Digital Nurture 4.0 – Deep Skilling Program

Java Full Stack Engineer (FSE)

Week 7– Mandatory Exercises Submission

Name: Adishree Devoor

GitHub Profile: https://github.com/Adishree16

Email: [adishreedevoor18@gmail.com](mailto:adishreedevoor18@gmail.com)

Skills Covered:

* React

**React**

**ReactJS Hands-On Lab**

**Question**

Create a React app called cricketapp with two components:

1. ListofPlayers

- Use map() to display 11 players' names and scores.

- Use arrow functions to filter players with score < 70.

2. IndianPlayers

- Use destructuring to show: Odd Team Players and Even Team Players

- Declare two arrays: T20players and RanjiTrophyPlayers

- Use spread operator to merge them.

- Show both components conditionally using a flag variable.

**Program**

App.js

import React from 'react';  
import ListofPlayers from './ListofPlayers';  
import IndianPlayers from './IndianPlayers';  
  
function App() {  
 const flag = true; // Change this to false to test IndianPlayers  
  
 return (  
 <div className="App">  
 <h1>Cricket App</h1>  
 {flag ? <ListofPlayers /> : <IndianPlayers />}  
 </div>  
 );  
}  
  
export default App;

ListofPlayers.js

import React from 'react';  
  
const ListofPlayers = () => {  
 const players = [  
 { name: 'Virat Kohli', score: 98 },  
 { name: 'Rohit Sharma', score: 85 },  
 { name: 'MS Dhoni', score: 60 },  
 { name: 'KL Rahul', score: 45 },  
 { name: 'Jadeja', score: 75 },  
 { name: 'Shami', score: 50 },  
 { name: 'Bumrah', score: 80 },  
 { name: 'Surya Kumar', score: 65 },  
 { name: 'Gill', score: 55 },  
 { name: 'Pant', score: 90 },  
 { name: 'Ashwin', score: 40 }  
 ];  
  
 const filteredPlayers = players.filter(player => player.score < 70);  
  
 return (  
 <div>  
 <h2>List of All Players</h2>  
 {players.map((player, index) => (  
 <p key={index}>{player.name} - {player.score}</p>  
 ))}  
  
 <h3>Players with Score below 70</h3>  
 {filteredPlayers.map((player, index) => (  
 <p key={index}>{player.name} - {player.score}</p>  
 ))}  
 </div>  
 );  
};  
  
export default ListofPlayers;

IndianPlayers.js

import React from 'react';  
  
const IndianPlayers = () => {  
 const allPlayers = [  
 'Virat', 'Rohit', 'Dhoni', 'Rahul', 'Jadeja', 'Shami', 'Bumrah', 'Surya', 'Gill', 'Pant', 'Ashwin'  
 ];  
  
 const oddPlayers = [];  
 const evenPlayers = [];  
  
 allPlayers.forEach((player, index) => {  
 if ((index + 1) % 2 === 0) {  
 evenPlayers.push(player);  
 } else {  
 oddPlayers.push(player);  
 }  
 });  
  
 const T20players = ['Hardik', 'Ishaan', 'Samson'];  
 const RanjiTrophyPlayers = ['Pujara', 'Rahane', 'Saha'];  
 const mergedPlayers = [...T20players, ...RanjiTrophyPlayers];  
  
 return (  
 <div>  
 <h2>Odd Team Players</h2>  
 {oddPlayers.map((p, i) => <p key={i}>{p}</p>)}  
  
 <h2>Even Team Players</h2>  
 {evenPlayers.map((p, i) => <p key={i}>{p}</p>)}  
  
 <h2>Merged T20 and Ranji Trophy Players</h2>  
 {mergedPlayers.map((p, i) => <p key={i}>{p}</p>)}  
 </div>  
 );  
};  
  
export default IndianPlayers;

Output

When flag = true:



When flag = false:



**ReactJS Hands-On Lab – JSX & Inline CSS (OfficeSpaceRentalApp)**

**Question**

Create a React application called 'officespacerentalapp' using JSX that includes:

1. A heading displayed using JSX.

2. An image of an office using JSX attributes.

3. An object with office details: Name, Rent, Address.

4. A list of office space objects displayed using map().

5. Apply inline CSS:

- Rent < 60000 → Red

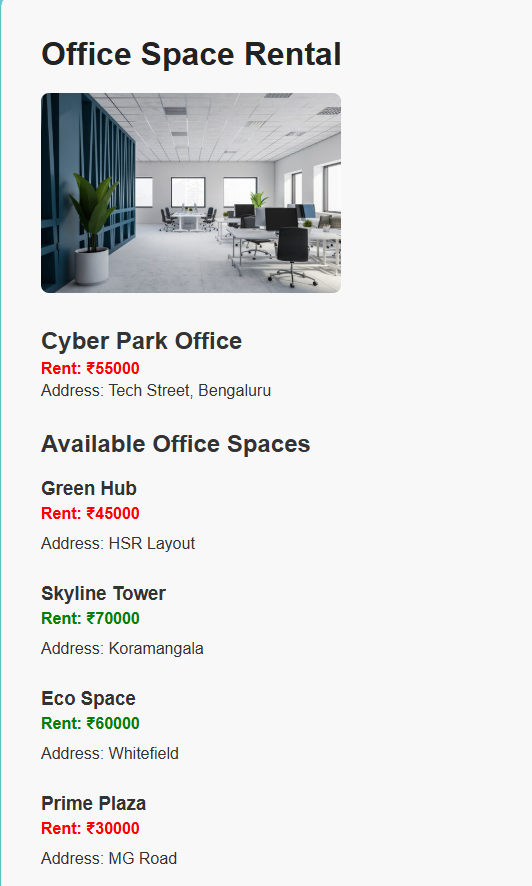
- Rent ≥ 60000 → Green

**Program**

App.js Code

import React from 'react';  
import './App.css';  
  
function App() {  
 const heading = <h1>Office Space Rental</h1>;  
  
 const office = {  
 name: "Cyber Park Office",  
 rent: 55000,  
 address: "Tech Street, Bengaluru",  
 image: "https://via.placeholder.com/300x200.png?text=Office+Space"  
 };  
  
 const officeList = [  
 { name: "Green Hub", rent: 45000, address: "HSR Layout" },  
 { name: "Skyline Tower", rent: 70000, address: "Koramangala" },  
 { name: "Eco Space", rent: 60000, address: "Whitefield" },  
 { name: "Prime Plaza", rent: 30000, address: "MG Road" }  
 ];  
  
 const getRentStyle = (rent) => ({  
 color: rent < 60000 ? "red" : "green"  
 });  
  
 return (  
 <div className="App">  
 {heading}  
 <img src={office.image} alt="Office" style={{ width: "300px", height: "200px" }} />  
 <h2>{office.name}</h2>  
 <p style={getRentStyle(office.rent)}>Rent: ₹{office.rent}</p>  
 <p>Address: {office.address}</p>  
  
 <h2>Available Office Spaces</h2>  
 {officeList.map((off, index) => (  
 <div key={index}>  
 <h3>{off.name}</h3>  
 <p style={getRentStyle(off.rent)}>Rent: ₹{off.rent}</p>  
 <p>Address: {off.address}</p>  
 </div>  
 ))}  
 </div>  
 );  
}  
  
export default App;

**Output**



**ReactJS Hands-On Lab – Event Handling & Currency Converter**

**Question**

Create a React app named 'eventexamplesapp' with the following features:

1. A counter with:

- 'Increment' button that calls multiple methods:

a. Increases count

b. Logs 'Hello' + static message

- 'Decrement' button that reduces the count

2. A 'Say Welcome' button that calls a function with 'Welcome' as an argument.

3. A synthetic event button that shows alert 'I was clicked' and logs event type.

4. A CurrencyConverter component that converts Rupees to Euro using a handleSubmit event.

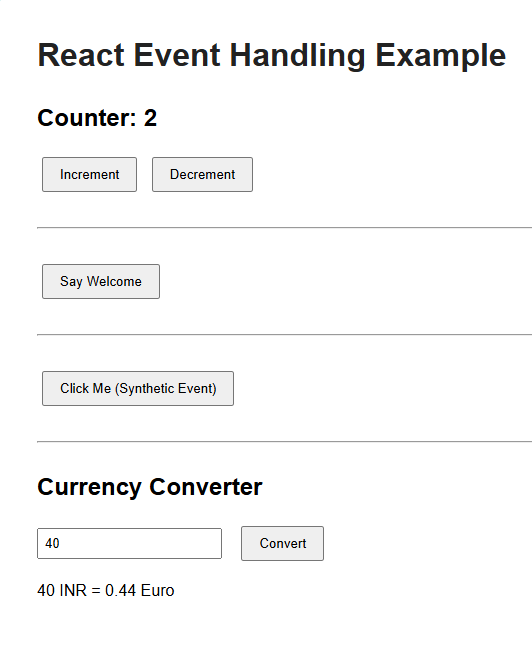
App.js Code

import React, { useState } from 'react';  
import CurrencyConverter from './CurrencyConverter';  
  
function App() {  
 const [count, setCount] = useState(0);  
  
 const handleIncrement = () => {  
 increaseValue();  
 sayHello();  
 };  
  
 const increaseValue = () => {  
 setCount(count + 1);  
 };  
  
 const sayHello = () => {  
 console.log("Hello, this is a static message!");  
 };  
  
 const handleDecrement = () => {  
 setCount(count - 1);  
 };  
  
 const sayWelcome = (msg) => {  
 alert(msg);  
 };  
  
 const handleSyntheticClick = (e) => {  
 alert("I was clicked");  
 console.log("Synthetic Event Type:", e.type);  
 };  
  
 return (  
 <div style={{ margin: '40px' }}>  
 <h1>React Event Handling Example</h1>  
  
 <h2>Counter: {count}</h2>  
 <button onClick={handleIncrement}>Increment</button>  
 <button onClick={handleDecrement}>Decrement</button>  
  
 <hr />  
  
 <button onClick={() => sayWelcome("Welcome")}>Say Welcome</button>  
  
 <hr />  
  
 <button onClick={(e) => handleSyntheticClick(e)}>Click Me (Synthetic Event)</button>  
  
 <hr />  
  
 <CurrencyConverter />  
 </div>  
 );  
}  
  
export default App;

CurrencyConverter.js Code

import React, { useState } from 'react';  
  
function CurrencyConverter() {  
 const [rupees, setRupees] = useState('');  
 const [euros, setEuros] = useState('');  
  
 const handleSubmit = () => {  
 const conversionRate = 0.011; // Example: 1 INR = 0.011 Euro  
 const result = parseFloat(rupees) \* conversionRate;  
 setEuros(result.toFixed(2));  
 };  
  
 return (  
 <div>  
 <h2>Currency Converter</h2>  
 <input  
 type="number"  
 placeholder="Enter amount in INR"  
 value={rupees}  
 onChange={(e) => setRupees(e.target.value)}  
 />  
 <button onClick={handleSubmit}>Convert</button>  
 {euros && <p>{rupees} INR = {euros} Euro</p>}  
 </div>  
 );  
}  
  
export default CurrencyConverter;

**Output**



**ReactJS Hands-On Lab – Conditional Rendering (Ticket Booking App)**

**Question**

Create a React Application named 'ticketbookingapp' with the following features:

1. Guest users can browse the flight details but cannot book tickets.

2. Logged-in users can book tickets.

3. Display different pages based on login state using conditional rendering.

4. Show 'Login' button when user is not logged in.

5. Show 'Logout' button and user page when logged in.

6. Clicking 'Login' changes to user view, clicking 'Logout' switches to guest view.

App.js Code

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={{ margin: '30px' }}>  
 <h1>Flight Ticket Booking</h1>  
  
 <div style={{ marginBottom: '20px' }}>  
 {isLoggedIn ? (  
 <button onClick={handleLogout}>Logout</button>  
 ) : (  
 <button onClick={handleLogin}>Login</button>  
 )}  
 </div>  
  
 {isLoggedIn ? <UserPage /> : <GuestPage />}  
 </div>  
 );  
}  
  
export default App;

