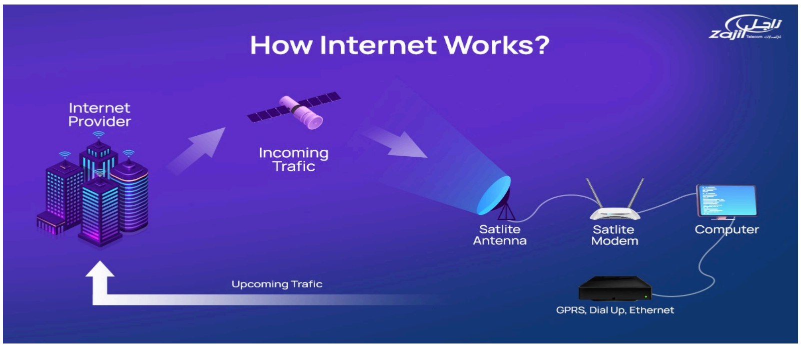
ASSIGNMENT 1
WEB TECHNOLOGIES

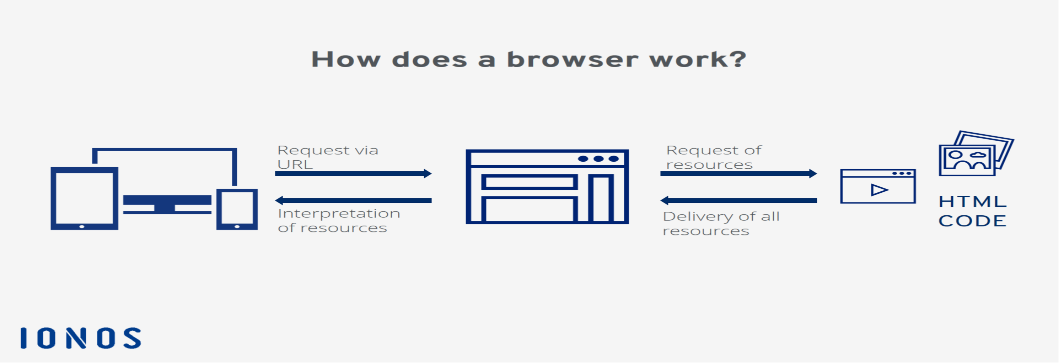
1.How internet works?

The Internet is a global network of computers connected together. It allows these computers (and other devices like smartphones and tablets) to communicate and share information. This happens through a system of routers and servers that help direct and manage data traffic. When you access a website or send an email, your device sends requests over the Internet to servers, which respond by sending back the requested information. This communication relies on protocols that ensure data is transmitted reliably and securely across the network.



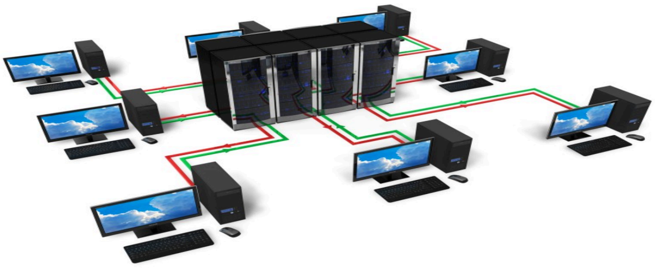
2.How browser works?

A browser is a software application that allows you to access and view websites on the internet. It interprets the code (HTML, CSS, JavaScript) that makes up web pages and displays them on your screen. When you type a web address or click a link, the browser sends a request to a server, which sends back the web page data. The browser then processes this data to show you the web page with text, images, and interactive elements like buttons and forms.



3.What is Server?

A server is a specialized computer or software that provides services or resources to other computers or programs over a network. It stores data, hosts websites, manages emails, or performs other tasks requested by client devices. Servers enable communication and sharing of information across networks like the Internet.



4.what are the types of server available?
Web Server: Delivers web pages and content over the internet. Examples include Apache, Nginx.
File Server: Stores and manages files accessible over a network. Examples include Windows File Server, Samba.
Database Server: Manages databases and provides database services. Examples include MySQL, PostgreSQL.
Mail Server: Manages email communication. Examples include Microsoft Exchange, Postfix.
Application Server: Hosts and manages applications. Examples include Apache Tomcat, Microsoft IIS.

