**Introduction to java script:-**

**JavaScript** is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language. It is also known as the scripting language for webpages. It is well-known for the development of web pages, and many non-browser environments also use it.

JavaScript is aWeakly typed **(dynamically typed)**. JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript is both an imperative and declarative type of language. JavaScript contains a standard library of objects, like array, date and Math and a core set of language elements like operators, **control structures**, and **statemnets**.

* **Client-side:** It supplies objects to control a browser and its Document Object Model(DOM) Like if client-side extensions allow an application to place elements on an HTML form and respond to user events such as **mouse clicks**, **form input**, and **page navigation** . Useful libraries for the client side are Angular, Node Js, vue js **[,](https://www.geeksforgeeks.org/vue-js/" \t "_blank)** and so many others.
* **Server-side:** It supplies objects relevant to running JavaScript on a server. For if the server-side extensions allow an application to communicate with a database, and provide continuity of information from one invocation to another of the application, or perform file manipulations on a server. The useful framework which is the most famous these days is node js.
* **Imperative language –**In this type of language we are mostly concerned about how it is to be done. It simply controls the flow of computation. The procedural programming approach, object, oriented approach comes under this as async await we are thinking about what is to be done further after the async call.

Why JavaScript?

JavaScript is widely used for web development because it enables interactive and dynamic content on websites. It's supported by all modern web browsers, making it a universal language for client-side scripting. It's also used for server-side development (Node.js), mobile app development (React Native), desktop app development (Electron), game development (Unity), and more.

JavaScript (JS) is a versatile language that can be used for both front-end (FE) and back-end (BE) development.

1. Front-End (FE) Development:

- In front-end development, JavaScript is primarily used to create interactive and dynamic user interfaces on web pages. It allows developers to manipulate the Document Object Model (DOM), handle user interactions, perform client-side form validation, create animations, and fetch data from servers asynchronously.

- JavaScript is often combined with HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) to create modern web applications. Libraries and frameworks such as React, Angular, and Vue.js further enhance JavaScript's capabilities for building complex and responsive front-end applications.

2. Back-End (BE) Development:

- With the introduction of Node.js, JavaScript can also be used for server-side development. Node.js is a runtime environment that allows developers to run JavaScript code outside the browser, making it possible to build scalable and high-performance back-end applications.

- JavaScript frameworks like Express.js provide a robust and minimalist web application framework for Node.js, enabling developers to create RESTful APIs, handle HTTP requests and responses, interact with databases, and implement server-side business logic using JavaScript.

- Additionally, JavaScript can be used for serverless computing, where developers write functions that run in response to events triggered by external sources (e.g., HTTP requests, database changes) without managing the server infrastructure. Platforms like AWS Lambda and Azure Functions support JavaScript as a language for serverless development.

Top 5 Practical Applications of JavaScript:-

1. Web Development

JavaScript is a scripting language used to develop web pages. Developed in Netscape, JS allows developers to create a dynamic and interactive web page to interact with visitors and execute complex actions. It also enables users to load content into a document without reloading the entire page. Most websites use JavaScript for validation and to support external applications, including PDF documents, widgets, flash applications. Some of the world’s largest tech companies use JavaScript to better the user experience.

### **2. Web Applications**

Various JavaScript frameworks are used for developing and building robust web applications. In an application like Google Maps, if users want to explore a map, all they have to do is click and drag the mouse to get a detailed view. This is powered by JavaScript, which interacts with the browser without communicating with the servers. Popular JavaScript front-end frameworks that help build web apps are React native React Angular and Vue. Netflix and PayPal were developed with Angular js JavaScript framework and Application Programming Interfaces (APIs).

### **3. Presentations**

A very popular application of JavaScript is to create interactive presentations as websites. The RevealJs and BespokeJs libraries can be used to generate web-based slide decks using HTML. The RevealJs helps create interactive slide decks with transitions styles, themes, and slide backgrounds in all Css color formats. The BespokeJs is a feature-heavy framework that includes features like scaling, animated bullet lists, syntax highlighting, etc. Even if a user is not fully conversant with a programming language, they can easily build presentations as websites using JavaScript.

### **4. server Applications**

JavaScript is also used to write server-side software through Node.js open-source runtime environment. Developers can write, test and debug code for fast and scalable network applications. JavaScript helps to generate content and manage HTTP requests. Top companies like Walmart, PayPal, Uber, GoDaddy, and many more have adopted Node.js for server infrastructure.

