Unit: 3 Web Technology II

Introduction:

Web technology is the mechanism, which enables two or more computing devices to communicate over a network (mainly the internet). Web technologies are the various tools and technologies that are utilized in the process of communication between different types of devices over internet.

Web technologies include:

- Web browsers
- Programming/ Scripting languages and frameworks used in development of websites.
- Database that are used in backend for collection of data of websites.
- Protocols (Rules for communication on websites or networks).
- Graphic, audio, video, and other multimedia elements.

Scripting Language:

A scripting language is a type of programming language that is designed for the development and execution of scripts, which are sequences of instructions that are interpreted or executed by a computer. Unlike traditional programming languages, scripting languages are often interpreted rather than compiled. This means that the source code is executed directly by an interpreter without the need for a separate compilation step.

Scripting languages are typically used for tasks such as automating repetitive processes, controlling software applications, or implementing specific functionality within a larger program.

Definition: *The scripting language is like programming language that is used to create dynamic and automate processing websites or applications.*

Some examples of scripting languages include Python, JavaScript, Ruby, Perl, and Bash. These languages are commonly used in various domains, such as web development, system administration, and automation.

Types of Scripting Language:

1. Client-side Scripting Language
2. Server-side Scripting Language

Client-side Scripting Language:

A client-side scripting language is a type of scripting language that primarily runs on the client-side, or the user's web browser, rather than on the server. It is used to enhance the functionality of websites by allowing the creation of dynamic and responsive user interfaces. Client-side scripting is essential for creating dynamic web pages and providing a more involving user experience. Basically, these kinds of scripting code are placed on HTML documents. The effective client-side scripting language reduce the workload of sever. Client-side scripting is especially important for tasks like form validation and dynamic content updates. It allows developers to create more responsive and interactive web applications without requiring continuous communication with the server for every user action.

Common client-side scripting languages include HTML, CSS, and JavaScript.

Server-side Scripting Language:

A server-side scripting language is a type of programming language that runs on a web server rather than on the client's browser. It is used to handle tasks on the server, generate dynamic web pages, process data, interact with databases, and perform various server-side operations. Server-side scripting is responsible for managing the server and handling the backend logic of web applications.

Server-side scripting languages are essential for implementing server logic, handling user authentication, managing databases, and performing other server-related tasks in web development. They work as conjunction of client-side scripting and backend to create dynamic and interactive web applications.

Common server-side scripting languages include PHP (Hypertext Preprocessor), Python, Node.js (JavaScript on the server), Ruby etc.

Different Between Server-side and Client-side Scripting languages.

Server-side Scripting Language	**Client-side Scripting languages.**
1. Server-side scripting is used at the backend, where the source code is hidden to the client-side (browser).	1. Client-side scripting is used at the font end which user can see from the browser.
2. Its primary function is to manipulate and provide access to the respective database as per the request.	2. Its main function is to provide the requested output to the end-user.
3. It runs on web server.	3. It runs on the user's computer.
4. When a server-side script is processed, it communicates to the server.	4. Client-side scripting does not need any server interaction.
5. Any server-side technology can be used, and it does not depend on the client.	5. It usually depends on the web browser and its version used in client.
6. It is useful in customizing the web pages and implementing dynamic changes in the websites.	6. It is used in minimized the load on the server
7. It is more secure than client-side scripting because server-side script is hidden from user.	7. It is not secured compared to server-side because client-side script is visible to the user.
8. Server-side language involves programming such as PHP, ASP.NET, Python, C#, JSP etc.	8. Client-side scripting language involves language such as HTML, CSS, and JavaScript.