

WEEK 7 – ASSIGNMENT

Superset ID: 6390124

React Exercises:-

Exercise 1:

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

```
return(  
  players.map((item)=>  
    {  
      return(  
        <div>  
          <li>Mr. {item.name}<span> {item.score}</span></li>  
        </div>  
      )  
    })  
  )  
)
```

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

```
return(  
  players.map((item)=>  
    {  
      if(item.score<=70)  
      {  
        players70.push(item);  
      }  
    })  
  ),  
)
```

2. IndianPlayers

- a. Display the Odd Team Player and Even Team players using the Destructuring features of ES6

```
export function OddPlayers([first,, third,, fifth]) {  
  return(  
    <div>  
      <li> First : {first}</li>  
      <li> Third : {third}</li>  
      <li> Fifth : {fifth}</li>  
    </div>))
```

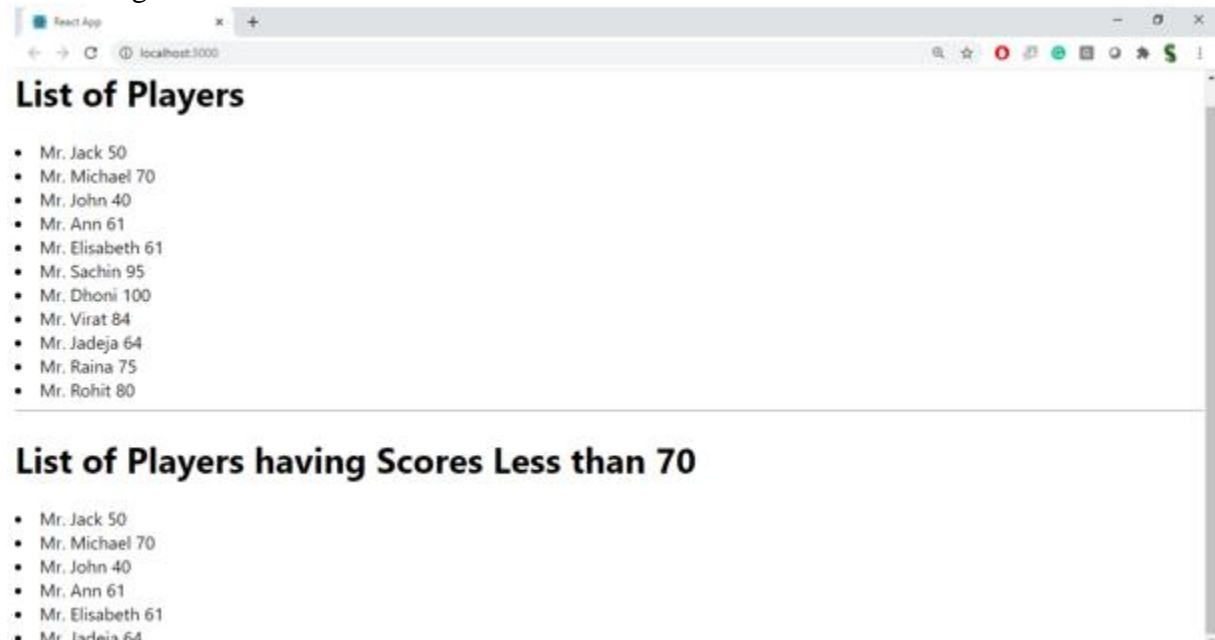
- b. Declare two arrays T20players and RanjiTrophy players and merge the two arrays and display them using the Merge feature of ES6

```
const T20Players=['First Player','Second Player','Third Player'];
const RanjiTrophyPlayers=['Fourth Player','Fifth Player','Sixth Player'];
export const IndianPlayers=[...T20Players, ...RanjiTrophyPlayers]
```

