**Week 7 React HandsOn**

Create a React Application named “cricketapp” with the following components:

1. ListofPlayers

* Declare an array with 11 players and store details of their names and scores using the map feature of ES6



* Filter the players with scores below 70 using arrow functions of ES6.



1. IndianPlayers
   1. Display the Odd Team Player and Even Team players using the Destructuring features of ES6



* 1. Declare two arrays T20players and RanjiTrophy players and merge the two arrays and display them using the Merge feature of ES6



Display these two components in the same home page using a simple if else in the flag variable.

**Output:**

When Flag=true



When Flag=false



**Hint:**



**App.js**  
  
import './App.css';

import { players, ListofPlayers, Scorebelow70 } from './Components/ListofPlayers';

import { IndianTeam, OddPlayers, EvenPlayers, IndianPlayers, ListofIndianPlayers } from './Components/IndianPlayers';

function App() {

  var flag = true;

  if (flag === true) {

    return (

      <div>

        <h1> List of Players </h1>

        <ListofPlayers players={players} />

        <hr />

        <h1> List of Players having Scores Less than 70 </h1>

        <Scorebelow70 players={players} />

      </div>

    )

  }

  else {

    return (

      <div>

        <div>

          <h1> Indian Team </h1>

          <h1> Odd Players </h1>

          {OddPlayers(IndianTeam)}

          <hr />

          <h1> Even Players </h1>

          {EvenPlayers(IndianTeam)}

        </div>

        <hr />

        <div>

          <h1> List of Indian Players Merged:</h1>

          <ListofIndianPlayers IndianPlayers={IndianPlayers} />

        </div>

      </div>

    )

  }

}

export default App;

**Components/IndianPlayers.js**  
  
export const IndianTeam = [

    "Sachin1",

    "Dhoni2",

    "Virat3",

    "Rohit4",

    "Yuvaraj5",

    "Raina6",

];

export function OddPlayers([first, , third, , fifth]) {

    return (

        <div>

            <li> First: {first} </li>

            <li> Third: {third} </li>

            <li> Fifth: {fifth} </li>

        </div>

    );

}

export function EvenPlayers([, second, , fourth, , sixth]) {

    return (

        <div>

            <li> Second: {second} </li>

            <li> Fourth: {fourth} </li>

            <li> Sixth: {sixth} </li>

        </div>

    );

}

const T20Players = ["First Player", "Second Player", "Third Player"];

const RanjiTrophyPlayers = ["Fourth Player", "Fifth Player", "Sixth Player"];

export const IndianPlayers = [...T20Players, ...RanjiTrophyPlayers];

export const ListofIndianPlayers = ({ IndianPlayers }) => {

    return (

        IndianPlayers.map((item, index) => {

            return (

                <div key={index}>

                    <li>Mr. {item}</li>

                </div>

            )

        })

    )

}

**Components/ListofPlayers.js**  
  
export const players = [

    { name: 'Jack', score: 50 },

    { name: 'Michael', score: 70 },

    { name: 'John', score: 40 },

    { name: 'Ann', score: 61 },

    { name: 'Elisabeth', score: 61 },

    { name: 'Sachin', score: 95 },

    { name: 'Dhoni', score: 100 },

    { name: 'Virat', score: 84 },

    { name: 'Jadeja', score: 64 },

    { name: 'Raina', score: 75 },

    { name: 'Rohit', score: 80 }

];

export const ListofPlayers = ({ players }) => {

    return (

        players.map((item) => {

            return (

                <div key={item.name}>

                    <li>Mr. {item.name}<span> {item.score} </span></li>

                </div>

            )

        })

    )

}

export const Scorebelow70 = ({ players }) => {

    const players70 = [];

    return (

        players.map((item) => {

            if (item.score <= 70) {

                players70.push(item);

