Name:Duggirala Bhagya Vivekini

Superset ID: 5673888

Cognizant Digital-Nurture-4.0-JavaFSE- Handson WEEK-7

ReactJS Hands-on Lab: Cricket Application with ES6 Features:

Create a New React Application  
npx create-react-app cricketapp  
Navigate into Your Application's Directory  
cd cricketapp  
Open the Project in Visual Studio Code  
code .  
Create the ListofPlayers.js Component

import React from 'react';

function ListofPlayers() {

const players = [

{ name: 'Virat Kohli', score: 120 },

{ name: 'Rohit Sharma', score: 95 },

{ name: 'KL Rahul', score: 65 },

{ name: 'Shreyas Iyer', score: 40 },

{ name: 'Hardik Pandya', score: 80 },

{ name: 'Ravindra Jadeja', score: 75 },

{ name: 'Jasprit Bumrah', score: 15 },

{ name: 'Mohammed Shami', score: 10 },

{ name: 'Kuldeep Yadav', score: 5 },

{ name: 'Yuzvendra Chahal', score: 20 },

{ name: 'Rishabh Pant', score: 110 }

];

const allPlayersList = players.map((player, index) => (

<li key={index}>{player.name} - Score: {player.score}</li>

));

const playersBelow70 = players.filter(player => player.score < 70);

const playersBelow70List = playersBelow70.map((player, index) => (

<li key={index}>{player.name} - Score: {player.score}</li>

));

return (

<div style={{ padding: '20px', border: '1px solid #ccc', margin: '15px', borderRadius: '8px', boxShadow: '0 2px 4px rgba(0,0,0,0.1)' }}>

<h2>All Players & Scores</h2>

<ul>{allPlayersList}</ul>

<h2>Players with Scores Below 70 (Filtered using Arrow Functions)</h2>

{playersBelow70.length > 0 ? (

<ul>{playersBelow70List}</ul>

) : (

<p>No players found with scores below 70.</p>

)}

</div>

);

}

export default ListofPlayers;

Create the IndianPlayers.js Component

import React from 'react';

function IndianPlayers() {

const indianCricketTeam = [

"Rohit Sharma (C)", "KL Rahul (WK)", "Shubman Gill", "Virat Kohli",

"Shreyas Iyer", "Hardik Pandya", "Ravindra Jadeja", "Mohammed Shami",

"Jasprit Bumrah", "Kuldeep Yadav", "Mohammed Siraj"

];

const [oddPlayer1, , oddPlayer2, , oddPlayer3, , oddPlayer4, , oddPlayer5, , oddPlayer6] = indianCricketTeam;

const [, evenPlayer1, , evenPlayer2, , evenPlayer3, , evenPlayer4, , evenPlayer5] = indianCricketTeam;

const t20Players = ["Suryakumar Yadav", "Rinku Singh", "Jitesh Sharma"];

const ranjiTrophyPlayers = ["Hanuma Vihari", "Sarfaraz Khan", "Priyank Panchal"];

const mergedPlayers = [...t20Players, ...ranjiTrophyPlayers];

return (

<div style={{ padding: '20px', border: '1px solid #ccc', margin: '15px', borderRadius: '8px', boxShadow: '0 2px 4px rgba(0,0,0,0.1)' }}>

<h2>Indian Team (Destructuring Example)</h2>

<h3>Odd Team Players:</h3>

<ul>

<li>{oddPlayer1}</li>

<li>{oddPlayer2}</li>

<li>{oddPlayer3}</li>

<li>{oddPlayer4}</li>

<li>{oddPlayer5}</li>

<li>{oddPlayer6}</li>

</ul>

<h3>Even Team Players:</h3>

<ul>

<li>{evenPlayer1}</li>

<li>{evenPlayer2}</li>

<li>{evenPlayer3}</li>

<li>{evenPlayer4}</li>

<li>{evenPlayer5}</li>

</ul>

<h2>Merged Players (T20 & Ranji Trophy)</h2>

<ul>

{mergedPlayers.map((player, index) => (

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

))}

</ul>

</div>

);

}

export default IndianPlayers;  
  