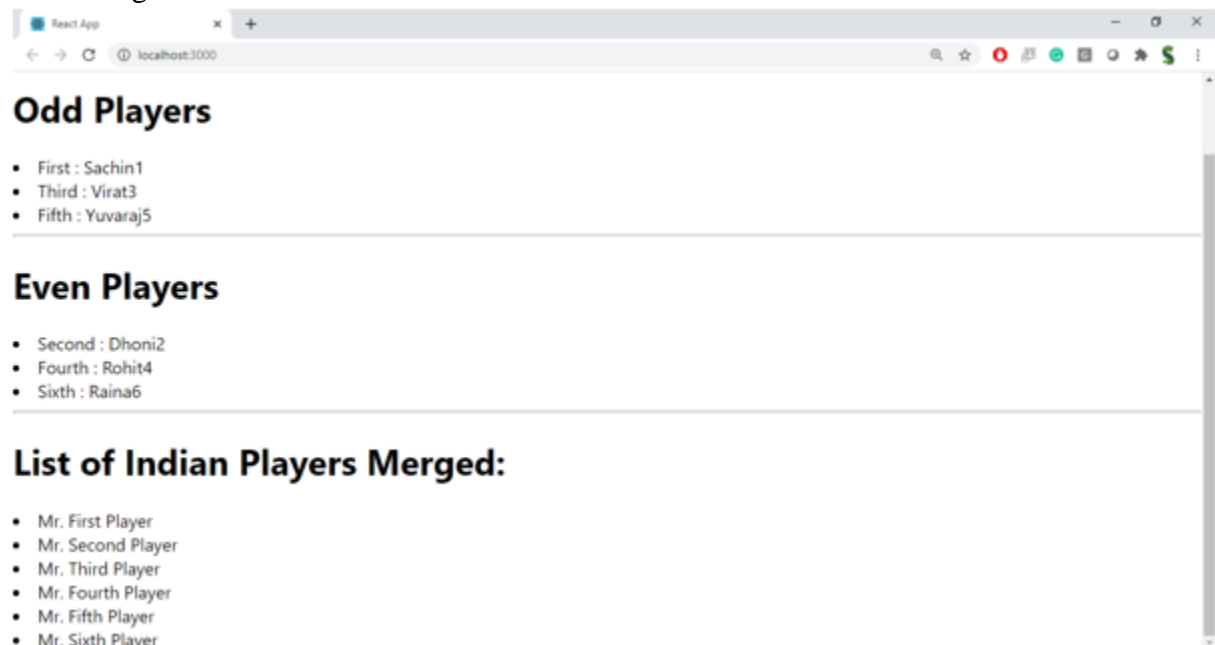
Display these two components in the same home page using a simple if else in the flag variable.

Output:

When Flag=true



When Flag=false



Hint:

```
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>
  )
}
```

src/ ListofPlayers.js

```
import React from 'react';

const players = [
  { name: 'Virat Kohli', score: 82 },
  { name: 'Rohit Sharma', score: 45 },
  { name: 'Shubman Gill', score: 73 },
  { name: 'Suryakumar Yadav', score: 68 },
  { name: 'Hardik Pandya', score: 51 },
  { name: 'Ravindra Jadeja', score: 37 },
  { name: 'Rishabh Pant', score: 59 },
  { name: 'KL Rahul', score: 88 },
  { name: 'Mohammed Shami', score: 33 },
```

```

    { name: 'Jasprit Bumrah', score: 41 },
    { name: 'Yuzvendra Chahal', score: 21 }
  ];

const ListofPlayers = () => {
  return (
    <div>
      <h2>All Players</h2>
      <ul>
        {players.map((player, index) => (
          <li key={index}>{player.name} - {player.score}</li>
        ))}
      </ul>

      <h2>Players with Score < 70</h2>
      <ul>
        {players.filter(p => p.score < 70).map((player, index) => (
          <li key={index}>{player.name} - {player.score}</li>
        ))}
      </ul>
    </div>
  );
};

export default ListofPlayers;

```

src/ IndianPlayers.js

```

import React from 'react';

const T20players = ['Ishan Kishan', 'Ruturaj Gaikwad', 'Rahul Tripathi'];
const RanjiTrophyPlayers = ['Baba Indraji', 'Rajat Patidar', 'Priyank Panchal'];

// Merge arrays
const allPlayers = [...T20players, ...RanjiTrophyPlayers];

const IndianPlayers = () => {
  const oddTeam = [];
  const evenTeam = [];

  allPlayers.forEach((player, index) => {
    if ((index + 1) % 2 === 0) {
      evenTeam.push(player);
    } else {
      oddTeam.push(player);
    }
  });

  return (
    <div>
      <h2>All Indian Players</h2>
      <ul>
        {allPlayers.map((player, i) => <li key={i}>{player}</li>)}
      </ul>

      <h3>Odd Team Players</h3>
    </div>
  );
};

```

```

        <ul>
          {oddTeam.map((player, i) => <li key={i}>{player}</li>)}
        </ul>

        <h3>Even Team Players</h3>
        <ul>
          {evenTeam.map((player, i) => <li key={i}>{player}</li>)}
        </ul>
      </div>
    );
  };

export default IndianPlayers;

```

src/ App.css

```

.App {
  text-align: center;
}
.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);
  }
}

.player-list {
  text-align: left;
  margin-left: 2rem;
}