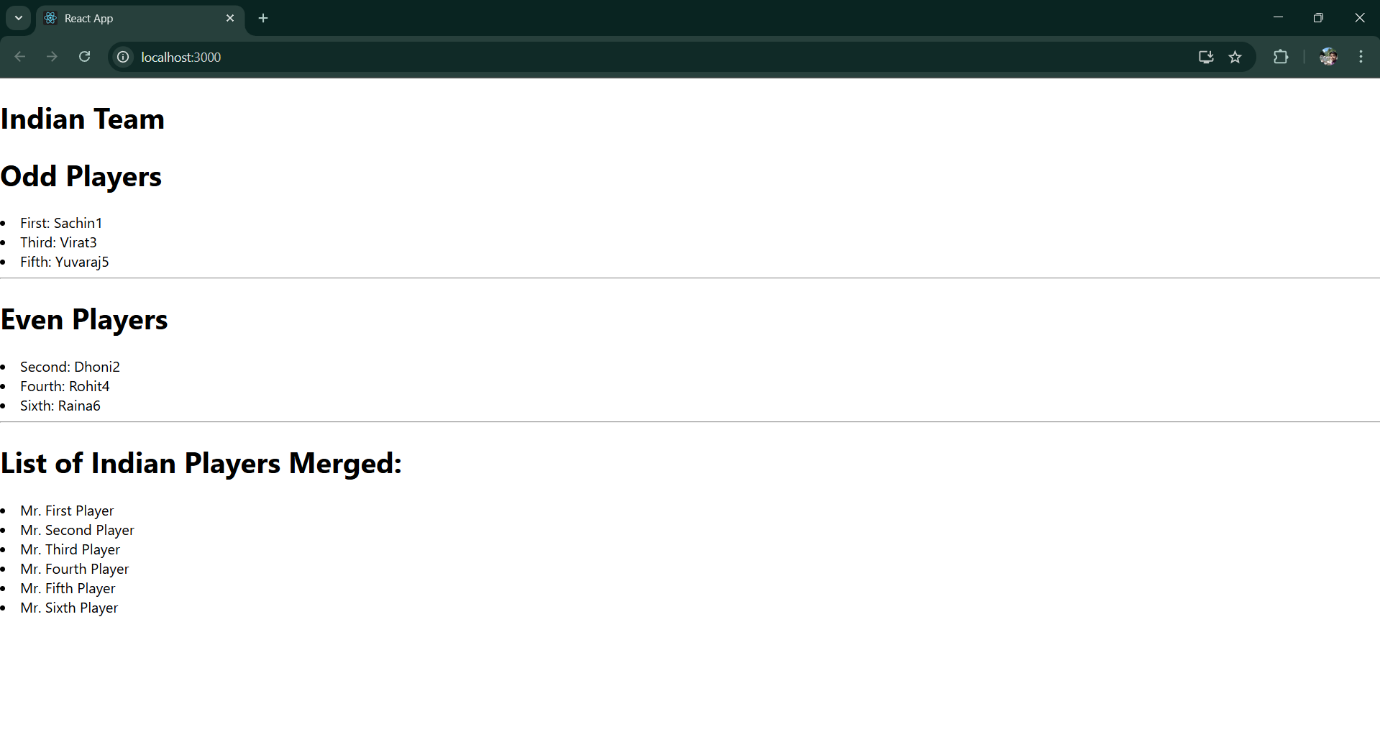
            }

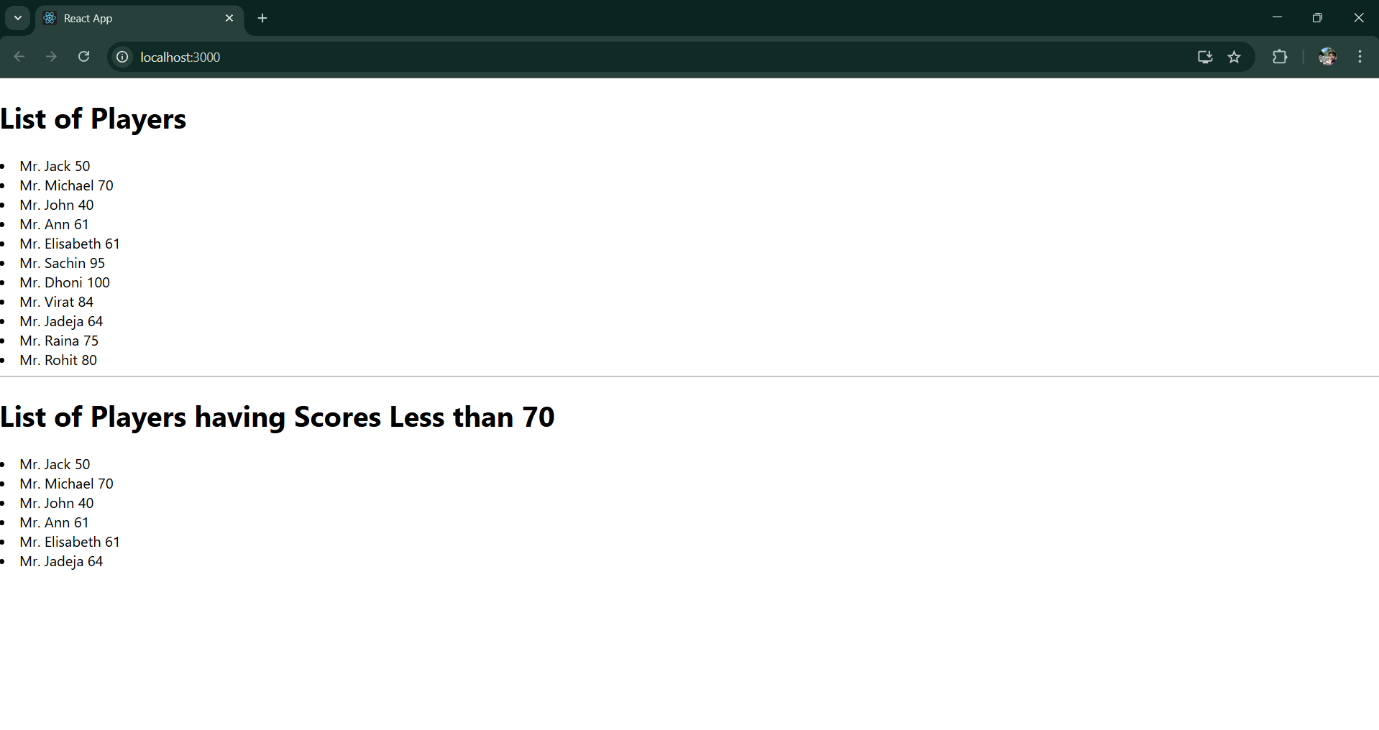
        }),

        <ListofPlayers players={players70} />

    )

}

**OUTPUT**  
  




Create a React Application named “officespacerentalapp” which uses React JSX to create elements, attributes and renders DOM to display the page.

Create an element to display the heading of the page.

Attribute to display the image of the office space

Create an object of office to display the details like Name, Rent and Address.

Create a list of Object and loop through the office space item to display more data.

To apply Css, Display the color of the Rent in Red if it’s below 60000 and in Green if it’s above 60000.

