REACT ASSIGNMENTS_1

1. what is emmet?

What is Emmet used for?

Emmet is a free add-on for your text editor. It **allows you to type shortcuts that are then expanded into full pieces of code**. By using Emmet, developers type less, they save both on keystrokes and time.

2. difference between library and framework?

a library is that code that is written by other (meta) developers for us to use.

In programming, a library is a collection of pre-written code or software components that can be reused by developers to solve common programming problems or add functionality to their programs.

Libraries typically contain functions, classes, and other reusable code components that can be imported or linked to a program. This can save developers time and effort by allowing them to reuse existing code rather than having to write everything from scratch.

multiple library together make a framework

3. what is CDN? why we use it?

CDN stands for Content Delivery Network. It is a distributed network of servers strategically located across different geographic locations worldwide. The primary purpose of a CDN is to deliver web content, such as images, CSS files, JavaScript files, and other static or dynamic assets, to end-users with improved performance, reliability, and scalability.

Here's why CDN is used and its benefits:

Faster Content Delivery: By leveraging a network of servers distributed globally, CDNs bring content closer to end-users. When a user requests a webpage, the content is served from the nearest CDN server, reducing the distance the data needs to travel. This proximity results in faster content delivery and reduced latency, leading to improved website performance and a better user experience.

4. What is React and why it is used?

React is a popular JavaScript library for building user interfaces. It was developed by Facebook and is now maintained by Facebook and a community of individual contributors. React is widely used in web development for creating interactive and dynamic UI components.

As react provides component based architecture so these components can be reused again and also if we need to refresh any component than we dont need to refresh whole webpage

5. What is Crossorigin in script tag?

When you include an external script in your HTML file using the `script` tag, the browser makes a request to the specified URL to fetch the script file. If the script is hosted on a different domain than the one serving the HTML file, the browser may restrict access to the script for security reasons, because it violates the same-origin policy.

To overcome this issue, you can use the `crossorigin` attribute to indicate that the script is safe to load from a different domain. The `crossorigin` attribute can take one of three values: `anonymous`, `use-credentials`, or an empty string.

- `anonymous`: This value indicates that the script can be loaded from a different domain, but the browser will not send any credentials (such as cookies) along with the request. This is the default value.
- `use-credentials`: This value indicates that the script can be loaded from a different domain, and the browser will send any credentials associated with the current origin with the request.

- An empty string: This value is equivalent to not using the `crossorigin` attribute at all, and is not recommended.

6. Difference between React and ReactDOM?

React is the core library for building user interfaces using a component-based approach, while ReactDOM is a package that focuses on rendering React components in the browser by interacting with the DOM. ReactDOM acts as the bridge between React and the browser environment, enabling the efficient updating and rendering of React components.

7. What is react.develoment.js and react.production.js file?

The `react.development.js` file is one of the JavaScript files provided by React for development purposes. It is a non-minified version of the React library that contains additional helpful error messages and warnings to assist developers during the development and debugging process.

The development version of React provides more descriptive error messages, warnings, and additional runtime checks to help developers catch common mistakes, identify improper usage of React APIs, and provide helpful suggestions for corrections. This can be valuable when writing and testing code, as it helps to pinpoint and address potential errors or incorrect usage early on.

The `react.production.js` file is a minified and optimized version of the React library specifically designed for production environments. It is used when deploying a React application to ensure optimal performance and minimize the file size.

The production version, `react.production.js`, is optimized for performance and file size. It has undergone minification, which removes unnecessary whitespace, comments, and simplifies the code structure. Minification helps to reduce the file size, resulting in faster downloads for users visiting the website.

Furthermore, the production version of React omits the additional error messages, warnings, and development-specific checks present in the development version. These

checks and warnings are helpful during development but are not necessary in a production environment where the focus is on performance and minimizing bandwidth usage.

8. What is Async and differ?

lets understand the what happens when we load the webpage

first part is html parsing

second part is loading the scripts

HTML parsing refers to the process of analyzing an HTML document's structure and extracting relevant information from it. HTML parsing is commonly performed in web development and data extraction tasks. It involves breaking down the HTML code into its constituent elements, such as tags, attributes, and content, in order to access and manipulate specific parts of the document.