```

src/ App.js

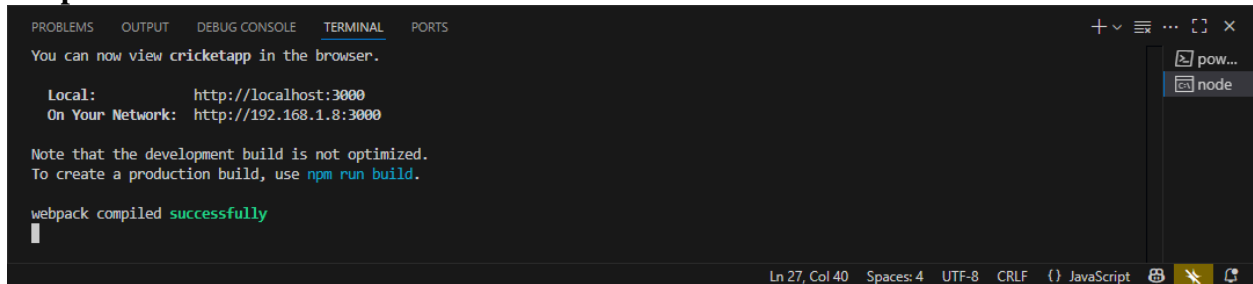
```
import React from 'react';
import './App.css';
import ListofPlayers from './ListofPlayers';
import IndianPlayers from './IndianPlayers';

function App() {
  const flag = true; // Change to false to test IndianPlayers component

  return (
    <div className="App">
      <h1>Welcome to Cricket App</h1>
      {flag ? <ListofPlayers /> : <IndianPlayers />}
    </div>
  );
}

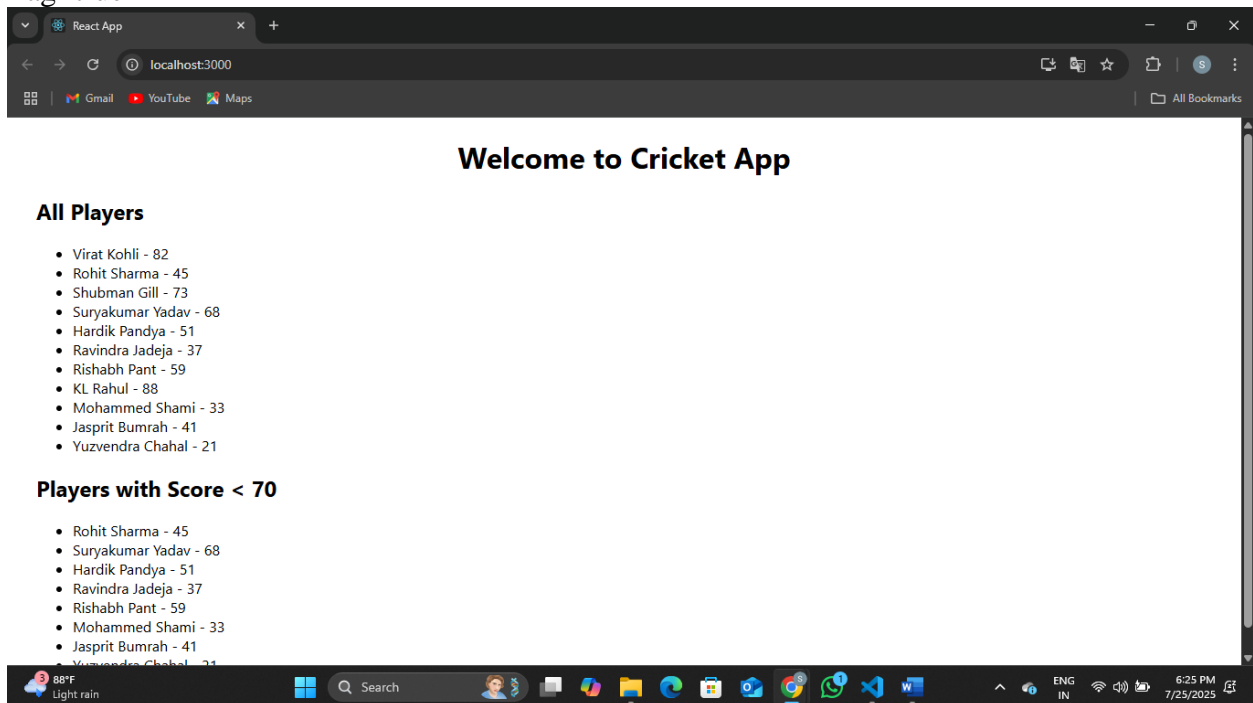
export default App;
```

Output:

