Page No.
Date

Mod 6

Q1) Advantages of React.js.

=> ① React.js is a popular javascript library used for building interfaces, particularly single-page applications.

② Advantages of React.js :

(i) It has components based architecture which promotes reusability, better organization and maintainibility of code.

(ii) React.js uses virtual DOM which enhances performance and ensures fast rendering of UI components.

(iii) React uses JSX, which allows developers to write HTML-like syntax with javascript.

(iv) React follows one-way data binding which improves control over data and makes application easier to debug.

(v) React simplifies the process of designing and debugging by using declarative UI.

(vi) React is flexible i.e. it allows customization according to project requirements.

(vii) React is easy to learn which shortens the learning curve and increases developer's productivity.

(viii) React improves SEO of applications compared to traditional javascript frameworks.

(ix) Rich ecosystem of libraries & tools in react makes it easier to build robust and scalable applications.

(x) React supports cross-platform development.

Q2) Features of React.js

=> ① copy from (Q1) first point

② Features of react.js:

(i) components based architecture:

(a) React applications are built using reusable and independent components.

(b) Eg:

```
function Button () {
        return <button> Click </button>;
}
```

(ii) Virtual DOM

(a) React uses a virtual DOM to improve performance.

(b) The virtual DOM is a lightweight representation of the actual DOM.

(c) React only updates the parts of the real DOM that have changed.

(iii) JSX

(a) JSX is a syntax extension for javascript that allows developers to write HTML-like code within javascript.

(b) eg:

```
const greeting = <h1> Hello </h1>;
```

(iv) Unidirectional data flow

(a) React follows a one-way binding approach where data flows from parent component to child component using props.

(b) eg:

```
function Component (props){
        return <h2>{props.message}</h2>;
}
```

(v) State Management

(a) React provides a state object to manage data that changes over time in a component.

(b) This allows dynamic updates to the UI without reloading the page

(c) eg: const [count, setCount] = useState(0);

(vi) React Router

(a) React router is a library for handling dynamic navigations in react applications.

(b) It allows user to create single-page applications with multiple views.

(c) eg: <Route path="/about"
component={AboutPage}/>

(vii) Lifecycle methods

(a) React components have lifecycle methods to manage behavior at different stages of their lifecycle.

(b) Eg: Fetching data when a component mounts using useEffect in functional components.

Note : You can add features in this answer as well

Q3) What is JSX? Explain its attributes with an example.

=> ① JSX also known as javascript XML is a syntax extension that allows developers to write HTML - like code within javascript.

② It is used in React.js to describe the structure of the user interface (UI)

③ Attributes in JSX :

ⓘ className

ⓐ In JSX, class is replaced with className to avoid conflict with javascript's class keyword.

ⓑ eg: <div className = "box" > Hello! </div>

(ii) Inline styles

ⓐ Inline styles are specified as an object with camelCase properties.

ⓑ eg: <h1 style = {{ color : 'blue', fontSize : '24 px' }} > Hey </h1>

(iii) Expressions

ⓐ Javascript expressions can be inserted using { }

ⓑ eg: const name = "React";
<p> Welcome to { name };

(iv) Boolean attributes

ⓐ Boolean attributes like disabled, checked or readOnly do not require a value.

ⓑ Setting them true or false dynamically controls their behaviour.

ⓒ eg: <input type = "checkbox" checked = {true}/>

Ⓥ Custom attributes.

ⓐ Developers can pass custom attributes (props) to React components for dynamic behaviour

(b) eg:< Button label = " click me" />