### **5. Web Servers**

Node.js allows developers to use JavaScript to create a web server. Node.js being event-driven, it moves to the next call without waiting for the response of the previous call. The servers quickly transfer chunks of data without buffering. The HTTP module uses the createServer () method for creating a server.

How is JavaScript used?

Introducing JavaScript in HTML is typically done by including `<script>` tags within the HTML document. Here's how you can do it:

1. Inline JavaScript: You can include JavaScript directly within the HTML document using the `<script>` tag. Place the `<script>` tag inside the `<head>` or `<body>` section of your HTML document.

2. External JavaScript File:

Alternatively, you can create a separate JavaScript file with a `.js` extension and include it in your HTML document using the `<script>` tag's `src` attribute.

html

What are APIs and how do they work?

Anyone who works with business software has heard terms like “API” and “webhooks” thrown around. In this article, we’ll not only cover what APIs are, but we’ll also answer the question: “How do APIs work?”

Understanding the function of APIs is the key to understanding how they can help business users in any role accomplish more, faster - without necessarily having to learn coding.

API tools have fundamentally transformed how developers write applications. They have introduced an entirely new vertical of “platform as a service” software companies. API-based tools are the reason why data integrations between essential business software are possible. In fact, [**API integrations**](https://tray.io/blog/what-is-an-api-integration-for-non-technical-people) have become essential to operations and revenue professionals.

API stands for “application programming interface.” An API is essentially a set of rules that dictate how two machines talk to each other. Some examples of API-based interactions include a cloud application communicating with a server, servers pinging each other, or applications interacting with an operating system. Whenever you use an app on your phone or computer or log onto Twitter or Facebook, you’re interacting with several different APIs behind the scenes. Nearly all businesses that use any kind of modern technology use APIs at some level to retrieve data or interact with a database for customers to use.

An API’s defined communication protocol is what enables developers to build, connect, and integrate applications quickly and at scale. Consider, as an example, Jeff Bezos' famously-issued 2002 mandate. Amazon's change of direction shows how APIs helped it move faster than its competitors, and is reportedly the [**reason**](https://www.cio.com/article/3218667/have-you-had-your-bezos-moment-what-you-can-learn-from-amazon.html) Amazon is so successful. Bezos ordered all of his teams to communicate and expose data and functionality through service interfaces, that is, APIs. Once the APIs and infrastructure were in place, Amazon’s teams were able to operate much more efficiently. Launching this new infrastructure enabled the creation of Amazon Web Services, which has since become Amazon’s largest revenue driver.

**Common features in programming languages:-**

**1) Variables**

**2) Operators**

**3) Data Types**

**4) Control Structures**

**5) Functions**

1. **Variables:-**

In programming, we often need a **named storage location** to store the data or values. Using variables, we can store the data in our program and access it afterward. In this article, we will learn about variables in programming, their types, declarations, initialization, naming conventions, etc.

Examples in real:- Money in your bank account, the temperature today, the number of shoes you have, your age, the weather, the price of the new iPhone, the position of the sun in the sky, you can go on and on. Even values that we consider as constants, that means their values are not expected to change, are also variables

1. **Operators:-**

In mathematics and computer [programming](https://www.techtarget.com/searchsoftwarequality/definition/program), an operator is a character that represents a specific mathematical or logical action or process. For instance, "x" is an arithmetic operator that indicates multiplication, while "&&" is a logical operator representing the logical AND function in programming.Depending on its type, an operator manipulates an arithmetic or logical value, or operand, in a specific way to generate a specific result. From handling simple arithmetic functions to facilitating the execution of complex algorithms, like security [encryption](https://www.techtarget.com/searchsecurity/definition/encryption), operators play an important role in the programming world.

**Examples:- +,-,/,%,&.**

**3) Data Types:-**

In Programming, data type is an attribute associated with a piece of data that tells a computer system how to interpret its value. Understanding data types ensures that data is collected in the preferred format and that the value of each property is as expected.

1. **Control Structures:-**

Control structures / Control statements enable a programmer to determine the order in which program statements are executed. These control structures allow you to do two things: 1) skip some statements while executing others, and 2) repeat one or more statements while some condition is true.