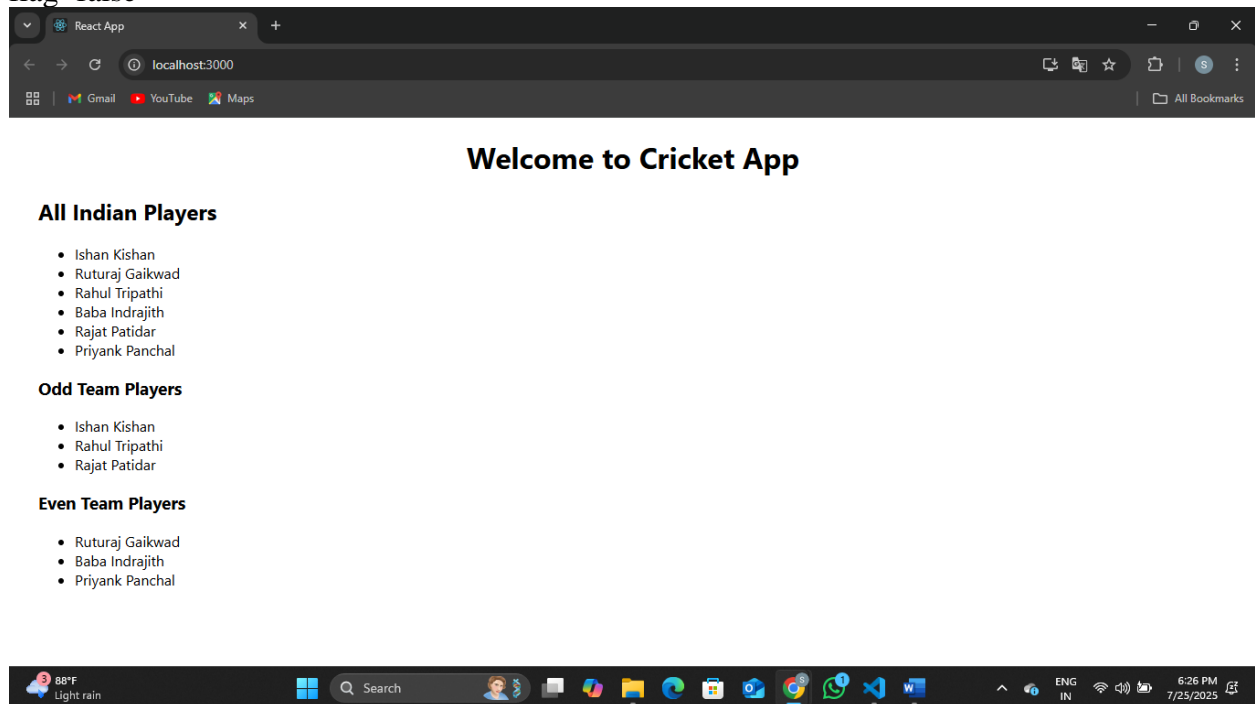


<http://localhost:3000/>

flag=true



flag=false



Exercise 2:

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.

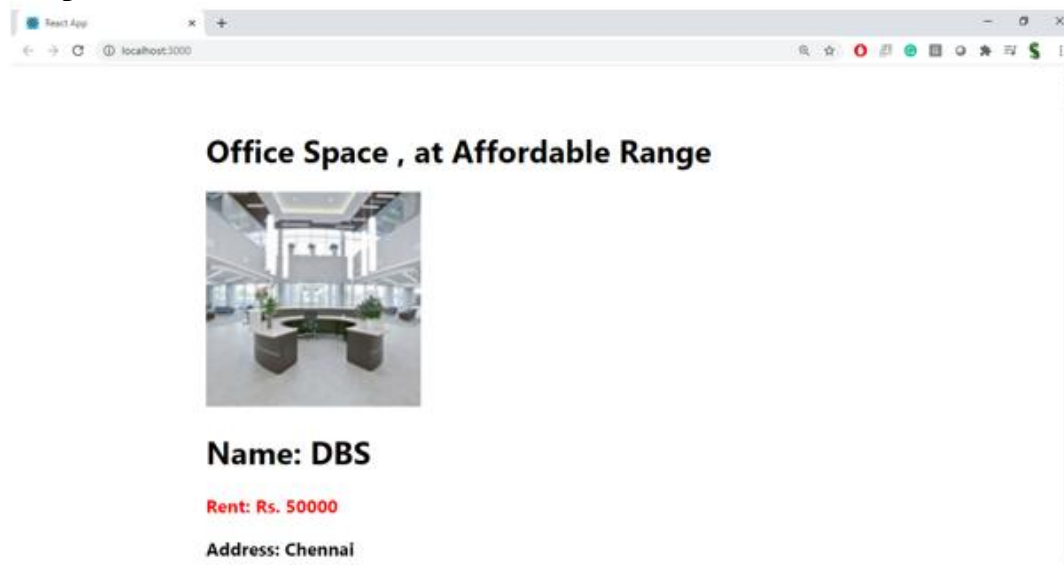
Attribute to display the image of the office space

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:

```
{
  let colors=[];
  if(ItemName.Rent<=60000)
  {
    colors.push("textRed");
  }
  else{
    colors.push("textGreen");
  }
}

const element="Office Space"
const jsxatt=<img src={sr} width="25%" height="25%" alt="Office Space"/>
const ItemName={Name:"DBS", Rent: 50000, Address:'Chennai'}
<h1>{element} , at Affordable Range </h1>
{jsxatt}
<h1>Name: {ItemName.Name}</h1>
<h3> Rent: Rs. {ItemName.Rent}</h3>
<h3> Address: {ItemName.Address}</h3>
```

