Developer or a Working Professional looking to enhance your **Software Development Skills**? Then, master the concepts of Full-Stack Development. Our [Full Stack Development - React and Node.js Course](#) will help you achieve this quickly. Learn everything from **Front-End to Back-End Development** with hands-on **Projects** and real-world examples. This course enables you to build scalable, efficient, dynamic web applications that stand out. Ready to become an expert in Full-Stack? Enroll Now and Start Creating the Future!



75

Next Article

[Introduction to JavaScript](#)

Similar Reads

JavaScript Course Understanding Code Structure in JavaScript

Inserting JavaScript into a webpage is much like inserting any other HTML content. The tags used to add JavaScript in HTML are `<script>` and...

4 min read

[Introduction to JavaScript Course Learn how to build a web browser!](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

4 min read

JavaScript Course Loops in JavaScript

Looping in programming languages is a feature that facilitates the execution of a set of instructions/functions repeatedly while some condition evaluates to...

4 min read

JavaScript Course Logical Operators in JavaScript

logical operator is mostly used to make decisions based on conditions specified for the statements. It can also be used to manipulate a boolean or set...

3 min read

JavaScript Course What is JavaScript ?

JavaScript is a very famous programming language that was originally started in the year of 1995 with the motive of making web pages alive. It is also an...

3 min read

JavaScript Course Operators in JavaScript

An operator is capable of manipulating a certain value or operand. Operators are used to performing specific mathematical and logical computations on...

7 min read

JavaScript Course Functions in JavaScript

Javascript functions are code blocks that are mainly used to perform a particular function. We can execute a function as many times as we want by...

4 min read

JavaScript Course Conditional Operator in JavaScript

JavaScript Conditional Operators allow us to perform different types of actions according to different conditions. We make use of the 'if' statement...

3 min read

JavaScript Course Objects in JavaScript

We have learned about different Data Types that javascript provides us, most

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

JavaScript vs Python : Can Python Overtop JavaScript by 2020?

This is the Clash of the Titans!! And no...I am not talking about the Hollywood movie (don't bother watching it...it's horrible!). I am talking about JavaScript...

5 min read

Article Tags :

[JavaScript](#)

[Web Technologies](#)

[Tutorials](#)

[Web-Tech Tutorials](#)



Corporate & Communications Address:- A-143, 9th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)
| Registered Address:- K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305



Company

[About Us](#)

[Legal](#)

[Careers](#)

[In Media](#)

Explore

[Job-A-Thon Hiring Challenge](#)

[Hack-A-Thon](#)

[GfG Weekly Contest](#)

[Offline Classes \(Delhi/NCR\)](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning Tutorial
ML Maths
Data Visualisation Tutorial
Pandas Tutorial
NumPy Tutorial
NLP Tutorial
Deep Learning Tutorial

Python Tutorial

Python Programming Examples
Django Tutorial
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question

DevOps

Git
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
DSA Interview Questions
Competitive Programming

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
NodeJs
Bootstrap
Tailwind CSS

Computer Science

GATE CS Notes
Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

Commerce

Accountancy
Business Studies
Economics
Management
HR Management

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

SQL
MYSQL
PostgreSQL
PL/SQL
MongoDB

Company-Wise Recruitment Process
Resume Templates
Aptitude Preparation
Puzzles
Company-Wise Preparation
Companies
Colleges

Competitive Exams

JEE Advanced
UGC NET
UPSC
SSC CGL
SBI PO
SBI Clerk
IBPS PO
IBPS Clerk

Free Online Tools

Typing Test
Image Editor
Code Formatters
Code Converters
Currency Converter
Random Number Generator
Random Password Generator

More Tutorials

Software Development
Software Testing
Product Management
Project Management
Linux
Excel
All Cheat Sheets
Recent Articles

Write & Earn

Write an Article
Improve an Article
Pick Topics to Write
Share your Experiences
Internships

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).