5.What is SEO? Importance of SEO?
SEO stands for Search Engine Optimization. It refers to the practice of optimizing websites and web pages to improve their visibility and ranking in search engine results pages (SERPs).

Importance of SEO:

1. **Increased Visibility**: SEO helps your website appear higher in search engine results, making it more likely that users will click on your link.
2. **More Traffic**: Higher rankings lead to more organic traffic to your website. Users are more likely to visit websites that appear on the first page of search results.
3. **Cost-Effective**: Compared to paid advertising, SEO can be a cost-effective strategy for long-term online visibility and traffic generation.
4. **Better User Experience**: SEO involves optimizing website speed, mobile-friendliness, and usability, which enhances the overall user experience.
5. **Builds Credibility and Trust**: Websites that rank higher in search results are often perceived as more credible and trustworthy by users.

6. What is Accessibility?
Accessibility refers to the design and creation of environments, products, and services that can be used by all people, including those with disabilities. The goal is to ensure that everyone, regardless of their physical, sensory, cognitive, or other limitations, can access, understand, and use these environments, products, and services effectively.

7. What is Markup Language?

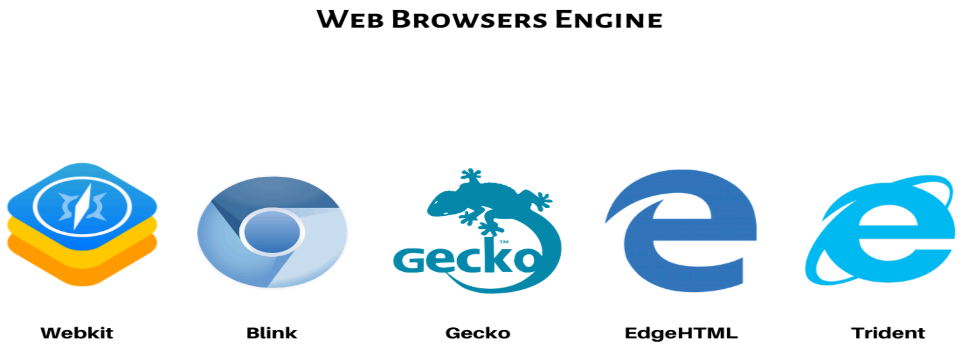
A markup language is a system for annotating a document in a way that is syntactically distinguishable from the text. It is used to define the structure, presentation, and semantics of the document's content. Markup languages use a set of tags or symbols placed within the text to specify how the text should be formatted, displayed, or interpreted.

8. What is HTML?

HTML (HyperText Markup Language) is the standard markup language used to create and design documents that are displayed on the web. HTML provides the basic structure of web pages, which is enhanced and modified by other technologies like CSS (Cascading Style Sheets) and JavaScript.

9. What is browser engine?

A browser engine, also known as a layout engine or rendering engine, is a core software component of a web browser that is responsible for displaying web content. It interprets HTML, CSS, JavaScript, and other web technologies to render the web pages and applications you see and interact with.



10. What is rendering engine? share the available rendering engine?

A rendering engine, also known as a layout engine or browser engine, is a core component of a web browser responsible for interpreting web code (HTML, CSS, JavaScript, etc.) and rendering the resulting content on the user's screen.

11. What is JavaScript Engine? share the available JS engine? Purpose of JS Engine?

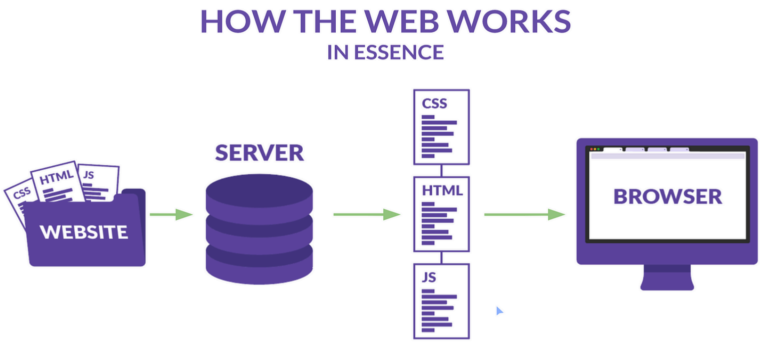
A JavaScript engine is a specialized software component embedded in web browsers and other runtime environments, designed to interpret and execute JavaScript code. It is crucial for enabling the dynamic and interactive features of modern web applications.

