**HTML5**

Hyper Text Markup Language, or HTML, is a fundamental web programming language used to create simple webpages. People may share their ideas with the world through websites thanks to HTML code.

An essential tool for creating websites is HTML5, the fifth iteration of the Hypertext Markup Language. It gives users access to a standardized markup language for organizing and displaying material on the internet. Web page functionality and interaction are improved because to new features and improvements brought by HTML5.

**HOW HTML5 WORK IN WEB DEVELOPMEMNT**

Web development relies heavily on HTML5, which gives web pages their structure and content. How HTML5 functions in web development is as follows:

**Markup Language:** The markup language syntax used by HTML5 surrounds the content with opening and closing tags. The p> tags, for instance, are used to define a paragraph: <p> it’s a paragraph </p>. Attributes for tags can also offer further functionality or information. The <img> tag, for instance, specifies an image and includes properties like <src> (source) and alt (alternative text).

**Structure:** The elements and tags provided by HTML5 establish the structure and content of a web page. These components are used by developers to arrange the various elements on a webpage, including headers, paragraphs, pictures, links, lists, tables, forms, and more. Each component has a distinct function and semantic meaning, which helps with accessibility and search engine optimization.

**Browser Rendering:** When a web page is loaded in a browser, the rendering engine of the browser interprets the HTML5 code. After reading the HTML markup, the browser creates a Document Object Model (DOM) that represents the page's structure. The components and content of the page may be interacted with and changed by JavaScript and CSS thanks to the DOM, which has a tree-like structure.

**Multimedia and APIs:** The <audio> and <video> elements in HTML5 provide native support for embedding multimedia material, such as audio and video, directly in web pages. Additionally, it includes a number of APIs (Application Programming Interfaces) that give site designers access to device features and functionalities. Among these APIs are the Geolocation API, the Canvas API for creating graphics, the Web Storage API for storing local data, the Web Workers API for doing background operations, and many more.

**Interaction and Behavior:** HTML5 provides built-in support for interactivity and dynamic behavior through JavaScript. JavaScript, a programming language, allows developers to manipulate the DOM, handle user events, make AJAX requests, perform client-side form validation, create animations, and much more. The integration of HTML5, CSS, and JavaScript enables the development of interactive and responsive web applications.

**Cross-Platform Compatibility:** HTML5 is made to function on a variety of hardware and software, including computers, laptops, smartphones, tablets, and even smart TVs. Support for HTML5 standards has been added by modern web browsers, ensuring uniform rendering and functioning across devices. HTML5 is a great option for creating online apps that are accessible to a large audience because of its cross-platform interoperability.

**CSS**

CSS stands for cascading style sheet. It is a simple design language intended to simplify the process of making web pages presentable. It separates the presentation layer from the content layer, allowing developers to define the visual aspects of a website independently of its structure.

**HOW CSS WORK IN WEB DEVELOPMENT**

The following are some of the ways in which CSS work in web development:

**Selectors:** CSS selectors identify the HTML elements to which the styles should be applied. Selectors may be based on the names of the elements, the class names, the IDs, the attributes, or the connections between the elements. You may choose to target certain elements, such as all paragraphs (p), elements with a particular class (.class-name), or elements with a particular ID (#element-id).

**CSS Syntax:** CSS has a straightforward syntax made up of selectors and declarations. While declarations specify the styles to be applied to such components, selectors specifically target HTML elements. A declaration is made up of a property and a value.

**Specificity and Cascading**: CSS employs a cascading order, allowing for the application of numerous CSS rules to the same element. Specificity in these situations decides which styles are dominant. The mix of selectors used determines how particular a rule is, and more specific rules take precedence over less specific ones. Following IDs, classes, and element selectors in order of specificity are inline styles.

**Linking CSS to HTML:** The CSS file must be linked to the HTML file in order to apply CSS styles to an HTML document. Usually, the HTML< head> section's <link> tag is used for this. As an alternative, you can also incorporate CSS straight into an HTML file by using inline styles.

**Box Model:** A fundamental idea in CSS, the box model describes how items are displayed on a web page. There are four elements: content, padding, border, and margin. These box model attributes may be changed to alter the size, spacing, and placement of components.