**OUTPUT:**



**Hint:**





**App.js**  
  
import './App.css';

import { officeList, OfficeSpaceRentalApp } from './Components/officespaces';

function App() {

  return (

    <OfficeSpaceRentalApp offices={officeList} />

  );

}

export default App;

**Components/officespaces.js**  
  
import sr from '../Modern+office+space+lounge+area+with+black+sofa+and+yellow+chairs.webp';

const office = {

    Name: 'DBS',

    Rent: 50000,

    Address: 'Chennai'

};

export const officeList = [

    office,

    { Name: 'Krysta Consultancy', Rent: 40000, Address: 'Pune' },

    { Name: 'Hyland', Rent: 80000, Address: 'Kolkata' },

    { Name: 'Google', Rent: 200000, Address: 'Bangalore' }

];

export const OfficeSpaceRentalApp = ({ offices }) => {

    const element = "Office Space"

    const jsxatt = <img src={sr} width='25%' height='25%' alt={element} />

    return (

        <div style={{ marginLeft: '50px' }}>

            <h1>{element}, at Affordable Range </h1>

            {jsxatt}

            {

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

                    let color = [];

                    if (item.Rent <= 60000) {

                        color.push('red');

                    } else {

                        color.push('green');

                    }

                    return (

                        <div key={index}>

                            <h1> Name: {item.Name} </h1>

                            <h3 style={{ color }}> Rent: {item.Rent} </h3>

                            <h3> Address: {item.Address} </h3>

                            <hr />

                        </div>

                    )

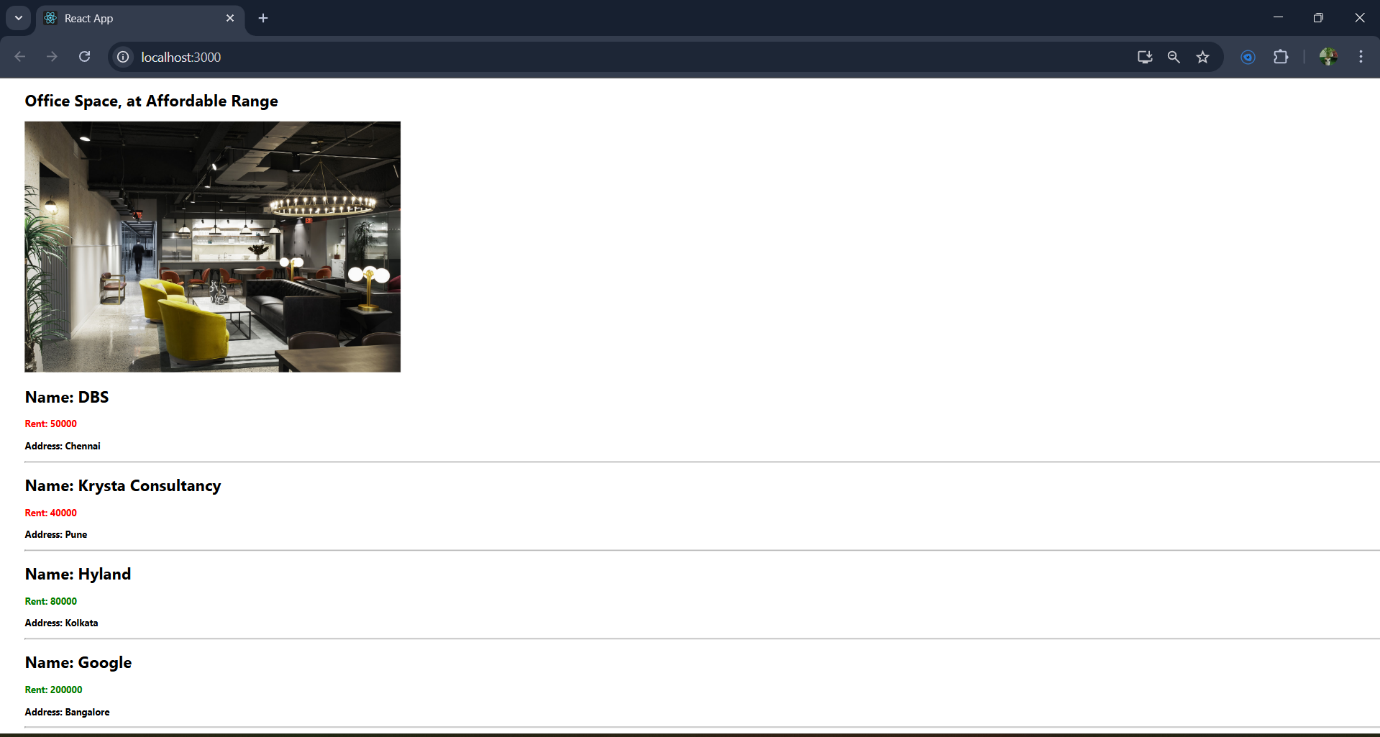
                })

            }

        </div >

    )

}

**OUTPUT**  
  


Create a React Application “eventexamplesapp” to handle various events of the form elements in HTML.

1. Create “Increment” button to increase the value of the counter and “Decrement” button to decrease the value of the counter. The “Increase” button should invoke multiple methods.
   1. To increment the value
   2. Say Hello followed by a static message.



1. Create a button “Say Welcome” which invokes the function which takes “welcome” as an argument.



1. Create a button which invokes synthetic event “OnPress” which display “I was clicked”



Create a “CurrencyConvertor” component which will convert the Indian Rupees to Euro when the Convert button is clicked.

Handle the Click event of the button to invoke the handleSubmit event and handle the conversion of the euro to rupees.