Modify App.js to Conditionally Render Components

import React from 'react';

import './App.css';

import ListofPlayers from './ListofPlayers';

import IndianPlayers from './IndianPlayers';

function App() {

const displayListofPlayers = true;

return (

<div className="App" style={{ textAlign: 'center', fontFamily: 'Arial, sans-serif' }}>

<h1>Cricket App</h1>

{displayListofPlayers ? (

<ListofPlayers />

) : (

<IndianPlayers />

)}

</div>

);

}

export default App;  
  
Run the React Application  
npm start  
  
OUTPUT:

A. Output when displayListofPlayers is true  
set const displayListofPlayers = true;

Cricket App

* All Players & Scores
* \* Virat Kohli - Score: 120
* \* Rohit Sharma - Score: 95
* \* KL Rahul - Score: 65
* \* Shreyas Iyer - Score: 40
* \* Hardik Pandya - Score: 80
* \* Ravindra Jadeja - Score: 75
* \* Jasprit Bumrah - Score: 15
* \* Mohammed Shami - Score: 10
* \* Kuldeep Yadav - Score: 5
* \* Yuzvendra Chahal - Score: 20
* \* Rishabh Pant - Score: 110
* Players with Scores Below 70 (Filtered using Arrow Functions)
* \* KL Rahul - Score: 65
* \* Shreyas Iyer - Score: 40
* \* Jasprit Bumrah - Score: 15
* \* Mohammed Shami - Score: 10
* \* Kuldeep Yadav - Score: 5
* \* Yuzvendra Chahal - Score: 20

**B. Output when displayListofPlayers is false**

set const displayListofPlayers = false;

Cricket App

* Indian Team (Destructuring Example)
* Odd Team Players:
* \* Rohit Sharma (C)
* \* Shubman Gill
* \* Shreyas Iyer
* \* Ravindra Jadeja
* \* Jasprit Bumrah
* \* Kuldeep Yadav
* Even Team Players:
* \* KL Rahul (WK)
* \* Virat Kohli
* \* Hardik Pandya
* \* Mohammed Shami
* \* Mohammed Siraj
* Merged Players (T20 & Ranji Trophy)
* \* Suryakumar Yadav
* \* Rinku Singh
* \* Jitesh Sharma
* \* Hanuma Vihari
* \* Sarfaraz Khan

\* Priyank Panchal

hands-on lab task **“officespacerentalapp”** using **ReactJS with JSX**, inline CSS, and DOM rendering  
Create the App  
npx create-react-app officespacerentalapp  
cd officespacerentalapp

npm start

Replace App.js Code  
import React from 'react';

import './App.css';

function App() {

const offices = [

{

name: "GreenSpace Tower",

rent: 55000,

address: "MG Road, Bangalore",

image: "https://via.placeholder.com/300x200?text=Office+1"

},

{

name: "Skyline Hub",

rent: 75000,

address: "Hitech City, Hyderabad",

image: "https://via.placeholder.com/300x200?text=Office+2"

},

{

name: "WorkNest",

rent: 60000,

address: "Sector 44, Gurgaon",

image: "https://via.placeholder.com/300x200?text=Office+3"

}

];

return (

<div className="App">

<h1>🏢 Office Space Rental Listings</h1>

{offices.map((office, index) => (

<div key={index} style={{ border: "1px solid gray", margin: "10px", padding: "10px", width: "320px" }}>

<img src={office.image} alt={office.name} style={{ width: "100%", height: "200px" }} />

<h2>{office.name}</h2>

<p><strong>Address:</strong> {office.address}</p>

<p style={{ color: office.rent < 60000 ? "red" : "green" }}>

<strong>Rent:</strong> ₹{office.rent}

</p>

</div>

))}

</div>

);

}

export default App;