src/ App.js

```
import React from 'react';
import './App.css';

function App() {
  // JSX Element - Heading
  const heading = <h1 style={{ textAlign: 'center' }}>Office Space Rental
  Portal</h1>;

  // JSX Attribute - Image
  const imageUrl = 'https://images.unsplash.com/photo-1560448204-
  e02f11c3d0e2'; // sample image

  // Office Object
  const office = {
    name: 'Premium Workspace',
    rent: 55000,
    address: '14th Floor, IT Park, Bangalore'
  };

  // Office List
  const officeList = [
    { name: 'Startup Bay', rent: 45000, address: 'MG Road, Chennai' },
    { name: 'Elite Hub', rent: 70000, address: 'Sector 62, Noida' },
    { name: 'CoWork Zone', rent: 60000, address: 'Koregaon Park, Pune' },
    { name: 'BizNest', rent: 80000, address: 'Salt Lake, Kolkata' },
    { name: 'Space360', rent: 30000, address: 'Anna Nagar, Chennai' }
  ];

  // Style Function
  const getRentStyle = (rent) => {
    return {
```



```

        color: rent < 60000 ? 'red' : 'green',
        fontWeight: 'bold'
    };
};

return (
    <div className="App" style={{ padding: '20px' }}>
        {heading}

        <img src={imageUrl} alt="Office Space" style={{ width: '60%', height:
'auto', margin: '20px auto', display: 'block', borderRadius: '8px' }} />

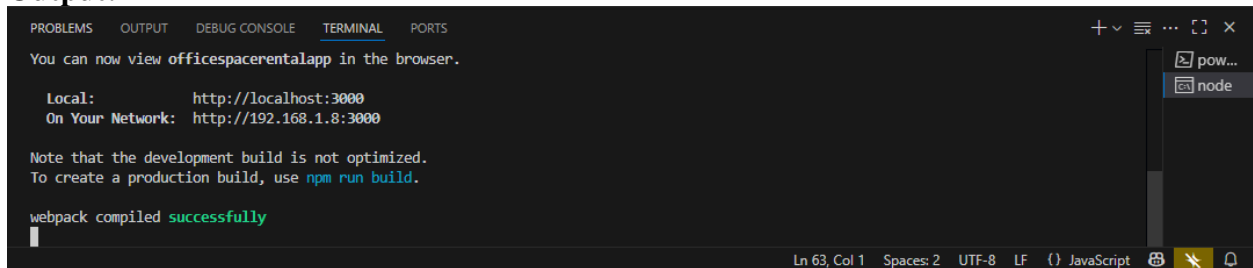
        <h2>Featured Office</h2>
        <div style={{ marginBottom: '20px' }}>
            <p>Name: {office.name}</p>
            <p style={getRentStyle(office.rent)}>Rent: ₹{office.rent}</p>
            <p>Address: {office.address}</p>
        </div>

        <h2>Available Spaces</h2>
        <ul style={{ listStyle: 'none', padding: 0 }}>
            {officeList.map((item, index) => (
                <li key={index} style={{ marginBottom: '15px', borderBottom: '1px
solid #ccc', paddingBottom: '10px' }}>
                    <p><strong>Name:</strong> {item.name}</p>
                    <p style={getRentStyle(item.rent)}><strong>Rent:</strong>
₹{item.rent}</p>
                    <p><strong>Address:</strong> {item.address}</p>
                </li>
            ))}
        </ul>
    </div>
);
}

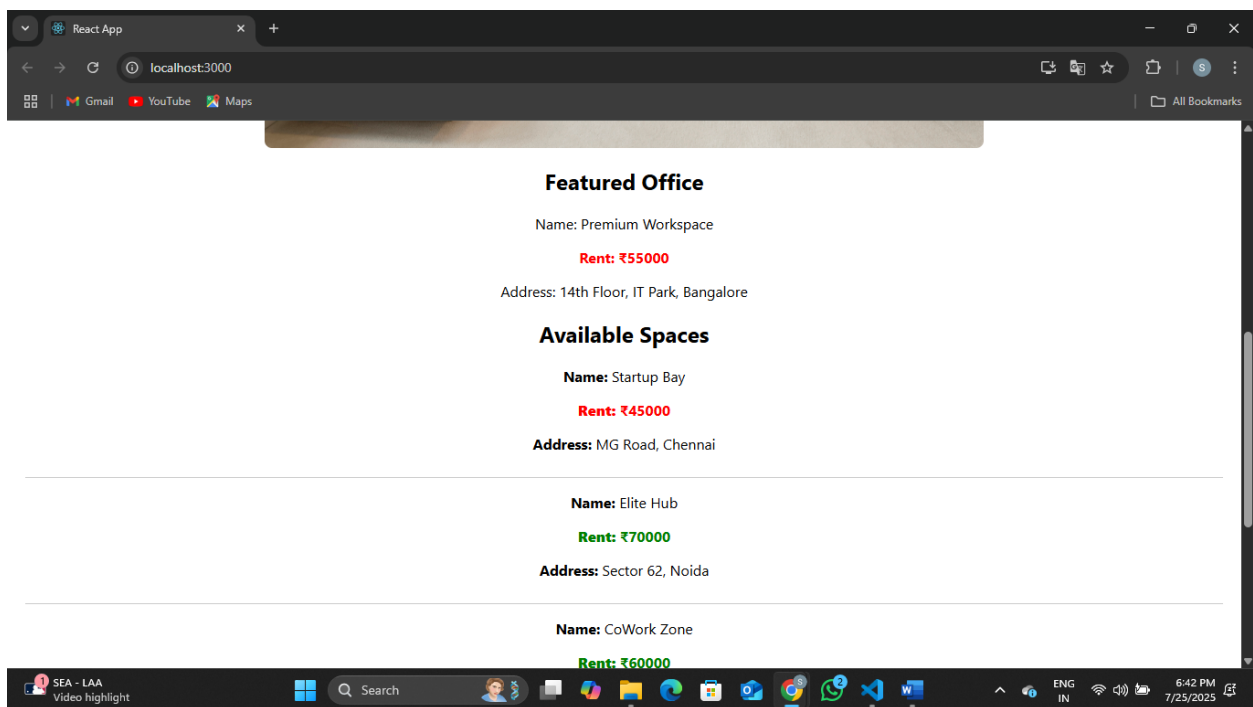
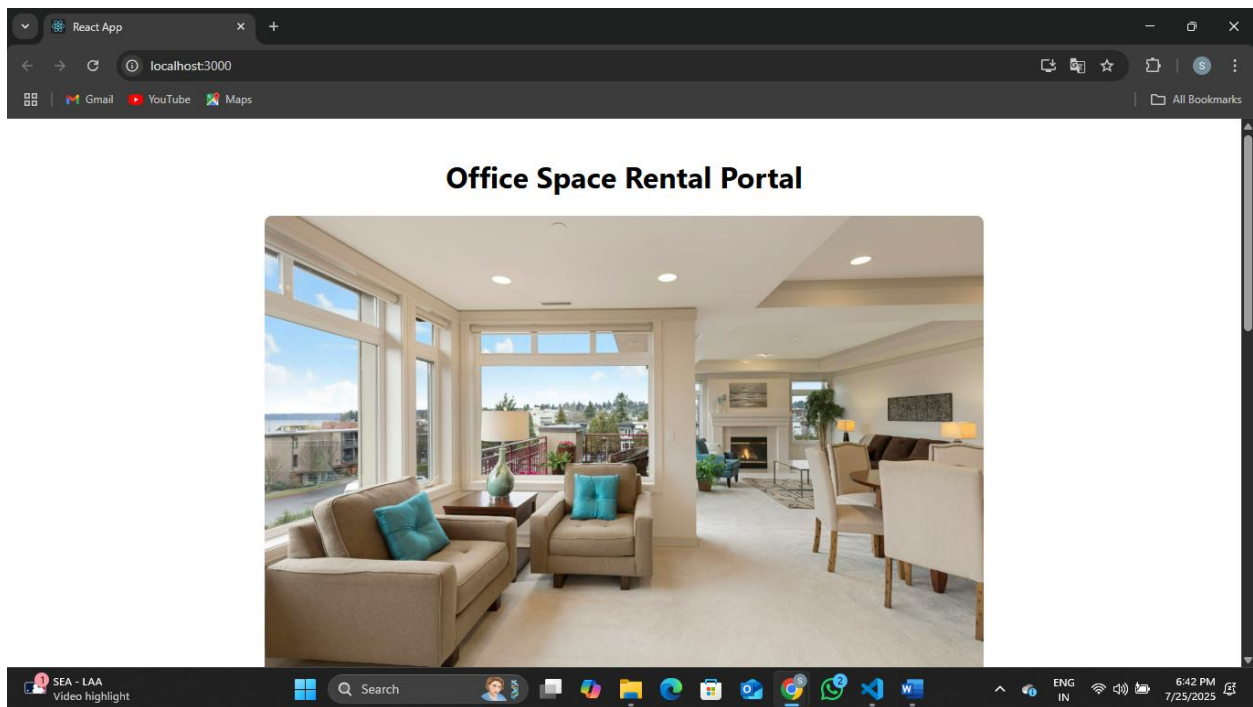
export default App;

