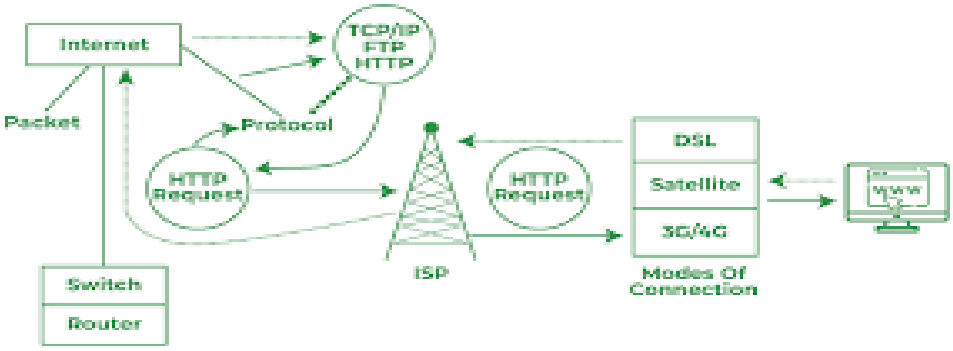
ASSIGNMENT_01

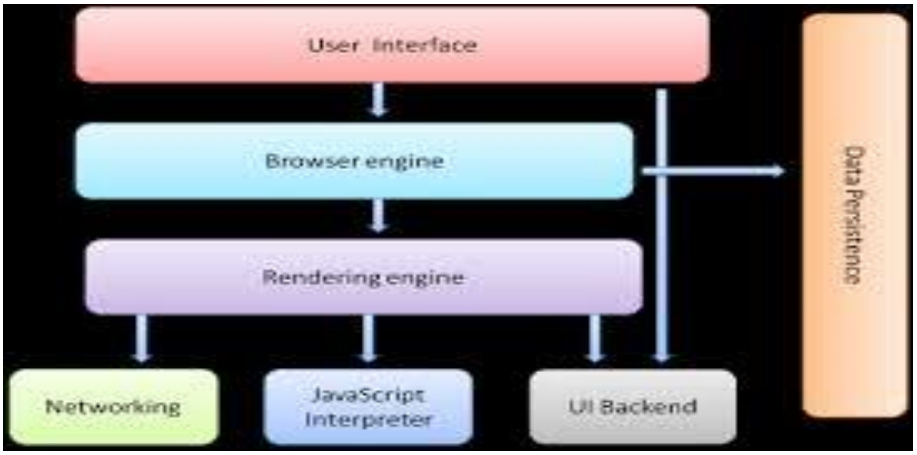
1.How internet works?
The internet is a global network of interconnected computers that communicate via standardized protocols such as TCP/IP. Information is exchanged in the form of packets routed through various networks until they reach their destination.

The Internet is a communication medium that consists of the help of a Network. A network is none other than the reserve of devices where millions of computers and other electronic devices get connected. It uses Transmission Control Protocols to exchange data with other devices.



2.How browser works?
A browser fetches and displays web pages. It communicates with web servers using protocols like HTTP or HTTPS to request resources like HTML, CSS, JavaScript, images, etc. It parses HTML to construct the Document Object Model (DOM), renders CSS to apply styles, executes JavaScript for interactivity, and displays the final web page to the user.

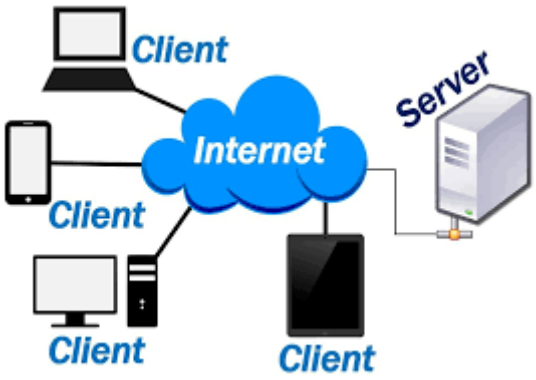
A web browser takes you anywhere on the internet. It retrieves information from other parts of the web and displays it on your desktop or mobile device. The information is transferred using the Hypertext Transfer Protocol, which defines how text, images and video are transmitted on the web.



3.What is Server?

A server is a computer or software that provides functionality to other programs or devices, known as clients. In the context of the web, a server hosts and serves web pages, applications, or data to users upon request.
A server is a computer program or device that provides a service to another computer program and its user, also known as the client. In a data center, the physical computer that a server program runs on is also frequently referred to as a server. That machine might be a dedicated server or it might be used for other purposes.

