ReactJS-HOL-1

Q/A

1. **Define SPA and its benefits**

   SPA (Single-Page Application) is a web application that loads a single HTML page and dynamically updates content without reloading the entire page.
   Some of the benefits include *Faster navigation and better user experience*, *Reduced server load*, *Seamless transitions between views*, and *Efficient data loading with APIs*.

2. **Define React and identify its working**

   React is a JavaScript library for building user interfaces, especially single-page applications.
   How it works:
   - Uses **components** to build UI
   - Utilizes a **virtual DOM** to efficiently update the UI
   - React compares the virtual DOM with the real DOM and applies only the necessary updates (diffing & reconciliation)

3. **Identify the differences between SPA and MPA**

   Differences between SPA and MPA (in points):
   - **SPA** loads a single HTML page and updates content dynamically without full page reloads.
   - **MPA** loads a new HTML page from the server for each navigation.
   - **SPA** offers faster, smoother user experience.
   - **MPA** is better for SEO and large content-heavy sites.
   - **SPA** uses client-side routing; **MPA** uses server-side routing.

4. **Explain Pros & Cons of Single-Page Application**

   Pros:

   - Smooth user experience
   - Fast interaction after first load
   - Easier to turn into a mobile app (via frameworks like React Native)

   Cons:
   - SEO challenges

- Initial load can be large
- Browser history and state management can be complex

5. **Explain about React**

   React is a **component-based** JavaScript library developed by Facebook for building **interactive UIs**. It is widely used for developing SPAs and supports server-side rendering as well.

6. **Define virtual DOM**

   The **virtual DOM** is an in-memory representation of the real DOM. React uses it to track changes efficiently and update only the parts of the DOM that changed, improving performance.

7. **Explain Features of React**

   - **Component-based architecture**
   - **JSX syntax** (JavaScript + XML)
   - **Virtual DOM** for efficient updates
   - **Unidirectional data flow**
   - **Reusable components**
   - **Hooks** for state and lifecycle management
   - **Strong ecosystem** with tools like Redux, React Router

Hands-on Experiment

Here I have created react app with *vite* as CRA is deprecated and not recommended by community and meta.

```jsx
// App.jsx
import './App.css'

export default function App() {
  return (
    <div style={{display: 'flex', justifyContent: 'center'}}>
      <h1>
        Welcome the first session of React
      </h1>
    </div>
  )
}
```

```js
// vite.config.js
import react from '@vitejs/plugin-react-swc'
import { defineConfig } from 'vite'
// https://vite.dev/config/

export default defineConfig({
  plugins: [react()],
  server: {
    port: 3000,
  },
})
```

Output

Welcome the first session of React