

1. Javascript library for building user interfaces

Focuses mainly on the view layer(UI)

Developed by Facebook.

l

why libraries

React handles only UI, not routing, HTTP, state management by default.

React give flexibility?

2) 'Virtual DOM':

A lightweight Javascript object copy of the real DOM

Lives in memory

Updates are fast.

Real DOM

React updates Virtual DOM first

3) Resuable UI building blocks

Each component returns JSX

can have its own state and props.

use hooks.

4) Hooks.

Functions that let you use state and lifecycle features.

used only in functional components.

Avoid complex class components

cleaner and more readable code.

5. Diff b/w useState and useReducer.

Simple state management

Best for primitive values

Easy to use

useReducer.

Complex static logic

Multiple related values

when to choose

Small / simple state - useState

Large / complex logic - useReducer.

6. Prop drilling.

Passing props through multiple components levels

Even when intermediate components don't need them.

Problems.

Heavy code

7. Lazy Loading

Load components only when needed

uses React.lazy() and suspense

why important

Reduces initial bundle size

Faster page load.

8. purpose of useEffect?

Handles side effects

Eg:

API calls

DOM Manipulation

Timers.

Event Listeners

Dependancy Array Behavior

[] - runs once.

9. Diff b/w useState and useState

useState

Triggers re-render when updated

Used for UI-related data

useRef

Does not trigger re-render

10. Role of key in React lists

Key.

unique identifiers for list items

Why import

Help React identify which items changed

improves performance

Keys must be unique.