Purpose of a JavaScript Engine

1. **Execution of JavaScript Code**: The primary function of a JavaScript engine is to execute JavaScript code, transforming it from human-readable syntax into machine code that can be processed by the computer.
2. **Just-In-Time (JIT) Compilation**: Modern JavaScript engines use JIT compilation techniques to optimize code execution.
3. **Garbage Collection**: JavaScript engines manage memory automatically through garbage collection, which identifies and frees up memory that is no longer in use, helping to prevent memory leaks.
4. **APIs and Runtime Environment**: JavaScript engines provide APIs and a runtime environment that support various functionalities such as manipulation of the DOM, handling events, and making network requests.
5. **Optimization**: Engines perform various optimizations to enhance the performance of JavaScript code, including inlining functions, optimizing loops, and reducing overhead from frequently called functions.

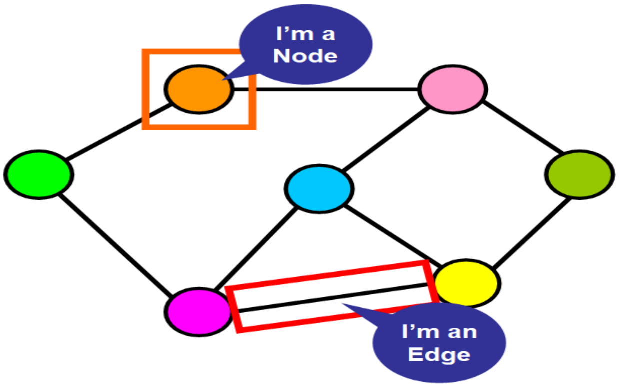
12. How website works?

The process from the moment a user requests a web page to when that page is displayed in their browser.



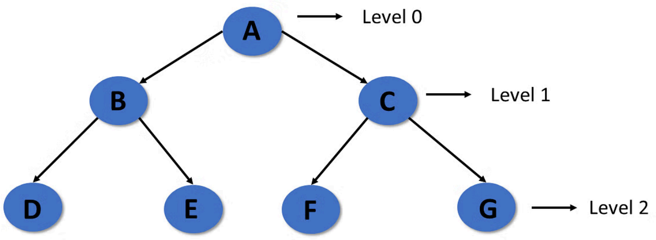
13. What is Data Structure?

A data structure is a specialized format for organizing, processing, storing, and retrieving data. Data structures are fundamental to computer science and software engineering because they provide efficient ways to manage data, which is crucial for designing efficient algorithms and systems.



14. Explain Tree Data Structure?

A tree is a widely used data structure in computer science that simulates a hierarchical tree structure with a set of connected nodes. It is a non-linear data structure, meaning its elements are not arranged sequentially but rather in a hierarchical manner.



15. What is user agent? share the list and its purpose?

A user agent is a software application that acts on behalf of a user to retrieve and present information from the internet. In the context of web browsing, a user agent is typically a web browser, but it can also refer to any software that interacts with web servers, including web crawlers, mobile apps, and other automated tools.

16. What is Hypertest?

Hypertext refers to text that contains links to other texts. It enables non-linear navigation of information, allowing users to interact with text beyond the linear reading of a physical document. The concept was pivotal in the development of the World Wide Web.

17. What is HTML Tags?

HTML tags are the building blocks of web pages. They are used to define the structure and content of a web document by marking different elements. HTML (Hypertext Markup Language) tags are enclosed in angle brackets `< >` and typically come in pairs: an opening tag and a closing tag. The opening tag denotes the beginning of an element, and the closing tag marks the end.