**App.js**  
  
import './App.css';

import { Count, EventButtons } from './Components/Count';

import CurrencyConverter from './Components/CurrencyConverter';

function App() {

  return (

    <div>

      <Count />

      <EventButtons />

      <CurrencyConverter />

    </div>

  );

}

export default App;

**App.css**  
  
.App {

  text-align: center;

}

.converter-container {

  width: 280px;

}

.form-row {

  display: flex;

  align-items: center;

}

.form-row label {

  width: 100px;

}

.form-row input,

.form-row textarea {

  flex: 1;

}

.App-logo {

  height: 40vmin;

  pointer-events: none;

}

@media (prefers-reduced-motion: no-preference) {

  .App-logo {

    animation: App-logo-spin infinite 20s linear;

  }

}

.App-header {

  background-color: #282c34;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: white;

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}

**Components/Count.js**  
  
import { useState } from "react"

export const Count = () => {

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

    const increment = () => {

        setCounter(count + 1);

        sayHello();

    }

    const decrement = () => {

        setCounter(count - 1);

    }

    const sayHello = () => {

        alert("Hello! Member1");

    }

    return (

        <>

            <p> {count} </p>

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

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

        </>

    );

}

export const EventButtons = () => {

    function sayWelcome(message) {

        alert(message);

    }

    const onPress = (e) => {

        console.log(e);

        alert("I was clicked");

    }

    return (

        <>

            <button onClick={() => sayWelcome("welcome")}> Say welcome </button><br />

            <button onClick={(e) => onPress(e)}> Click on me </button><br />

        </>

    );

}

**Components/CurrencyConverter.js**  
  
import { useState } from "react";

import '../App.css';

const CurrencyConverter = ({ val = '', curr = '' }) => {

    const [amount, setAmount] = useState(val);

    const [currency, setCurrency] = useState(curr);

    const eurosToRupees = 80;

    const rupeesToEuros = 1 / eurosToRupees;

    const handleSubmit = () => {

        const trimmedCurrency = currency.trim().toLowerCase();

        let result = 0;

        if (trimmedCurrency !== "euro" && trimmedCurrency !== "rupee") {

            alert("Error: Please enter either 'Euro' or 'Rupee' as the currency");

            return;

        }

        if (isNaN(parseFloat(amount))) {

            alert("Error: Please enter valid amount");

            return;

        }

        (trimmedCurrency === "euro") ? (result = parseFloat(amount) \* eurosToRupees):(result = parseFloat(amount) \* rupeesToEuros);

        alert(`Converted to ${currency} Amount is ${result}`);

    };

    return (

        <div>

            <h1 style={{ color: 'green' }}>Currency Convertor!!!</h1>

            <div className="converter-container">

                <div className="form-row">

                    <label htmlFor="amount">Amount:</label>

                    <input

                        type="number"

                        name="amount"

                        value={amount}

                        onChange={(e) => setAmount(e.target.value)}

                    />

                </div>

                <div className="form-row">

                    <label htmlFor="currency">Currency:</label>

                    <textarea

                        name="currency"

