**React HOL-13**

**Explain various ways of conditional rendering**

React supports multiple ways to conditionally render elements:

1. **Using if statements:**

if (isLoggedIn) {

return <Dashboard />;

} else {

return <Login />;

}

1. **Using ternary operator (? :):**

{isLoggedIn ? <Dashboard /> : <Login />}

1. **Using logical AND (&&):**

{hasAccess && <AdminPanel />}

1. **Element variables:**  
   Assign JSX to a variable based on a condition and render it later.

let button;

if (isSubscribed) {

button = <Unsubscribe />;

} else {

button = <Subscribe />;

}

return <div>{button}</div>;

**Explain how to render multiple components**

You can render multiple components by placing them together inside a container like a <div>, React.Fragment, or <>...</>:

return (

<>

<Navbar />

<Sidebar />

<Content />

</>

);

**Define list component**

A **list component** is a component that renders a list of items using array methods like map().

**Example:**

function NameList(props) {

return (

<ul>

{props.names.map((name, index) => (

<li key={index}>{name}</li>

))}

</ul>

);

}

**Explain about keys in React applications**

**Keys** help React identify which items in a list have changed, been added, or removed. They must be **unique and stable** among siblings.

**Correct use:**

{items.map(item => <li key={item.id}>{item.name}</li>)}

**Explain how to extract components with keys**

When rendering lists, each item can be passed into a **separate component**, with a unique key.

**Example:**

function ListItem(props) {

return <li>{props.value}</li>;

}

function NameList(props) {

return (

<ul>

{props.names.map((name, index) => (

<ListItem key={index} value={name} />

))}

</ul>

);

}

**Explain React Map, map() function**

React uses JavaScript's built-in map() function to **render lists of elements**.

**Example:**

const numbers = [1, 2, 3];

const listItems = numbers.map((num) => <li key={num}>{num}</li>);

return <ul>{listItems}</ul>;

map() creates a **new array** of JSX elements from a data array.

**CourseDetails.jsx**

import React from 'react';

function CourseDetails() {

const courses = [

{ id: 1, name: 'React Fundamentals' },

{ id: 2, name: 'Advanced React' },

{ id: 3, name: 'Fullstack Dev with MERN' }

];

return (

<div>

<h2>Course Details</h2>

<ul>

{courses.map(course => (

<li key={course.id}>{course.name}</li> // Keyed list component

))}

</ul>

</div>

);

}

export default CourseDetails;

**BookDetails.jsx**

import React from 'react';

function BookDetails() {

const books = ['React Basics', 'JavaScript Mastery', 'Node.js Guide'];

return (

<div>

<h2>Book Details</h2>

<ul>

{books.map((book, index) => (

<li key={index}>{book}</li> // Using keys with map()

))}

</ul>

</div>

);

}

export default BookDetails;

**BlogDetails.jsx**

import React from 'react';

function BlogDetails() {

return (

<div>

<h2>Blog Details</h2>

<p>Check out our latest blogs on React, JavaScript, and more!</p>

</div>

);

}

export default BlogDetails;

**Nav.jsx**

import React from 'react';

function Nav({ onSelect }) {

return (

<div>

<button onClick={() => onSelect('book')}>Book Details</button>

<button onClick={() => onSelect('blog')}>Blog Details</button>

<button onClick={() => onSelect('course')}>Course Details</button>

</div>

);

}

export default Nav;

**App.jsx**

import React, { useState } from 'react';

import BookDetails from './components/BookDetails';

import BlogDetails from './components/BlogDetails';

import CourseDetails from './components/CourseDetails';

import Nav from './components/Nav';

function App() {

const [section, setSection] = useState('book');

// 1. Conditional rendering using if-else

let sectionComponent;

if (section === 'book') {

sectionComponent = <BookDetails />;

} else if (section === 'blog') {

sectionComponent = <BlogDetails />;

} else {

sectionComponent = <CourseDetails />;

}

return (

<div>

<h1>Blogger App</h1>

<Nav onSelect={setSection} />

{/\* 2. Using element variable (sectionComponent) \*/}

{sectionComponent}

{/\* 3. Using ternary operator \*/}

{/\* {section === 'book' ? <BookDetails /> : section === 'blog' ? <BlogDetails /> : <CourseDetails />} \*/}

{/\* 4. Using && operator (only shows book section) \*/}

{/\* {section === 'book' && <BookDetails />} \*/}

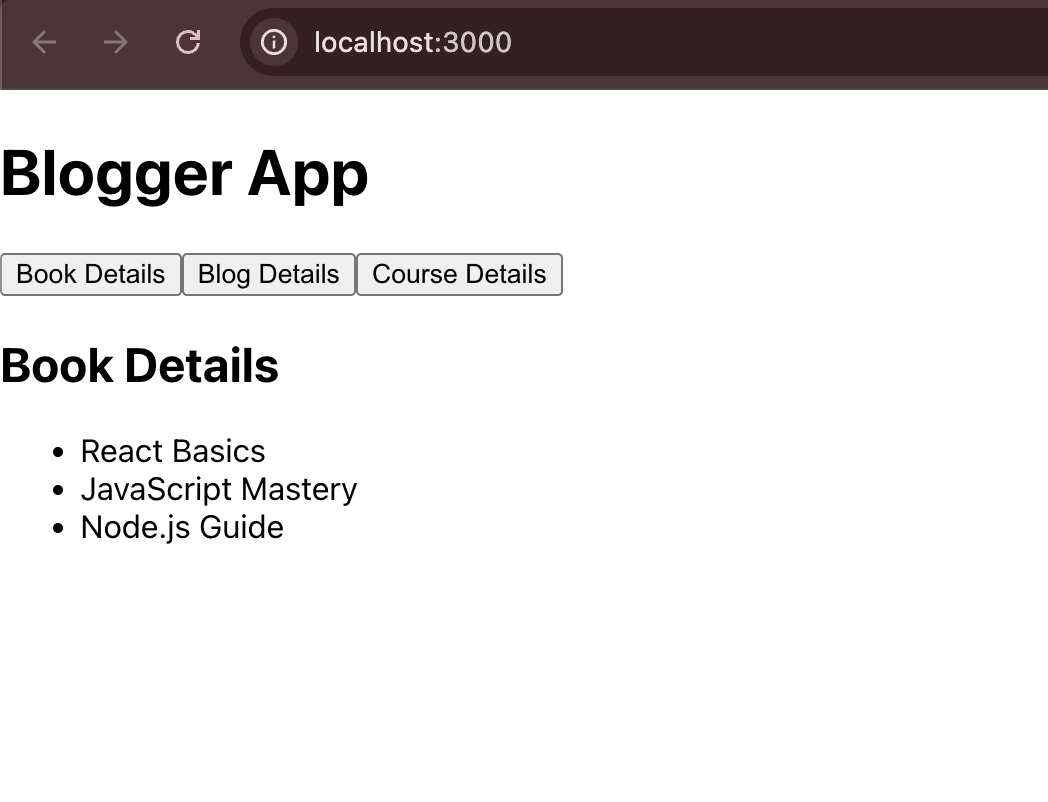
</div>

);

}

export default App;

**Output**

****

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**