In the client/server programming model, a server program awaits and fulfills requests from client programs, which might be running in the same, or other computers. A given application in a computer might function as a client with requests for services from other programs and as a server of requests from other programs.



4.What are the types of server available?
Types of Servers and Their Applications
1. Application Server
These servers host web apps (computer programs that run inside a web browser) allowing users in the network to run and use them preventing the installation of a copy on their own computers. These servers need not be part of the World Wide Web. Their clients are computers with a web browser.
2. Catalog Server
These servers maintain an index or table of contents of information that can be found across a large distributed network. Distributed networks may include computers, users, files shared on

file servers, and web apps. Examples of catalog servers are directory servers and name servers. Their clients are any computer program that needs to find something on the network. An example can be a domain member attempting to log in, an email client looking for an email address, or a user looking for a file

3. Communication Server

These servers maintain an environment needed for one communication endpoint to find other endpoints and then communicate with them. These servers may or may not include a directory of communication endpoints and a presence detection service, depending on the openness and security parameters of the network. Their clients are communication endpoints.

4. Computing Server

These servers share vast amounts of computing resources which include CPU and random-access memory over a network. Any computer program that needs more CPU power and RAM than a personal computer can probably afford can use these types of servers. The client must be a networked computer to implement the client–server model which is a necessity.

5. Database Server

These servers maintain and share any form of database over a network. A database is an organized collection of data with predefined properties that may be displayed in a table. Clients of these servers are spreadsheets, accounting software, asset management software, or virtually any computer program that consumes well-organized data, especially in large volumes.

6. Fax Server

These servers share one or more fax machines over a network which eliminates the hassle of physical access. Any fax sender or recipient is the client of these servers.

7. File Server

Shares files and folders, storage space to hold files and folders, or both, over a network. Networked computers are the intended clients, even though local programs can be clients.

8. Game Server

These servers enable several computers or gaming devices to play multiplayer games. Personal computers or gaming consoles are their clients.

9. Mail Server

These servers make email communication possible in the same way as a post office makes snail mail communication possible. Clients of these servers are senders and recipients of email.

10. Print Server

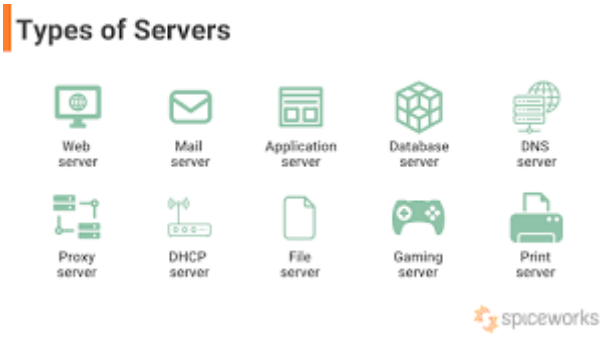
These servers share one or more printers over a network which eliminates the hassle of physical access. Their clients are computers in need of printing something.

11. Proxy Server

This server acts as an intermediary between a client and a server accepting incoming traffic from the client and sending it to the server. Reasons to use a proxy server include content control and filtering, improving traffic performance, preventing unauthorized network access, simply routing the traffic over a large and complex network. Their clients are any networked computer.

12. Web Server

These servers host web pages. A web server is responsible for making the World Wide Web possible. Each website has one or more web servers. Their clients are computers with a web browser.



5.What is Accessibility?
Accessibility refers to designing products, devices, services, or environments that are accessible to people with disabilities. In web design, accessibility ensures that websites and web applications are usable by everyone, including those with visual, auditory, motor, or cognitive impairments.

6.What is Markup Language?
Markup languages are computer languages that are used to structure, format, or define relationships between different parts of text documents with the help of symbols or tags inserted in the document. These languages are more readable than usual programming languages with strict syntax. There are several markup languages available but the most popular among them are as follows.

HTML
XML
XHTML
SGML

7.What is HTML?
HTML is a markup language used to create and structure web pages. It defines the structure and content of a web page using a variety of tags and attributes.

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

8.What is browser engine?
A browser engine (layout engine) is a software component that renders web content such as HTML, CSS, and images. Examples include Blink (used in Chrome, Edge), Gecko (used in Firefox), and WebKit (used in Safari).