                        maxLength={5}

                        defaultValue={curr}

                        onChange={(e) => setCurrency(e.target.value)}

                    />

                </div>

                <div>

                    <button onClick={handleSubmit} style={{ marginLeft: '50%' }}>Submit</button>

                </div>

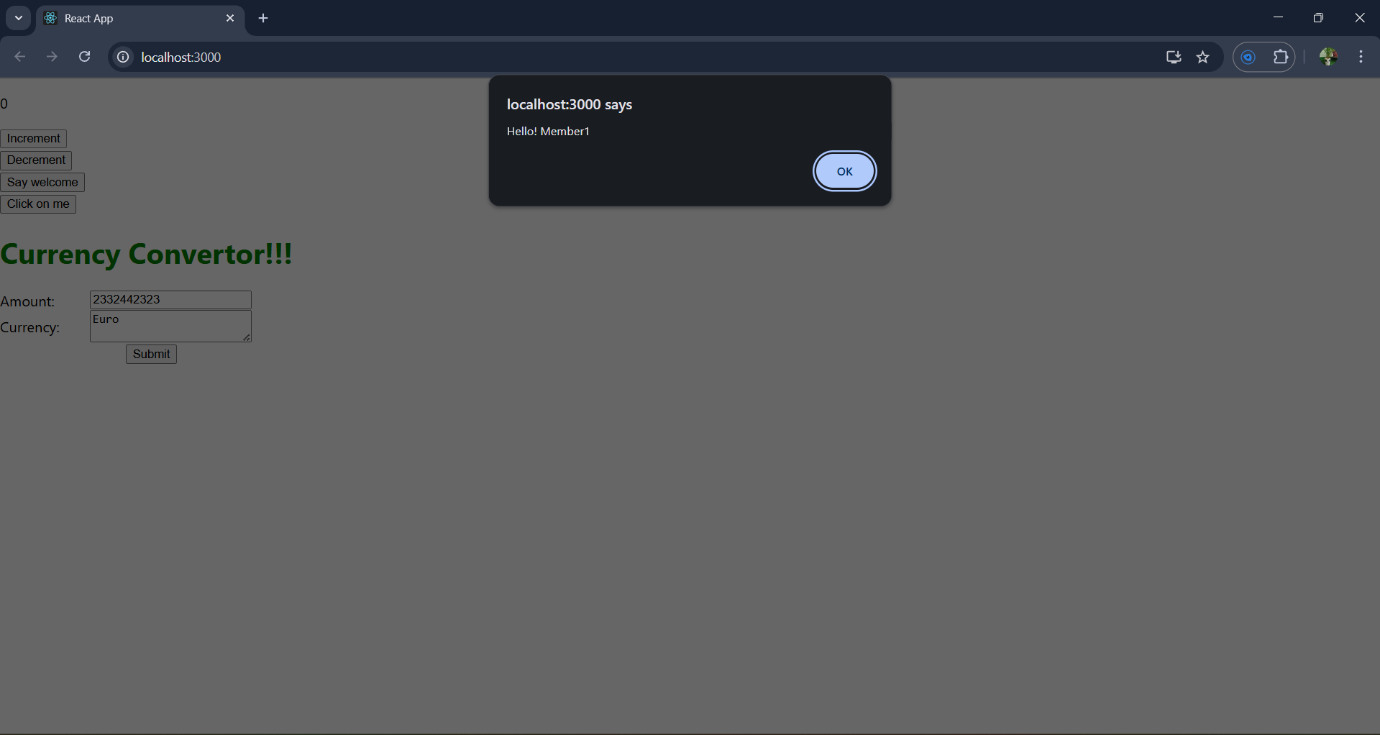
            </div>

        </div>

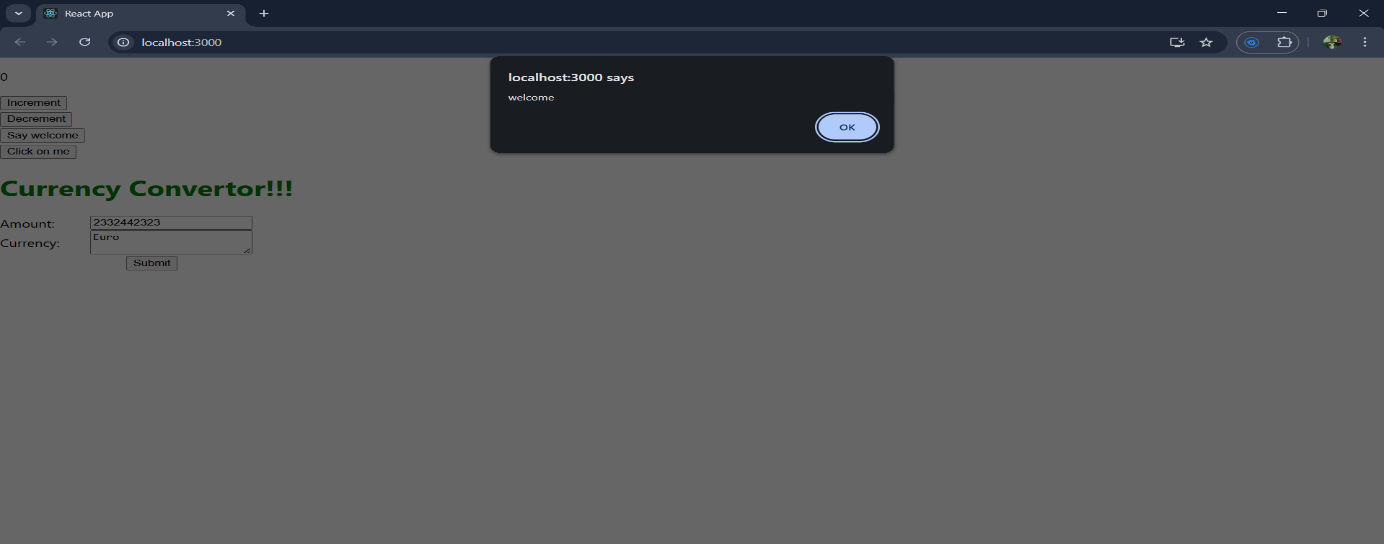
    );

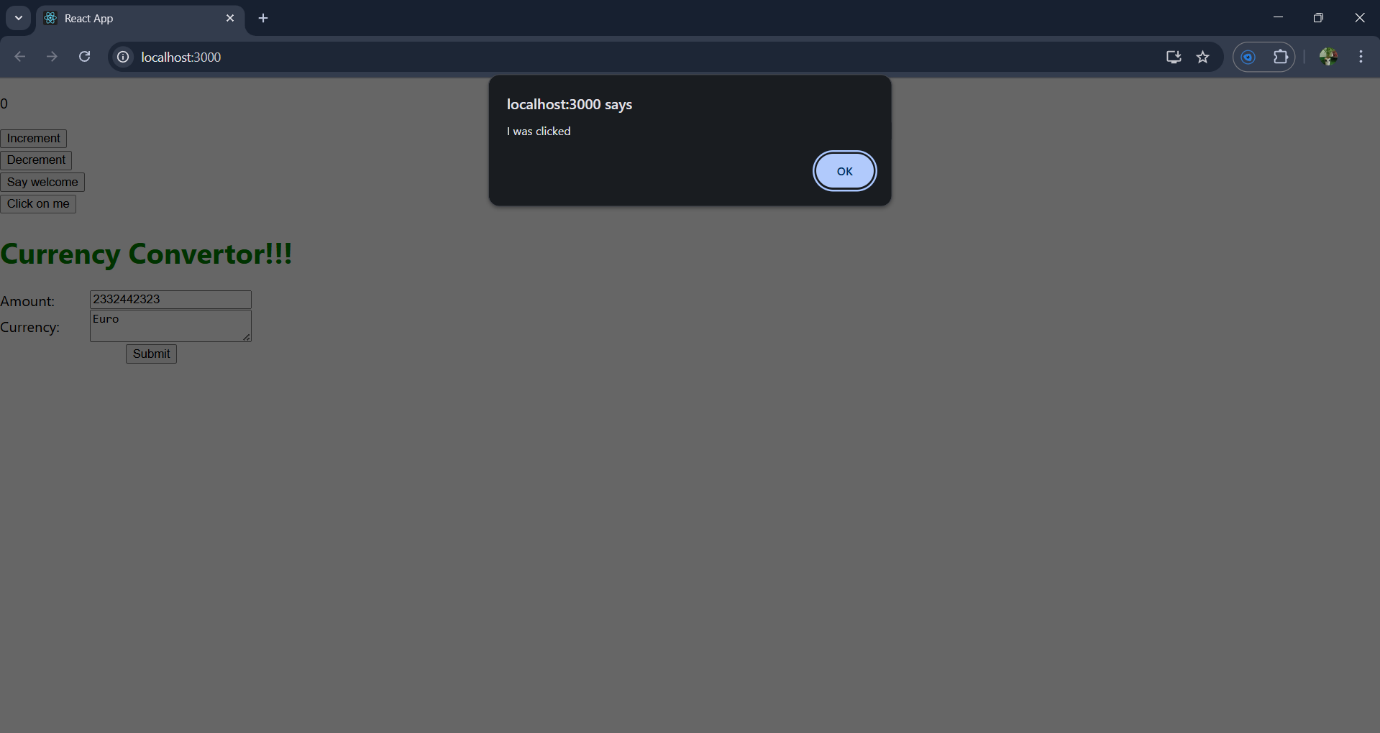
};