HTML parsing can be done using programming languages such as JavaScript, Python, or libraries like BeautifulSoup and lxml in Python, or built-in functions in JavaScript like `querySelector` or `getElementById`. These tools provide convenient methods to parse HTML and extract the required data efficiently.

now scripts can loadeded normally and async and defer

In JavaScript, there are different methods to load scripts into an HTML document. Two commonly used methods are:

first is synchornous ⇒ **<script src="script.js"></script>**

1. **Synchronous**: By default, when a script tag is encountered in the HTML document, the browser will pause parsing the HTML and execute the script immediately. This means that the rendering of the page will be blocked until the script is fully loaded and executed. The script is fetched and executed synchronously in the order they appear in the HTML document.

it means synchornous mein jab tak content fetech hoga using script tag us samay html parsing nhi hogi and jab script run hogi tab bhi html parsing stopped hogi

second is asynchornous ⇒ **<script src="script.js" async></script>**

in async scirpt fetch hogi internet se continusoly to html parsing but when fetching script completed and then html parsing is stopped and js code run and after runnnign the whole script than html parsing start again

third is defer ⇒ **<script src="script.js" defer></script>**

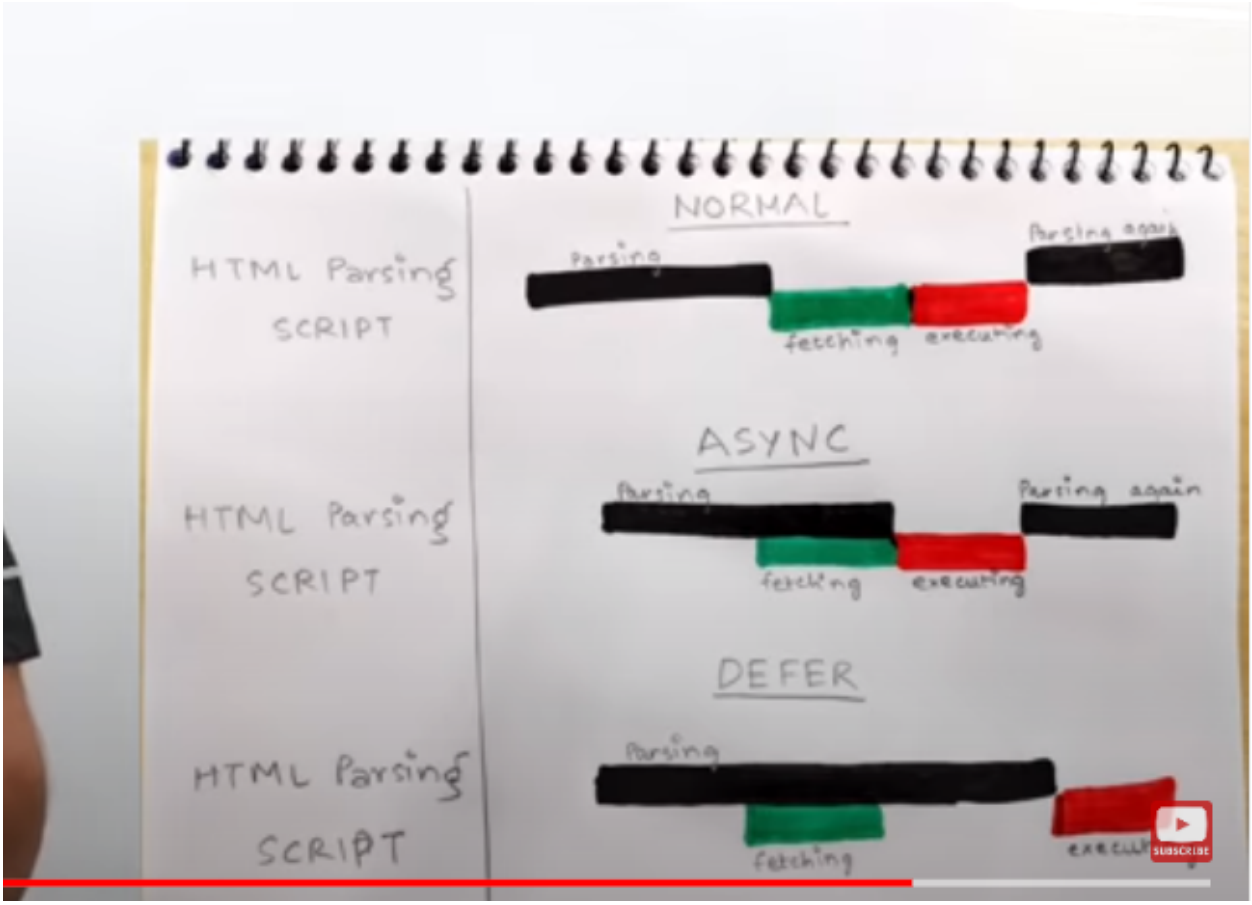
defer mein html parsing continuosly chalti rahegi and side by side script fetch hogi internet se but script run tabhi hogi jab html parsing complete ho jaygi

asycnc will not guarntee execution of code line by line like which need to load first etc

if in case if second line code is dependent on first line than we dont use async in that case because if secocnd line is executed first it may give error as it is depoendent on first line

in that case we use defer

defer guarntee line by line execution of code



loading of script is furthur involved two steps ⇒ one is loading the data from internet and third is actually running code line by line.