```

Output:



<http://localhost:3000/>

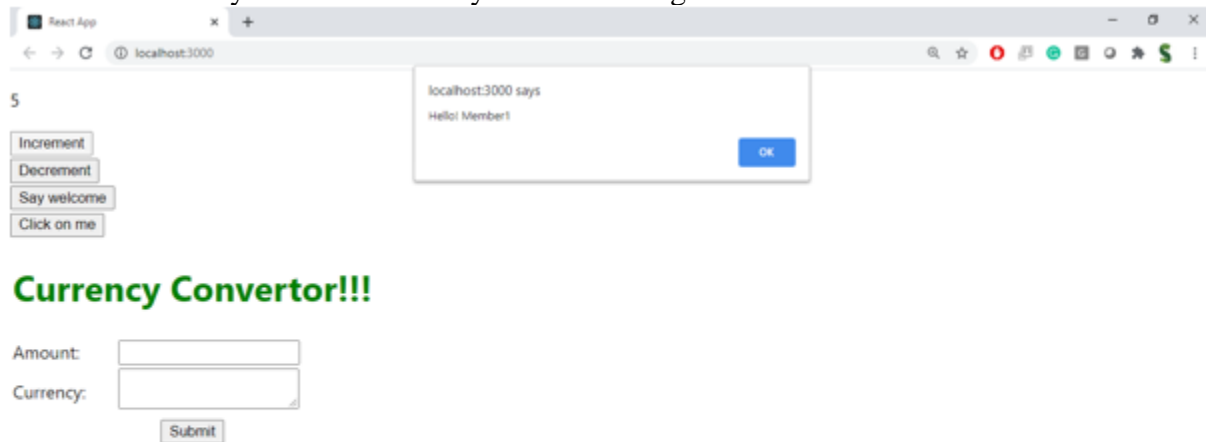


Exercise 3:

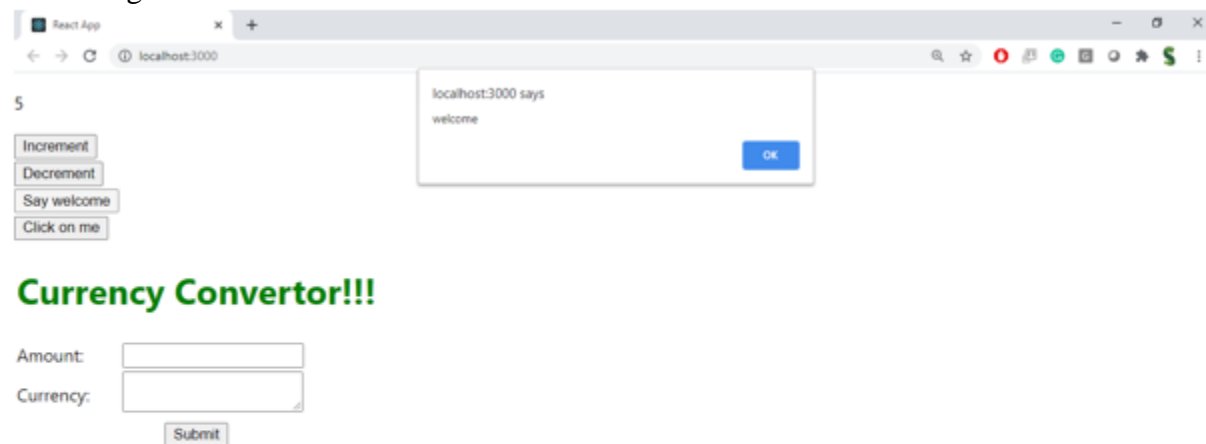
Create a React Application “eventexamplesapp” to handle various events of the form elements in HTML.

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.
 - a. To increment the value

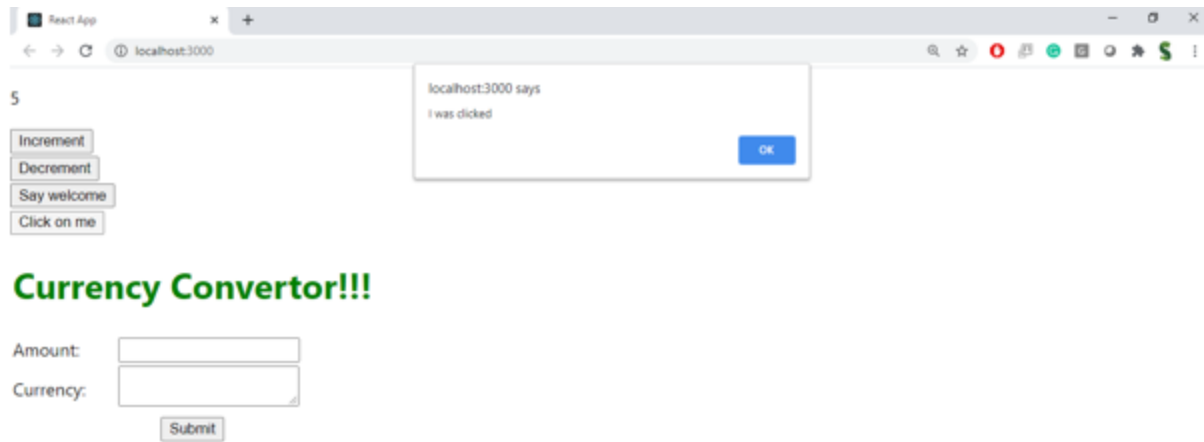
b. Say Hello followed by a static message.