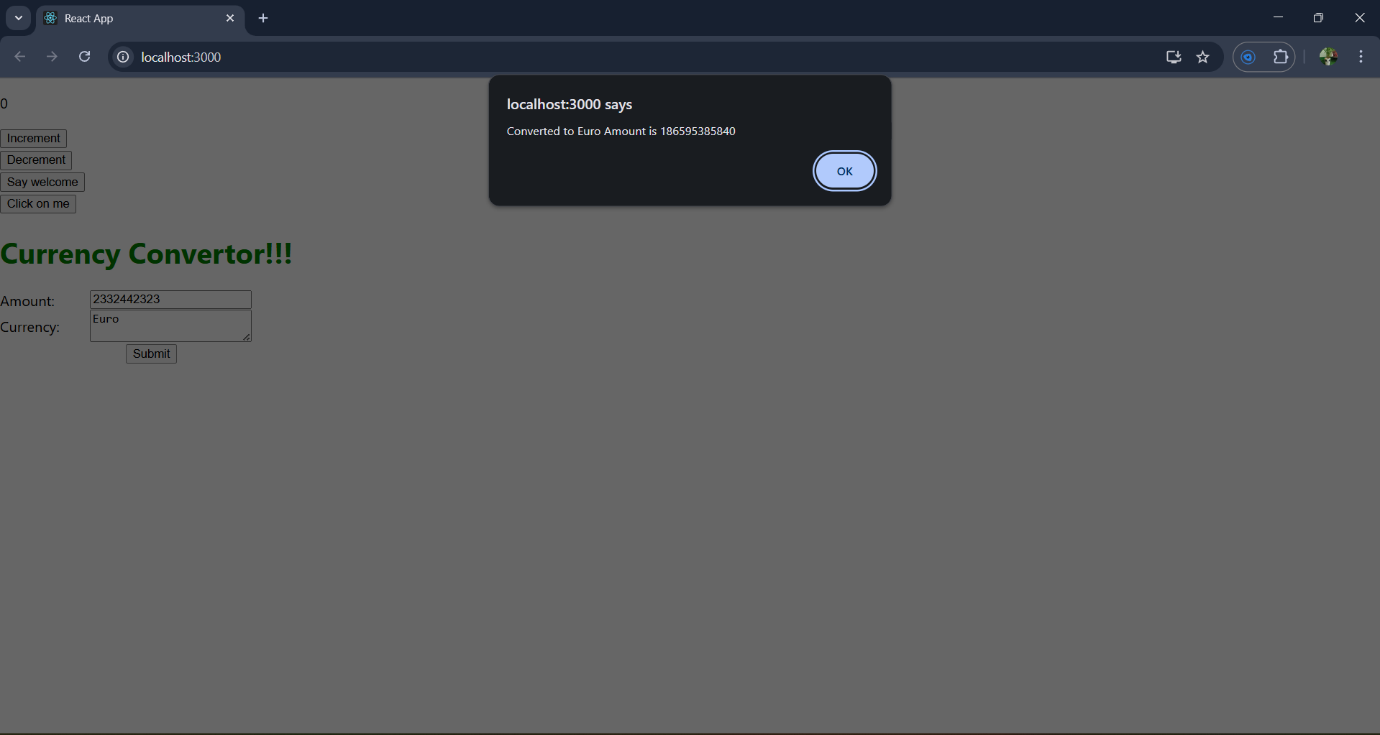
export default CurrencyConverter;

**OUTPUT**  
  










Create a React Application named “ticketbookingapp” where the guest user can browse the page where the flight details are displayed whereas the logged in user only can book tickets.

The Login and Logout buttons should accordingly display different pages. Once the user is logged in the User page should be displayed. When the user clicks on Logout, the Guest page should be displayed.