App.css Styling  
.App {

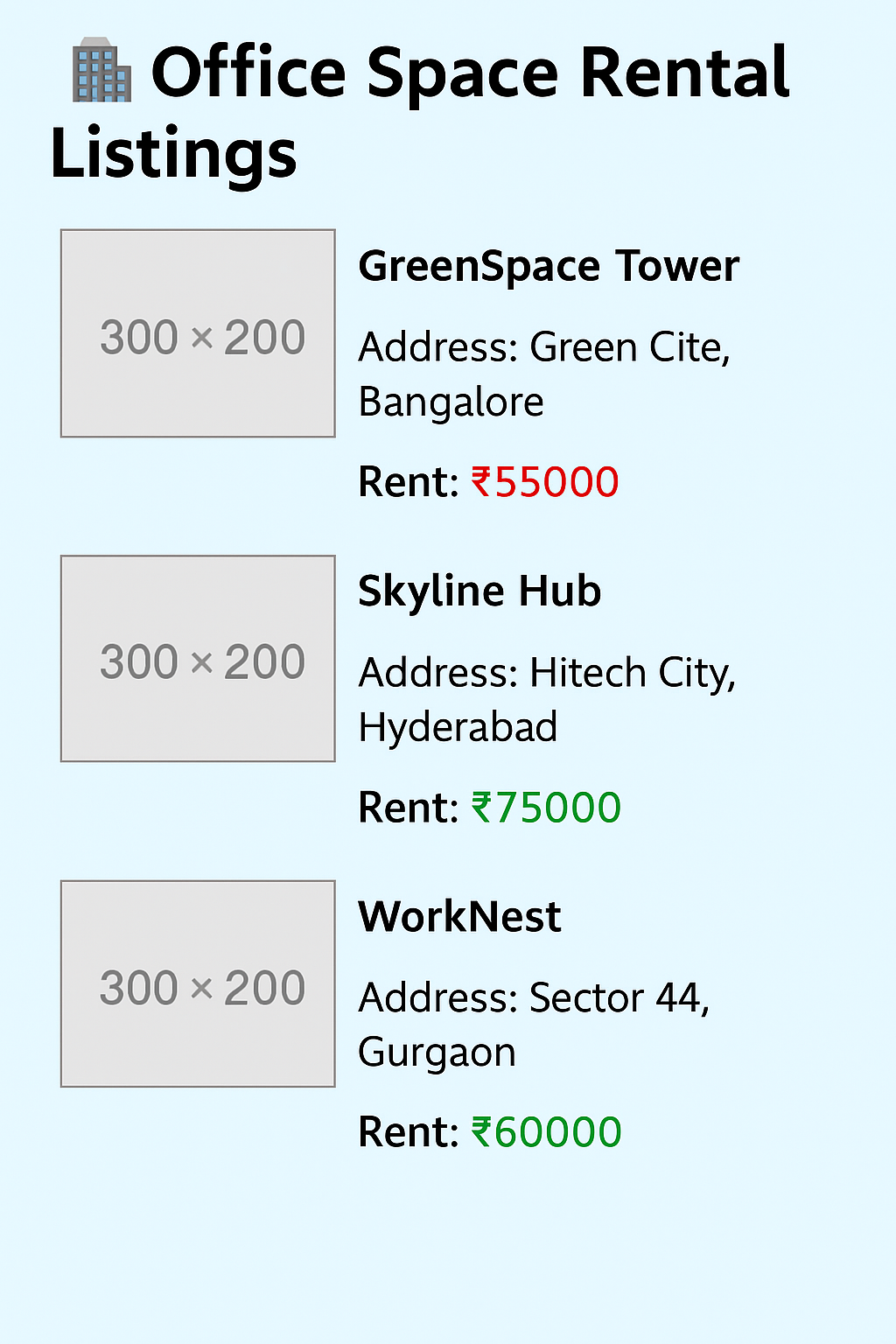
text-align: center;

font-family: Arial, sans-serif;

padding: 20px;

background-color: #f0f8ff;

}

OUTPUT:  
  
  
**"eventexamplesapp"** ReactJS hands-on lab.

Project Setup Command  
npx create-react-app eventexamplesapp

cd eventexamplesapp

npm start

App.js Code  
import React, { useState } from 'react';

import './App.css';

function App() {

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

const [message, setMessage] = useState('');

const [euro, setEuro] = useState('');

const [rupees, setRupees] = useState('');

const increment = () => {

setCount(count + 1);

sayHello();