For example, `<p>` is an opening tag for a paragraph, and `</p>` is the closing tag. Everything between these tags forms the paragraph element on a web page.

HTML tags can also have attributes that provide additional information about the element, such as `class`, `id`, `style`, etc. These attributes help define the appearance, behavior, or functionality of the element.

Overall, HTML tags are fundamental for creating structured and semantic web pages, allowing browsers to interpret and display content correctly according to their defined elements and attributes.

18. What is HTML Attributes?
HTML attributes provide additional information about HTML elements. They are used within the opening tag of an element and are typically written as name-value pairs. Attributes modify the behavior or appearance of an element or provide metadata about the element.
Eg:-
<a href="https://example.com">Visit Example</a>

19. What is HTML Elements?
HTML elements are the fundamental building blocks of HTML documents, used to define the structure and content of web pages. An HTML element typically consists of a start tag, content, and an end tag (if it's not a self-closing tag). Elements can also include attributes that provide additional information about the element.

20. How do convert elements to tree?
Converting HTML elements into a tree structure is fundamental to understanding how web pages are structured and how browsers interpret and render them. This process involves visualizing the hierarchy and relationships between nested HTML elements. Here's a step-by-step outline of how HTML elements are converted into a tree:

1. **Start with the HTML Document**:
   - The entire HTML document is considered the root of the tree. It represents the top-level structure of the web page.
2. **Identify HTML Elements**:
   - Each HTML element, such as `<html>`, `<head>`, `<body>`, `<div>`, `<p>`, `<a>`, etc., is represented as a node in the tree.

21. What is DOCTYPE?
DOCTYPE (Document Type Declaration) is an instruction or a directive in HTML that informs the web browser about the version of HTML or XHTML in which the web page is written. It is placed at the very beginning of an HTML document, before the `<html>` tag, and helps the browser render the web page correctly by specifying the rules and standards that the HTML document follows.

The DOCTYPE declaration is not an HTML tag; rather, it is a standalone instruction. Its primary purposes include:

1. **Document Type Definition**: It defines which version of HTML or XHTML the document adheres to, specifying the rules and syntax that the browser should use to parse and render the content.
2. **Quirks vs Standards Mode**: In older browsers, the presence or absence of a valid DOCTYPE declaration can determine whether the browser renders the page in quirks mode (emulating older, non-standard behavior) or standards mode (following the specified HTML standards).
3. **Validation**: It allows validators to check whether the HTML code conforms to the specified version of HTML or XHTML. Validators use the information provided by the DOCTYPE to ensure that the document structure and syntax are correct.

22. What are the ways we can save html file?
Text Editor or IDE:

- **Save As**: If you're editing an HTML file in a text editor (like Sublime Text, Visual Studio Code, Notepad++, etc.) or an Integrated Development Environment (IDE) (like JetBrains IntelliJ IDEA, Eclipse, etc.), you can simply use the "Save As" option from the File menu. This allows you to specify the file name, location, and ensure it is saved with a `.html` extension.

Browser:

- **Save Page As**: You can save an HTML file from a web browser by right-clicking anywhere on the page and selecting "Save Page As" or "Save As". This option lets you save the entire HTML document along with associated resources (like images, CSS files) to your local machine.

Command Line:
If you're working in a command-line interface (CLI) or terminal, you can use commands like `curl` or `wget` to download HTML files from a web server to your local directory. For example:
bash
Copy code

```
curl -o index.html https://example.com
```

- This command downloads the HTML file from `https://example.com` and saves it as `index.html` in the current directory.

Content Management Systems (CMS):

- If you're using a CMS (like WordPress, Joomla, etc.), there are built-in options to export or save web pages as HTML files. This is useful for backing up content or transferring it to another platform.

Browser Developer Tools:

- In modern browsers (Chrome, Firefox, Edge, etc.), you can use the Developer Tools to inspect elements of a web page. This includes viewing and saving the HTML source code directly from the "Elements" tab, often accessible by right-clicking on the page and choosing "Inspect" or pressing `Ctrl+Shift+I` (or `Cmd+Option+I` on Mac).