**Hint:**







**App.js**  
  
import './App.css';

import React, { useState } from 'react';

import Greeting , { LoginButton, LogoutButton } from './Components/Greetings';

function App(){

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

  const onClick = () => {

    setIsLoggedIn(prev => !prev);

  }

  return (

    <div style={{ margin: '15%' }}>

      <Greeting isLoggedIn={isLoggedIn} />

      {isLoggedIn ? (

        <LogoutButton onClick={onClick} />

      ) : (

        <LoginButton onClick={onClick} />

      )}

    </div>

  );

}

export default App;

**Components/Greetings.js**  
  
function LoginButton(props){

    return (

        <button onClick={props.onClick}>

            Login

        </button>

    );

}

function LogoutButton(props) {

    return (

        <button onClick={props.onClick}>

            Logout

        </button>

    );

}

const UserGreeting = () => {

    return <h1>Welcome back</h1>;

}

const GuestGreeting = () => {

    return <h1>Please sign up.</h1>;

}

function Greeting(props) {

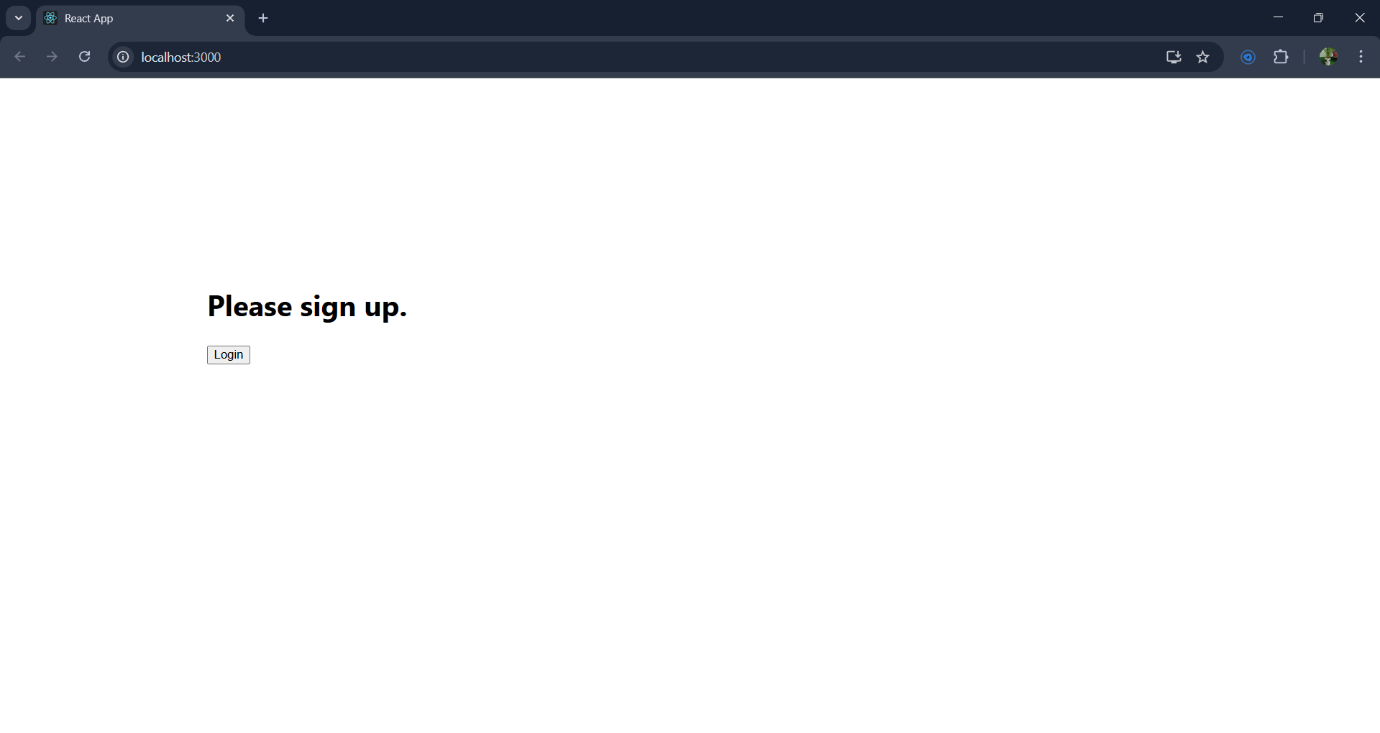
    const isLoggedIn = props.isLoggedIn;