**JAVASCRIPT**

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

**HOW JAVASCRIPT WORK IN WEB DEVELOPMENT**

Here are some of the ways in which JavaScript work in web development:

Event handling: Using JavaScript's event handling feature, you may react to various events including button clicks, form submissions, mouse movements, and keyboard input. HTML components can have event handlers attached to them, which are programs that run in response to a predefined event. Event handlers let you build interactivity and carry out tasks based on user interactions, such verifying form inputs, showing notifications, or dynamically modifying information.

Client-Side Execution: JavaScript is a client-side scripting language, which means that the user's browser on their device actually executes it. The browser parses and interprets the JavaScript code when a web page with JavaScript is loaded. It interacts with the components of the web page and the browser APIs while running the code sequentially and carrying out various activities.

Server-Side Development: JavaScript isn't just for client-side programming. You may run JavaScript code on the server using Node.js, a JavaScript runtime built on Chrome's V8 engine. By using the same language (JavaScript) for an application's client and server sides, full-stack development becomes possible.

**JQUERY**

jQuery is a lightweight JavaScript library which is blazing fast and concise. This library was created by John Resig in 2006 and jQuery has been designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax.

**HOW JQUERY WORK IN WEB DEVELOPMENT**

Here are some of the ways in which jQuery work in web development:

**Effects and Animation:** jQuery makes it easier to add animations and visual effects to web pages. It offers ways to change the appearance of elements, slide elements, animate CSS attributes, fade elements in and out, and more. Without using difficult CSS or JavaScript code, you may develop interesting and interactive interfaces by using JQuery's animation and effects capabilities.

**Event Handling**: jQuery simplifies event handling by providing methods to attach event listeners to HTML elements. You can easily respond to events such as clicks, mouse movements, form submissions, and keyboard interactions. jQuery's event handling methods help you write concise and cross-browser-compatible code for event-driven interactions.

**Controlling the DOM:** JQuery offers a variety of ways to control the DOM. You may update an element's content, append or prepend elements, add or delete classes, adjust element properties, and more. These techniques make it easier to update content, change styles, or change the layout of a web page.

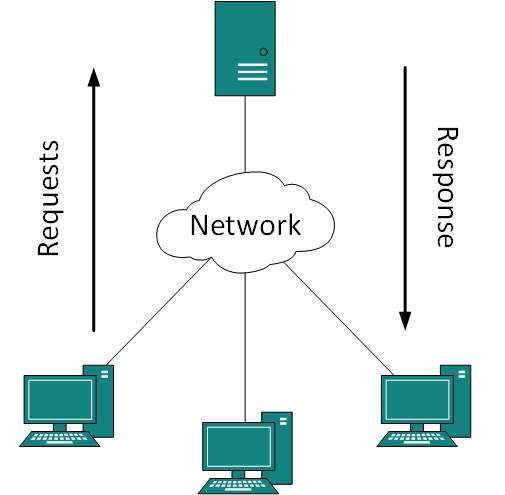
**SERVER AND CLIENT-SIDE INTERACTION**

**Server-side:** The term "server-side" describes operations that take place on the web server, which hosts the website or online application. These procedures, which include duties like obtaining information from a database, producing a web page, and managing user input, are normally carried out by the server before the website or web application is sent to the user's device.

**Client-side:** it refers to operations that take place on the user's computer, usually in their web browser. After the website or web application has been downloaded to the user's device, these operations are carried out, and they may involve activities like rendering and displaying a web page, managing user interactions, or performing JavaScript code.

**Server and client interaction:** refers to the communication and exchange of information between a client and a server in a network.in this concept, the clients make requests to servers, and servers respond with the requested data or perform requested actions.

The image below shows client and server interaction



In this diagram:

The "Client" represents the user's device.

The "Server" represents the computer or network of computers that handle client requests and provide responses.

The arrows indicate the flow of the request and response between the client and the server.

The client initiates the interaction by sending a request to the server.

The server receives the request, processes it, and generates an appropriate response.

The server sends the response back to the client.

The client receives the response and can take further actions based on it.