Online Tools and Services:

- There are various online tools and services that allow you to download web pages as HTML files. These can be useful for saving specific pages or content from the web without needing to navigate through a browser.

23. What is charset? why we need to use this?

Charset (character set) refers to the encoding scheme that maps characters to numeric codes. In the context of web development and HTML, charset specifically refers to the character encoding used to represent text on web pages.

Why do we need to use charset?

1. Text Representation: Different character encodings represent characters (letters, numbers, symbols) in different ways. For example, ASCII (American Standard Code for Information Interchange) encodes basic Latin characters, while UTF-8 (Unicode Transformation Format-8) supports a wider range of characters, including various languages and symbols.
2. Compatibility: Specifying a charset ensures that web browsers interpret and display text correctly according to the intended encoding. If the charset is not specified or is incorrectly set, browsers may display characters incorrectly or fail to render certain characters at all.
3. Multi-language Support: Modern websites often need to support multiple languages and character sets. UTF-8 is the most commonly used encoding because it supports a wide range of characters and is compatible with ASCII. It allows web developers to create web pages that can display content in various languages and scripts seamlessly.
4. SEO and Accessibility: Using the correct charset is crucial for search engine optimization (SEO) and accessibility. Search engines and screen readers rely on proper encoding to index and interpret content accurately. Incorrect or missing charsets can impact SEO rankings and accessibility compliance.

24. What is meta data? what is the purpose of it?

Metadata refers to data that provides information about other data. In the context of web development and HTML, metadata refers to additional information about an HTML document that is not directly part of the content visible to users, but rather provides supplementary details for browsers, search engines, and other automated systems. The purpose of metadata is to describe the characteristics, properties, and attributes of the document itself.

Purpose of Metadata:

1. **SEO (Search Engine Optimization)**: Metadata helps search engines understand the content of a web page and index it appropriately. Metadata such as meta titles, meta descriptions, and meta keywords provide information that search engines use to display search results effectively.
2. **Browser Behavior**: Metadata can influence how web browsers render and display web pages. For example, specifying a character encoding (`<meta charset="UTF-8">`) ensures that browsers interpret and display text correctly according to the specified encoding.
3. **Social Media Sharing**: Metadata like Open Graph tags (`og:title`, `og:description`, `og:image`) and Twitter Card tags (`twitter:title`, `twitter:description`, `twitter:image`) provide structured data that social media platforms use when users share links. This ensures that shared content appears correctly formatted with appropriate titles, descriptions, and images.

4. **Accessibility**: Some metadata can enhance accessibility by providing information about the language of the document (`<meta http-equiv="Content-Language" content="en">`) or specifying accessibility features (`<meta name="viewport" content="width=device-width, initial-scale=1.0">` for responsive design).
5. **Analytics and Tracking**: Metadata can include tags for analytics services (`<meta name="google-site-verification" content="...">`) or tracking scripts (`<meta name="robots" content="index, follow">`) to monitor and analyze visitor behavior on a website.

25. Explain Web Application Architecture?

Web application architecture refers to the structure and organization of components that make up a web application and how they interact with each other to deliver functionality to users. It encompasses both the client-side (frontend) and server-side (backend) components, as well as the communication protocols and technologies used to facilitate interactions between these components. Here’s an overview of the typical components and layers involved in web application architecture:

Client-Side Components (Frontend):

1. **User Interface (UI)**:
   - The UI layer is what users interact with directly. It consists of HTML for structure, CSS for presentation and styling, and JavaScript for behavior and interactivity. Modern web applications often use frontend frameworks and libraries like React, Angular, or Vue.js to manage complex UI components and state.
2. **Client-Side Scripting**:
   - JavaScript is used extensively for client-side scripting to handle user interactions, perform form validations, make asynchronous requests (AJAX), and dynamically update the UI without reloading the entire page (SPA - Single Page Application).
3. **Browser**:
   - The web browser interprets and renders the HTML, CSS, and JavaScript to display the web application to users. It also manages user interactions, events, and data input.