};

const decrement = () => {

setCount(count - 1);

};

const sayHello = () => {

setMessage('Hello! You clicked Increment!');

};

const sayWelcome = (text) => {

setMessage(text);

};

const handleClick = (event) => {

alert('I was clicked');

};

const handleEuroChange = (e) => {

setEuro(e.target.value);

};

const handleSubmit = () => {

const rupeeValue = parseFloat(euro) \* 90; // Assuming 1 Euro = 90 INR

setRupees(rupeeValue.toFixed(2));

};

return (

<div className="App">

<h1>React Event Handling Demo</h1>

<h2>Counter: {count}</h2>

<button onClick={increment}>Increment</button>

<button onClick={decrement}>Decrement</button>

<p>{message}</p>

<button onClick={() => sayWelcome('Welcome to React Event Handling!')}>Say Welcome</button>

<button onClick={handleClick}>Click me (Synthetic Event)</button>

<hr />

<h2>Currency Converter (Euro to INR)</h2>

<input type="number" placeholder="Enter Euro" value={euro} onChange={handleEuroChange} />

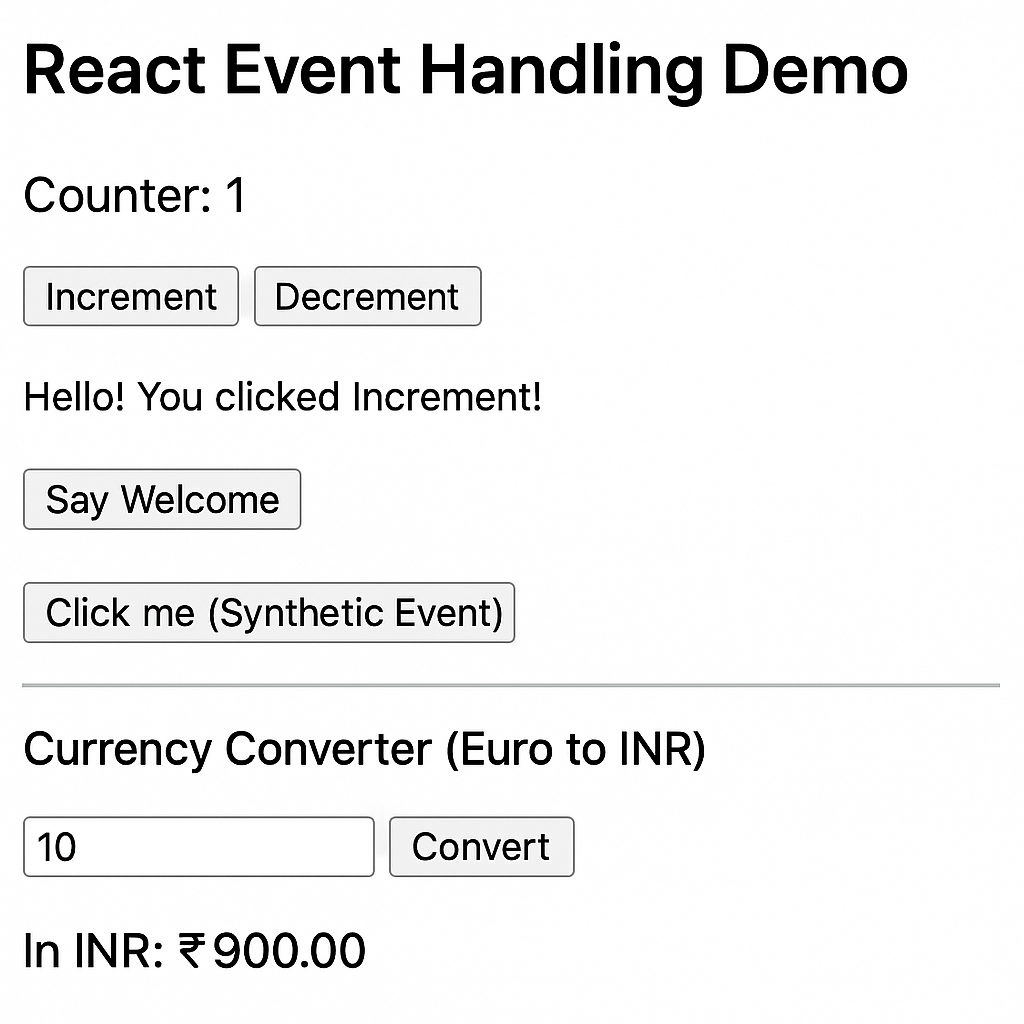
<button onClick={handleSubmit}>Convert</button>

