1️⃣React क्या है?

- React एक JavaScript library है जो user interfaces (UI) बनाने के लिए use होती है।
- Developed by Facebook
- Specially Single Page Applications (SPA) और dynamic web apps बनाने में use होती है।

Key Features: | Feature | Explanation | |---------|------------| | Component-Based | UI को छोटे, reusable components में divide किया जाता है | | Virtual DOM | Efficient updates – सिर्फ changed parts re-render होते हैं | | Declarative | UI को declaratively define करते हैं | | JSX | HTML-like syntax inside JavaScript |

2️⃣React क्यों use करते हैं?

- Reusable Components → बार-बार same component use कर सकते हैं
- Fast Rendering → Virtual DOM की वजह से fast updates
- Maintainable Code → Component-based architecture से large apps manage होते हैं
- Interactive UI → State और props के साथ dynamic content show
- Community & Ecosystem → Libraries और tools support

3️⃣React Installation / Setup

Using Create React App:

```
npx create-react-app my-app
cd my-app
npm start
```

HTML में CDN:

```
<script src="https://unpkg.com/react@18/umd/react.development.js"></script>
<script src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></
script>
```

4️⃣React Components

Functional Component

```
function App() {
  return <h1>Hello World!</h1>;
}
```

Class Component

```
import React, { Component } from "react";
class App extends Component {
  render() {
    return <h1>Hello World!</h1>;
  }
}
```

5️⃣JSX

```
const element = <h1>Hello React</h1>;
const name = "Sunil";
const element2 = <h1>Hello {name}</h1>;
```

6️⃣Props

```
function Greeting({ name }) {
  return <h2>Hello, {name}</h2>;
}
function App() {
  return <Greeting name="Sunil" />;
}
```

7️⃣State

```
import React, { useState } from "react";
function Counter() {
  const [count, setCount] = useState(0);
  return (
    <div>
      <h2>Count: {count}</h2>
      <button onClick={() => setCount(count + 1)}>Increase</button>
    </div>
  );
}
```

8️⃣Conditional Rendering

```
{isLoggedIn ? <Dashboard /> : <Login />}
{messages.length > 0 && <p>You have messages</p>}
```

9️⃣List, Map, Keys

```
const users = ["Sunil", "Aman", "Rakesh"];
<ul>
  {users.map(user => <li key={user}>{user}</li>)}
</ul>
```

1 0️⃣CRUD Example

- Add, Edit, Delete users dynamically using **State + Props + Conditional Rendering + List/Map/Keys**
- Refer to Interactive Dashboard Example (Login + UserCard + AddUserForm + Dashboard)

1 1️⃣Summary Table

Concept	Use	Example
Component	Reusable UI piece	<App />
Props	Parent → Child data	<Greeting name="Sunil" />
State	Dynamic data	const [count,setCount]=useState(0)
Conditional Rendering	Render based on condition	{isLoggedIn ? <Dashboard /> : <Login />}
List/Map/Keys	Render array dynamically	users.map(u=><li key={u}>{u}</li>)
CRUD	Add/Edit/Delete data	Dashboard example

Conclusion

- React = Dynamic UI library for modern web apps
- Features = Component-based, Virtual DOM, Declarative, JSX
- Use = Single Page Apps, Interactive dashboards, E-commerce sites, Social apps
- Key Concepts = Components, Props, State, Conditional Rendering, List/Map/Keys, CRUD

Complete Interactive React Example: Covers Login/Logout, Dynamic User List, Add/Edit/Remove Users, Props, State, List/Map/Keys, Conditional Rendering for practice.