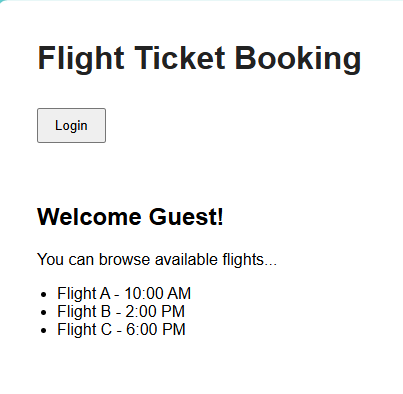
GuestPage.js Code

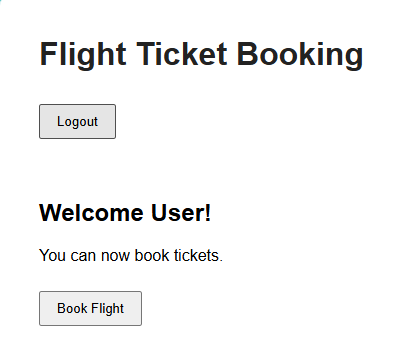
import React from 'react';  
  
function GuestPage() {  
 return (  
 <div>  
 <h2>Welcome Guest!</h2>  
 <p>You can browse available flights, but booking is allowed only after login.</p>  
 <ul>  
 <li>Flight A - 10:00 AM</li>  
 <li>Flight B - 2:00 PM</li>  
 <li>Flight C - 6:00 PM</li>  
 </ul>  
 </div>  
 );  
}  
  
export default GuestPage;

UserPage.js Code

import React from 'react';  
  
function UserPage() {  
 return (  
 <div>  
 <h2>Welcome User!</h2>  
 <p>You can now book tickets.</p>  
 <button>Book Flight</button>  
 </div>  
 );  
}  
  
export default UserPage;

**Output**





**ReactJS Conditional Rendering Lab - bloggerapp**

**Question**

Create a React App named 'bloggerapp' with 3 components:  
- BookDetails  
- BlogDetails  
- CourseDetails  
  
Implement conditional rendering in as many ways as possible using these components.

**Program**

App.js

import React, { useState } from "react";  
import BookDetails from "./BookDetails";  
import BlogDetails from "./BlogDetails";  
import CourseDetails from "./CourseDetails";  
  
function App() {  
 const [selectedComponent, setSelectedComponent] = useState("book");  
  
 return (  
 <div className="App">  
 <h1>Welcome to Blogger App</h1>  
  
 <div>  
 <button onClick={() => setSelectedComponent("book")}>Book</button>  
 <button onClick={() => setSelectedComponent("blog")}>Blog</button>  
 <button onClick={() => setSelectedComponent("course")}>Course</button>  
 </div>  
  
 {/\* Conditional Rendering Methods \*/}  
  
 {/\* 1. If-Else method \*/}  
 {(() => {  
 if (selectedComponent === "book") return <BookDetails />;  
 else if (selectedComponent === "blog") return <BlogDetails />;  
 else if (selectedComponent === "course") return <CourseDetails />;  
 else return <div>Select a component</div>;  
 })()}  
  
 {/\* 2. Ternary Operator \*/}  
 {/\* selectedComponent === "book" ? <BookDetails /> : <BlogDetails /> \*/}  
  
 {/\* 3. Logical && Operator \*/}  
 {/\* selectedComponent === "course" && <CourseDetails /> \*/}  
 </div>  
 );  
}  
  
export default App;

BookDetails.js

import React from "react";  
const BookDetails = () => {  
 return (  
 <div>  
 <h2>Book Details</h2>  
 <p>Title: React for Beginners</p>  
 <p>Author: Adishree Devoor</p>  
 </div>  
 );  
};  
export default BookDetails;

BlogDetails.js

import React from "react";  
const BlogDetails = () => {  
 return (  
 <div>  
 <h2>Blog Details</h2>  
 <p>Title: React Conditional Rendering</p>  
 <p>Date: August 1, 2025</p>  
 </div>  
 );  
};  
export default BlogDetails;

CourseDetails.js

import React from "react";  
const CourseDetails = () => {  
 return (  
 <div>  
 <h2>Course Details</h2>  
 <p>Course: Frontend with React</p>  
 <p>Duration: 6 weeks</p>  
 </div>  
 );  
};  
export default CourseDetails;

**Output**