{rupees && <p>In INR: ₹{rupees}</p>}

</div>

);

}

export default App;  
  
OUTPUT:  


Hands-on for Create a React app named **"ticketbookingapp"** that demonstrates **conditional rendering**:

* A **guest** sees the flight details but **cannot book** tickets.
* A **logged-in user** can **book** tickets.
* Display different components/pages based on **Login/Logout** button.

App.js

import React, { useState } from 'react';

import GuestPage from './GuestPage';

import UserPage from './UserPage';

function App() {

const [isLoggedIn, setIsLoggedIn] = useState(false);

const handleLogin = () => setIsLoggedIn(true);

const handleLogout = () => setIsLoggedIn(false);

return (

<div style={{ textAlign: 'center', marginTop: '50px' }}>

<h1>Flight Ticket Booking App</h1>

{

isLoggedIn ? (

<>

<UserPage />

<button onClick={handleLogout}>Logout</button>

</>

) : (

<>

<GuestPage />

<button onClick={handleLogin}>Login</button>

</>

)

}

</div>

);

}

export default App;

GuestPage.js  
import React from 'react';

function GuestPage() {

return (

<div>

<h2>Welcome, Guest!</h2>

<p>You can view available flights, but please login to book tickets.</p>

<ul>

<li> Flight A123 - Hyderabad to Bangalore</li>

<li> Flight B456 - Chennai to Mumbai</li>

</ul>

</div>

);

}

export default GuestPage;

UserPage.js  
import React from 'react';