9.What is rendering engine? Share the available rendering engines.
A rendering engine is part of a browser or application that interprets markup languages and stylesheets to present visual content. Examples include:

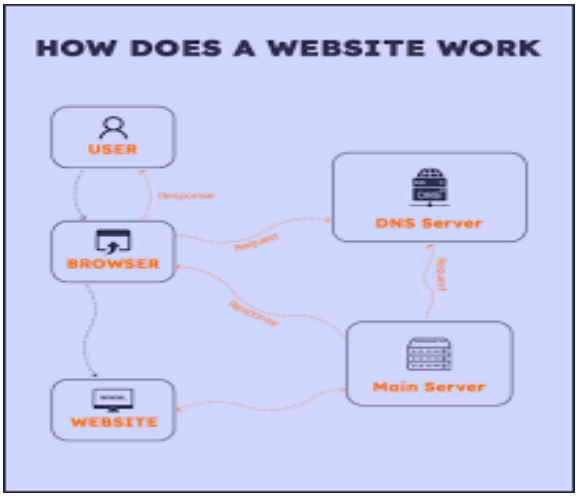
Blink (Chrome, Edge)
Gecko (Firefox)
WebKit (Safari)
Trident (older versions of Internet Explorer)

10.What is JavaScript Engine? Share the available JS engines. Purpose of JS Engine?
A JavaScript engine executes JavaScript code. Examples include:

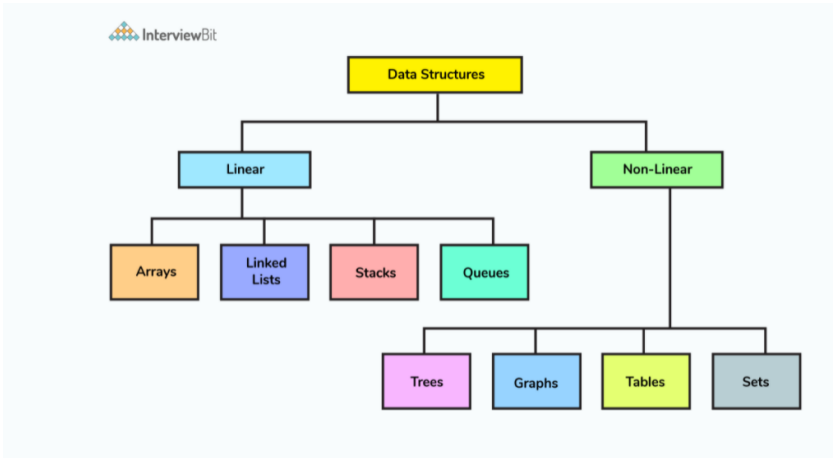
V8 (used in Chrome, Node.js)
SpiderMonkey (used in Firefox)
JavaScriptCore (used in Safari)
Chakra (used in Microsoft Edge, deprecated)
The purpose of a JS engine is to parse and execute JavaScript code within web browsers or server-side environments like Node.js.

11.How website works?

Websites work by browsers sending requests to servers for web pages, which are then delivered to the browser as HTML, CSS, JavaScript, images, etc. The browser renders these resources to display the web page to the user.



12.What is Data Structure?
A data structure is a way of organizing and storing data in a computer so that it can be accessed and manipulated efficiently. Examples include arrays, linked lists, trees, graphs, etc.



13.Explain Tree Data Structure?
A tree data structure consists of nodes connected by edges where each node has zero or more children nodes. It has a hierarchical structure with a root node at the top and branches leading to leaf nodes at the bottom.

14.What is user agent? Share the list and its purpose?
A user agent is a string of text sent by browsers to websites to identify themselves. Examples include:

Chrome: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Firefox: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:89.0) Gecko/20100101 Firefox/89.0

Safari: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/15.1 Safari/605.1.15
Its purpose is to inform servers about the user's operating system, browser version, and capabilities.

15.What is Hypertest?
It seems there might be a typo here. If you meant "Hypertext," it refers to text displayed on a computer or other electronic device with references (hyperlinks) to other text that the reader can immediately access.

16.What is HTML Tags?
HTML tags are keywords enclosed in angle brackets (<>) that define the structure and content of web pages. They consist of an opening tag, content, and a closing tag (e.g., Content).

17.What is HTML Attributes?
HTML attributes provide additional information about HTML elements. They are specified within the start tag and can modify the element's behavior or appearance (e.g., Link).

18.What is HTML Elements?
HTML elements are components of an HTML document defined by tags that structure content on a web page. Examples include headings (<h1> to <h6>), paragraphs ( ), images (<img>), etc.

