**1. ReactJS – HOL**

**1. Define SPA and Its Benefits**

**SPA (Single-Page Application)** is a web application that dynamically updates the content of a single web page without requiring a full page reload.

**Benefits of SPA:**

1. **Faster Load Times**: Only initial load is heavy; further interactions are fast.
2. **Seamless User Experience**: No page refresh, leading to a smoother navigation.
3. **Reduced Server Load**: After the first load, only data is fetched (not full HTML pages).
4. **Easy Debugging**: Tools like Chrome DevTools and React Developer Tools make debugging easier.
5. **Improved Performance**: Less bandwidth used, faster content rendering.

**2. Define React and Identify Its Working**

**React** is a **JavaScript library** developed by Facebook for building **user interfaces**, especially for **single-page applications**.

**How React Works:**

* It uses **components** to build UI.
* Each component has its own **state** and **props**.
* React utilizes a **virtual DOM** to efficiently update and render components.
* Changes in state trigger re-rendering of only affected parts (not the whole page).

**3. Identify the Differences Between SPA and MPA**

| **Feature** | **SPA (Single Page App)** | **MPA (Multi Page App)** |
| --- | --- | --- |
| Navigation | No page reload (client-side routing) | Each interaction reloads a new page |
| Speed | Faster after first load | Slower due to multiple full-page reloads |
| Development Complexity | Moderate with React, Vue, Angular | Simpler with plain HTML/CSS/JS |
| SEO | Difficult (unless SSR is used) | Easier as each page is separate |
| Backend Integration | Typically uses APIs (REST/GraphQL) | Server renders content and returns HTML |

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

**Pros:**

* Fast and responsive UI.
* Reduced bandwidth usage.
* Better user experience.
* Easier to convert into mobile apps with frameworks like React Native.

**Cons:**

* Poor SEO unless SSR (Server Side Rendering) is used.
* Initial load time may be higher.
* Harder to manage browser history and back button.
* Security risks if not implemented carefully (e.g., XSS).

**5. Explain About React**

**React:**

* **Component-based**: UIs are broken down into reusable components.
* **Declarative**: You describe *what* you want to show, not *how*.
* **Efficient**: Uses virtual DOM to optimize UI updates.
* **Flexible**: Can be used with Redux, React Router, Axios, etc.

React is primarily for building UI but can be extended for full front-end applications using libraries like:

* **React Router** for routing
* **Redux** or **Context API** for state management
* **Axios/Fetch** for API integration

**6. Define Virtual DOM**

**Virtual DOM** is a **lightweight in-memory representation** of the actual DOM.

**How It Works:**

* When a component changes, a new virtual DOM is created.
* React compares it with the previous version using a **diffing algorithm**.
* Only the parts that changed are updated in the **real DOM** (efficient and fast).

**7. Explain Features of React**

1. **Component-Based Architecture** – Build encapsulated components that manage their own state.
2. **Unidirectional Data Flow** – Props go from parent to child; state is managed internally.
3. **JSX Syntax** – Combines JavaScript and HTML/XML-like syntax.
4. **Lifecycle Methods** – For managing component behavior during mount, update, and unmount.
5. **React Native Support** – For building native mobile apps.
6. **Hooks (since React 16.8)** – Enables state and lifecycle in function components (e.g., useState, useEffect).

--------------------------------------------------------------------------------------------------------------------------

**1. Create a new React Application with the name “myfirstreact”, Run the application to print “Welcome to the first session of React” as heading of that page.**

**Step 1:** Open Terminal and create a new react project

***npx create-react-app myfirstreact***

***cd myfirstreact***

**Step 2:** In App.js, replace the entire code with

**App.js**

import React from 'react';

function App() {

  return (

    <div>

      <h1>Welcome to the first session of React!!!</h1>

    </div>

  );

}

export default App;

**Step 3:** In terminal, type ***npm start*** and this will run your code.

**OUTPUT:**