function UserPage() {

return (

<div>

<h2>Welcome, User!</h2>

<p>You are logged in. You can now book tickets.</p>

<button>Book Flight A123</button>

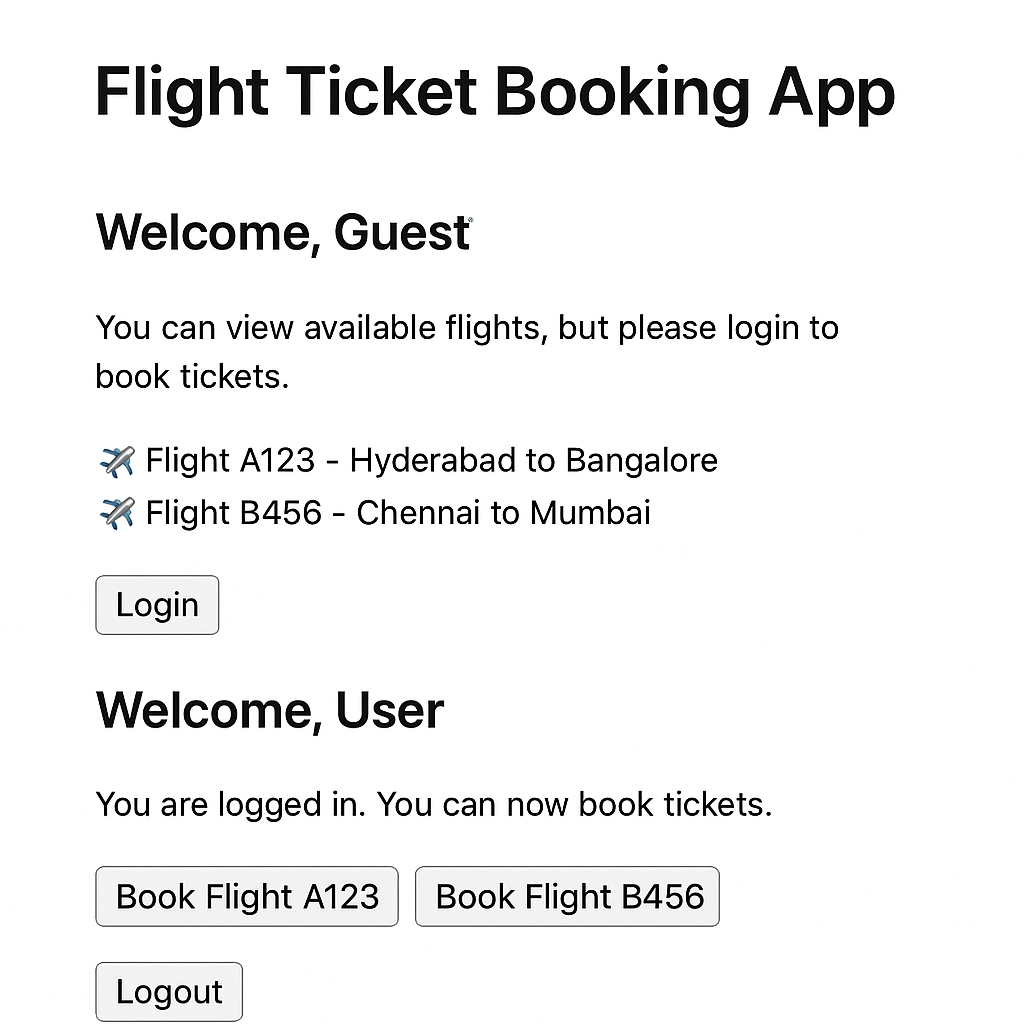
<button>Book Flight B456</button>

</div>

);

}

export default UserPage;

OUTPUT:  


Hands on for Creating a React app called **bloggerapp** with 3 components:

* **BookDetails**
* **BlogDetails**
* **CourseDetails**

Implement **conditional rendering** using **as many ways as possible**

**BookDetails.js**import React from 'react';

function BookDetails() {

return (

<div>

<h2>Book Details</h2>

<p>Title: React Simplified</p>

<p>Author: Dan Abramov</p>

</div>

);

}:  
  
  
**BlogDetails.js**

import React from 'react';

function BlogDetails() {

return (

<div>

<h2>Blog Details</h2>

<p>Title: Understanding JSX</p>

<p>Author: Sophie Alpert</p>

</div>

);

}

export default BlogDetails;  
  
CourseDetails.js  
import React from 'react';

function CourseDetails() {

return (

<div>

<h2>Course Details</h2>

<p>Course: Full Stack Development</p>

<p>Platform: Coursera</p>

</div>

);

}

export default CourseDetails;  
  
**App.js** with multiple types of conditional rendering

import React, { useState } from 'react';

import BookDetails from './components/BookDetails';

import BlogDetails from './components/BlogDetails';

import CourseDetails from './components/CourseDetails';

function App() {

const [activeComponent, setActiveComponent] = useState("book");

const showComponent = () => {

switch (activeComponent) {

case "book":

return <BookDetails />;

case "blog":

return <BlogDetails />;

case "course":

return <CourseDetails />;

default:

return <h2>No Component Selected</h2>;

}

};

const shouldShowCourse = true; // for logical && example

return (

<div style={{ padding: '20px' }}>

<h1>BloggerApp - Conditional Rendering</h1>

<button onClick={() => setActiveComponent("book")}>Show Book</button>

<button onClick={() => setActiveComponent("blog")}>Show Blog</button>

<button onClick={() => setActiveComponent("course")}>Show Course</button>

<hr />

{/\* 1. Using switch-case \*/}

{showComponent()}

{/\* 2. Using ternary operator \*/}

<div>

<h3>Using Ternary Operator:</h3>

{activeComponent === "book" ? <BookDetails /> : <p>Not Book Details</p>}

</div>

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

<div>

<h3>Using Logical &&:</h3>

{shouldShowCourse && <CourseDetails />}

</div>

{/\* 4. Using if-else in code block \*/}

<div>

<h3>Using If-Else:</h3>

{

(() => {

if (activeComponent === "blog") return <BlogDetails />;

else return <p>No Blog Selected</p>;

})()

}

</div>

</div>

);

}

export default App;

OUTPUT:  