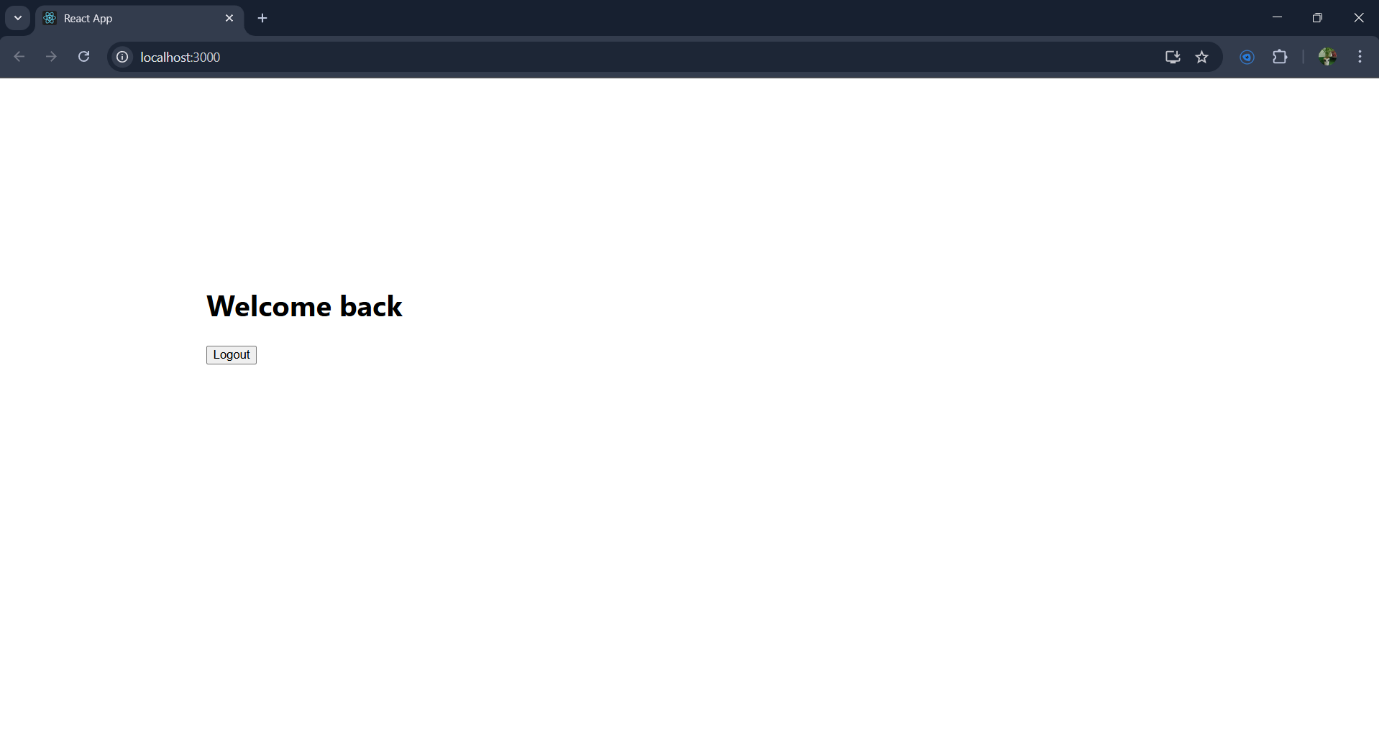
    return isLoggedIn ? <UserGreeting /> : <GuestGreeting />;

}

export { LoginButton, LogoutButton };

export default Greeting;

**OUTPUT**  
  




Create a React App named “bloggerapp” in with 3 components.

1. Book Details
2. Blog Details
3. Course Details

Implement this with as many ways possible of Conditional Rendering.



**Hint:**







**App.js**  
  
import './App.css';

import Details from './Components/Details';

import books from './Components/Book';

import blogs from './Components/Blog';

import courses from './Components/Course';

function App() {

  return (

    <Details books={books} blogs={blogs} courses={courses} />

  );

}

export default App;

**App.css**  
  
.App {

  text-align: center;

}

.details-container {

  display: flex;

  justify-content: space-between;

  padding: 20px;

  gap: 20px;

  margin-top: 15vh;

}

.details-container>div {

  flex: 1;

  padding: 10px;

}

.st2,

.mystyle1 {

  border-right: 7px solid green;

}

.App-logo {

  height: 40vmin;

  pointer-events: none;

}

@media (prefers-reduced-motion: no-preference) {

  .App-logo {

    animation: App-logo-spin infinite 20s linear;

  }

}

.App-header {

  background-color: #282c34;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: white;

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}

**Components/Book.js**  
  
const books = [

    {

        id: 101,

        bname: 'Master React',

        price: 670

    },

    {

        id: 102,

        bname: 'Deep Dive into Angular 11',

        price: 800

    },

    {

        id: 103,

        bname: 'Mongo Essentials',

        price: 450

    }

]

export default books;

**Components/Blog.js**  
  
const blogs = [

    {

        id: 101,

        topic: 'React Learning',

        author: 'Stephen Biz',

        details: 'Welcome to learning React!'

    },

    {

        id: 102,

        topic: 'Installation',

        author: 'Schewzdenier',

        details: 'You can install React from npm.'

    }

]

export default blogs;

**Components/Course.js**  
  
const courses = [

    {

        id: 1,

        cname: 'Angular',

        date: '4/5/2021'

    },

    {

        id: 2,

        cname: 'React',

        date: '6/3/20201'

    }

]

export default courses;

**Components/Details.js**  
  
import '../App.css';

const Details = (props) => {

    const bookdet = (

        !(props.books || props.books.length > 0) || (

            <ul>

                {props.books.map((book) =>

                    book ? (

                        <div key={book.id}>

                            <h3>{book.bname}</h3>

                            <h4>{book.price}</h4>

                        </div>

                    ) : null

                )}

            </ul>

        )

    );

    const coursedet = (

        props.courses && props.courses.length > 0 ? (

            <ul>

                {props.courses.map((course) =>

                    course && (

                        <div key={course.id}>

                            <h1>{course.cname}</h1>

                            <h4>{course.date}</h4>

                        </div>

                    )

                )}

            </ul>

        ) : (

            <p>No courses available.</p>

        )

    );

    const content = (

        (props.blogs && props.blogs.length > 0) && (

            <ul>

                {props.blogs.map((blog) =>

                    !blog || (

                        <div key={blog.id}>

                            <h1>{blog.topic}</h1>

                            <h4>{blog.author}</h4>

                            <p>{blog.details}</p>

                        </div>

                    )

                )}

            </ul>

        )

    );

    return (

        <div className="details-container">

            <div className="mystyle1">

                <h1>Course Details</h1>

                {coursedet}

            </div>

            <div className="st2">

                <h1>Book Details</h1>

                {bookdet}

            </div>

            <div className="v1">

                <h1>Blog Details</h1>

                {content}

            </div>

        </div>

    );

};

export default Details;

**OUTPUT**  
  