19.How do you convert elements to a tree?
Elements in an HTML document are converted to a tree-like structure known as the Document Object Model (DOM) during parsing. Each HTML element becomes a node, and their relationships form a hierarchical tree.

20.What is DOCTYPE?
DOCTYPE (Document Type Declaration) is an instruction to web browsers and validators about the version of HTML or XHTML used in a web document. It ensures the document is parsed correctly and rendered according to standards.

21.What are the ways we can save HTML file?
HTML files can be saved locally on a computer using various methods:

Save directly from a text editor (e.g., Notepad, VS Code).
Export from a web browser (e.g., right-click and "Save Page As...").
Download from a web server via FTP or other file transfer protocols.
22.What is charset? Why do we need to use this?
Charset (character encoding) specifies how characters are represented in a document. It ensures that text displays correctly across different platforms and languages. Common charsets include UTF-8, ISO-8859-1, etc.

23.What is meta data? What is the purpose of it?

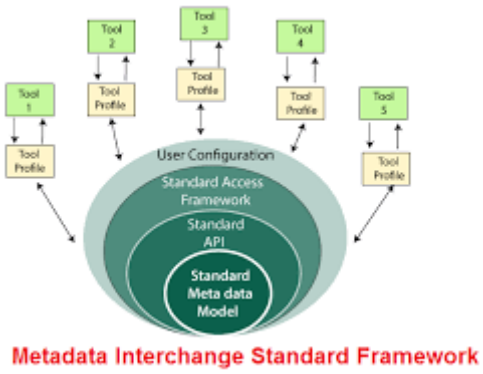
Metadata is data about the data or documentation about the information which is required by the users. In data warehousing, metadata is one of the essential aspects.

Metadata includes the following:

1. The location and descriptions of warehouse systems and components.
2. Names, definitions, structures, and content of data-warehouse and end-users views.
3. Identification of authoritative data sources.
4. Integration and transformation rules used to populate data.
5. Integration and transformation rules used to deliver information to end-user analytical tools.
6. Subscription information for information delivery to analysis subscribers.
7. Metrics used to analyze warehouses usage and performance.
8. Security authorizations, access control list, etc.

Metadata is used for building, maintaining, managing, and using the data warehouses. Metadata allow users access to help understand the content and find data.

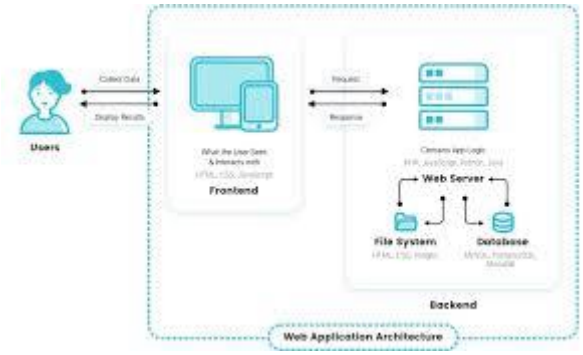
Metadata is data that describes other data. In HTML, <meta> tags provide metadata about the HTML document (e.g., character set, author, viewport settings). They are used by browsers, search engines, and other web services.



Metadata Interchange Standard Framework

24.Explain Web Application Architecture?
Web Application Architecture refers to the structure and layout of components that make up a web application. It typically includes:

Client-side: The interface and interactions seen by users (HTML, CSS, JavaScript).
Server-side: Backend logic and data storage (server, database, application server).
Networking: Protocols and communication channels between clients and servers (HTTP, WebSocket).
Security: Measures to protect data and prevent unauthorized access (SSL/TLS, authentication).



25. Explain Web Application Architecture?

Web application architecture is a mechanism that gives us a clarification that how the connection is established between the client and the server. It determines how the components in an application communicate with each other. It doesn't matter what is the size and the complexity level of the application is, they all follow the same principle only the details may differ.

In technical terms, when a user makes a request on a website, various components of the applications, user interfaces, middleware systems, databases, servers, and the browser interact with each other. Web Application Architecture is a framework that ties up this relation together and maintains the interaction between these components.

When a user interacts with a website and gets the response back from the server's end, the whole process executes within a few seconds. The most important thing we need to notice here is the code which has been passed to the browser. This code may or may not have particular instructions telling the browser how to respond with respect to the different types of user inputs. That's why a web application architecture includes all the sub-components and external applications interchanges for an entire software application. A web application architecture has to deal with the reliability, scalability, security, and robustness due to a large amount of global network traffic.