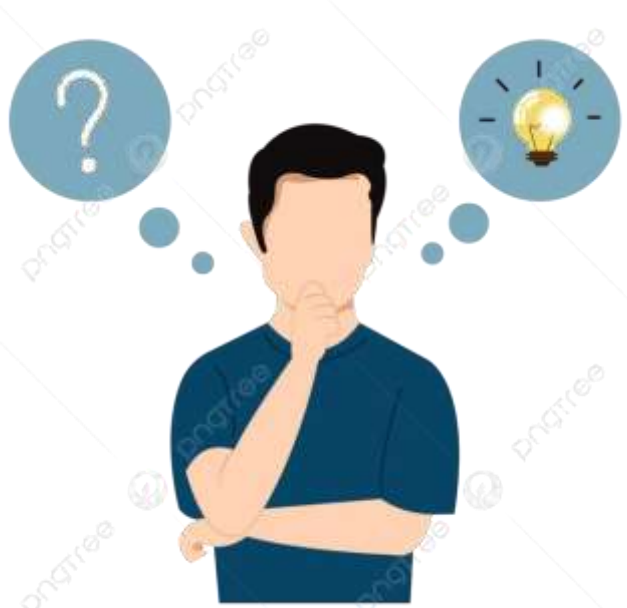
CHAPTER 10

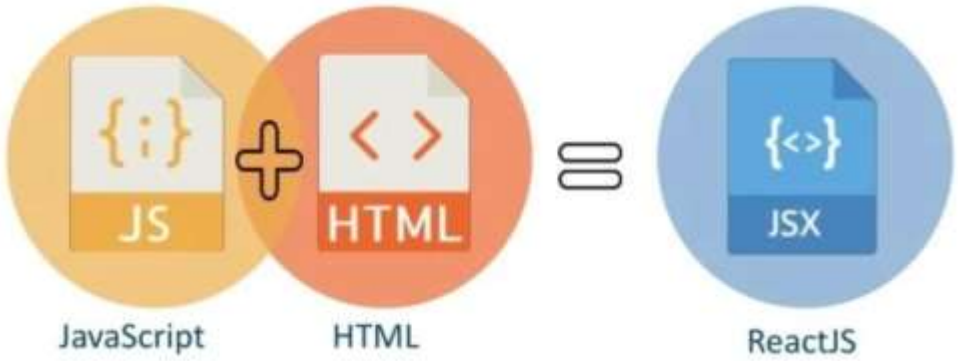
IMPLEMENTING SINGLE-PAGE APPLICATIONS

Introduction to React js

Q. What is React.js?

- React.js is a **Frontend Javascript library** used for building user interface (UI's).
- Specially react.js is used to create **single-page applications** (SPA).
- It allows developers to create reusable UI components, making the development process faster and more efficient.





Library vs Framework

Q. What is Library?

Library is a collection of pre-written code that you can call to perform specific tasks. It doesn't have syntax or rules to be followed. You can use them however you see fit but they don't provide you with everything, you might need to use other libraries.

Example - React JS, GSAP

Q. What is Framework?

Framework is a complete structure for building an application. It provides rules, patterns, and tools, and often dictates how your code should be structured. It is built over your libraries and there is a proper syntax and rules we have to follow. You can't do and use it however you see fit but it provides you with everything, no need of anything else.

Example - Tailwind, Angular

SPA vs MPA

Q. What is SPA?

- A SPA stands for **single-page application**.
- Loads a **single HTML page** and updates content dynamically without reloading the page.
- **Faster after the first load**, as only necessary data is updated.
- Uses **client-side routing** (React Router) to switch views without reloading.
- Examples: **Gmail, Facebook, Twitter, Netflix, YouTube Web App.**

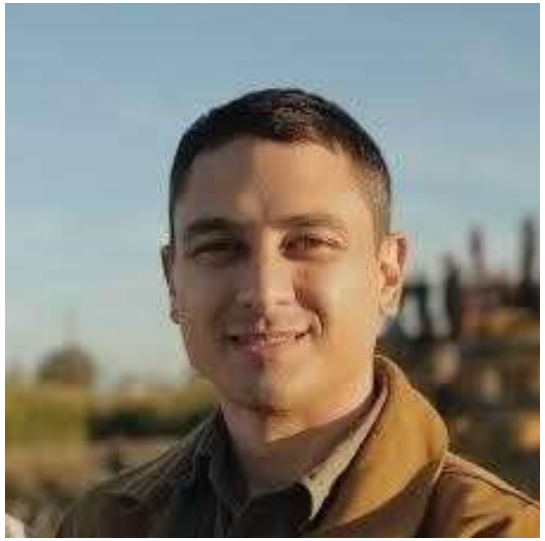
Q. What is MPA?

- A MPA stands for **Multi-page applications.**
- Each page is a **separate HTML file**, making navigation slower.
- **Slower performance**, as the browser loads everything again on each page change.
- Every new page request **reloads the entire page from the server.**
- Uses **server-side routing** (URLs trigger full-page reloads).
- Examples: **Amazon, Wikipedia, News Websites, Banking Portals.**

History to React js

Q. Who created React.js?

- The **SPA** is not a new term it present since **2002**. But we didn't have technology at that time.
- In **2009,** we got **angularjs** for creating SPA, but it was not optimized and had bugs.
- In year **2011 React JS**, was created by **Jordan Walke** while working as a software engineer at **Facebook**.
- It was first used in Facebook's News Feed and later in **Instagram**.
- It is was made and **open-source** library in 2013 to used by developers worldwide.
- As of March 2025, React.js has had a total of **19 major versions** released.



Jordan Walke

Characteristics of React js

- **React JS is a JS Library :**

React.js is a lightweight library created using **JavaScript.** It primarily focuses on the view layer of an application.

- **Free and Open Source :**

React is maintained by **Meta** (formerly Facebook) and is **open source**, allowing developers worldwide to contribute and improve it

- **Component-Based Architecture :**

A **component** in React splits your **UI into small, reusable pieces.** Each component handles its own **logic and rendering** , making your app **modular and maintainable** .

- **Declarative :**

Declarative means you tell React what you want, and React figures out how to do it. Instead of manually updating the DOM, you describe the UI state, and React updates it efficiently.

- **Unidirectional Data Flow :**

Unidirectional Data Flow in React means **data flows in one direction** - from parent component to child components via **props**. This ensures predictable data flow, making it easier to debug.

- **Learn once write anywhere :**

"Learn Once, Write Anywhere" in React Native means you can use your React knowledge to build apps for multiple platforms **(web, iOS, Android)** without learning separate technologies for each.

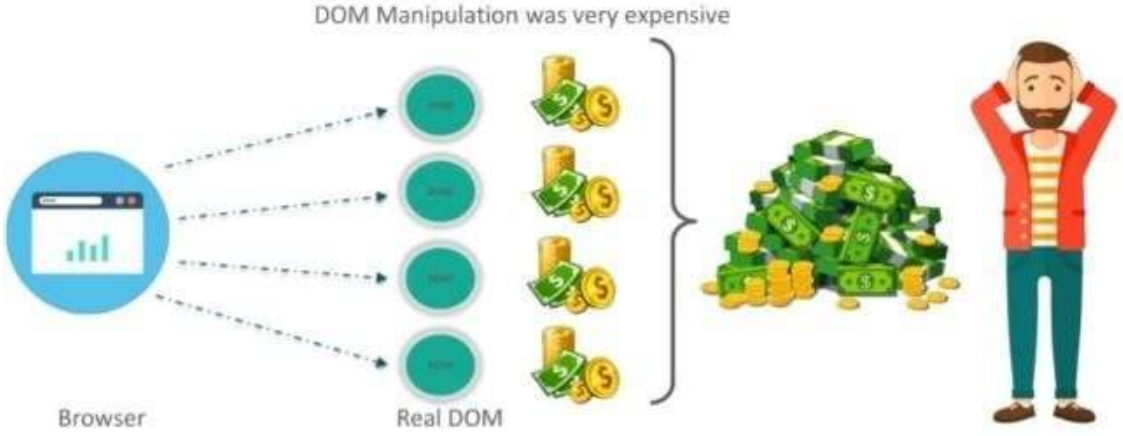
- **Virtual DOM:**

React uses a **Virtual DOM** to optimize updates. React does not update the real D O M directly because it's slow. Instead, it uses a Virtual DOM, which is a **lightweight copy of the real DOM.**

Without React.js vs With React.js

Q. Without react.js?

- In **traditional JavaScript**, every UI update directly modifies the **Real DOM**. The problem is that the DOM is **slow.**
- And DOM manipulation was very expensive.



Q. With react.js?

- React does not update the **Real DOM** directly.
- Instead of it uses **virtual dom.**
- Keeps UI fast and smooth, even in large applications.