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

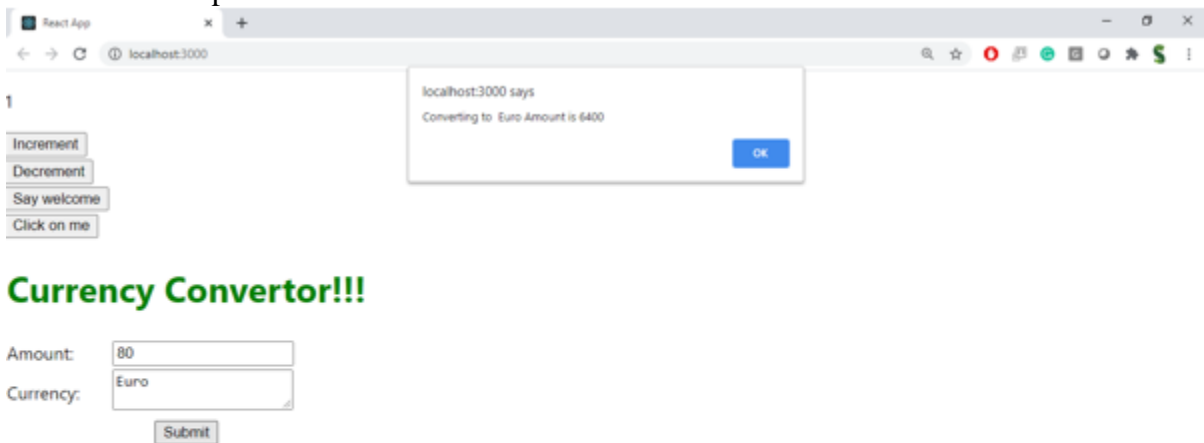


3. Create a button which invokes synthetic event “OnPress” which display “I was clicked”



Create a “CurrencyConverter” 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.



src/ CounterEvents.js

```
import React, { Component } from 'react';

class CounterEvents extends Component {
  constructor(props) {
    super(props);
    this.state = {
      count: 0,
      message: ''
    }
  }
}
```

```

    };
  }

  increment = () => {
    this.setState({ count: this.state.count + 1 });
    this.sayHello();
    this.sayStaticMessage();
  };

  decrement = () => {
    this.setState({ count: this.state.count - 1 });
  };

  sayHello = () => {
    console.log('Hello from the button!');
  };

  sayStaticMessage = () => {
    console.log('This is a static message after increment.');
```

```

  };

  sayWelcome = (msg) => {
    this.setState({ message: msg });
  };

  handleClick = (event) => {
    alert('I was clicked (Synthetic Event)');
    console.log('Synthetic Event Object:', event);
  };

  render() {
    return (
      <div style={{ textAlign: 'center', padding: 20 }}>
        <h2>Counter: {this.state.count}</h2>
        <button onClick={this.increment}>Increment</button>{' '}
        <button onClick={this.decrement}>Decrement</button>

        <br /><br />
        <button onClick={() => this.sayWelcome('Welcome to React Events!')}>
          Say Welcome
        </button>

        <p>{this.state.message}</p>

        <br />
        <button onClick={this.handleClick}>Synthetic OnPress</button>
      </div>
    );
  }
}

export default CounterEvents;
```

src/ CurrencyConverter.js

```
import React, { useState } from 'react';
```

```

function CurrencyConverter() {
  const [rupees, setRupees] = useState('');
  const [euros, setEuros] = useState('');

  const handleSubmit = (e) => {
    e.preventDefault();
    const euroValue = (parseFloat(rupees) / 90).toFixed(2); // Example rate:
    ₹90 = €1
    setEuros(euroValue);
  };

  return (
    <div style={{ textAlign: 'center', marginTop: 40 }}>
      <h2>Currency Converter (INR → EUR)</h2>
      <form onSubmit={handleSubmit}>
        <input
          type="number"
          value={rupees}
          onChange={(e) => setRupees(e.target.value)}
          placeholder="Enter INR"
          required
        />
        <button type="submit">Convert</button>
      </form>
      {euros} && <h3>€ {euros}</h3>
    </div>
  );
}

export default CurrencyConverter;

```

src/ App.js

```

import React from 'react';
import './App.css';
import CounterEvents from './CounterEvents';
import CurrencyConverter from './CurrencyConverter';

function App() {
  return (
    <div className="App">
      <h1>React Events & Currency App</h1>
      <CounterEvents />
      <CurrencyConverter />
    </div>
  );
}

export default App;

```

Output:

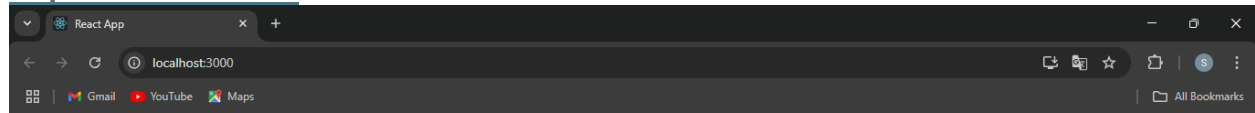
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
You can now view eventexamplesapp in the browser.

Local:      http://localhost:3000
On Your Network: http://192.168.1.8:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

<http://localhost:3000/>



React Events & Currency App

Counter: 0

Increment Decrement

Say Welcome

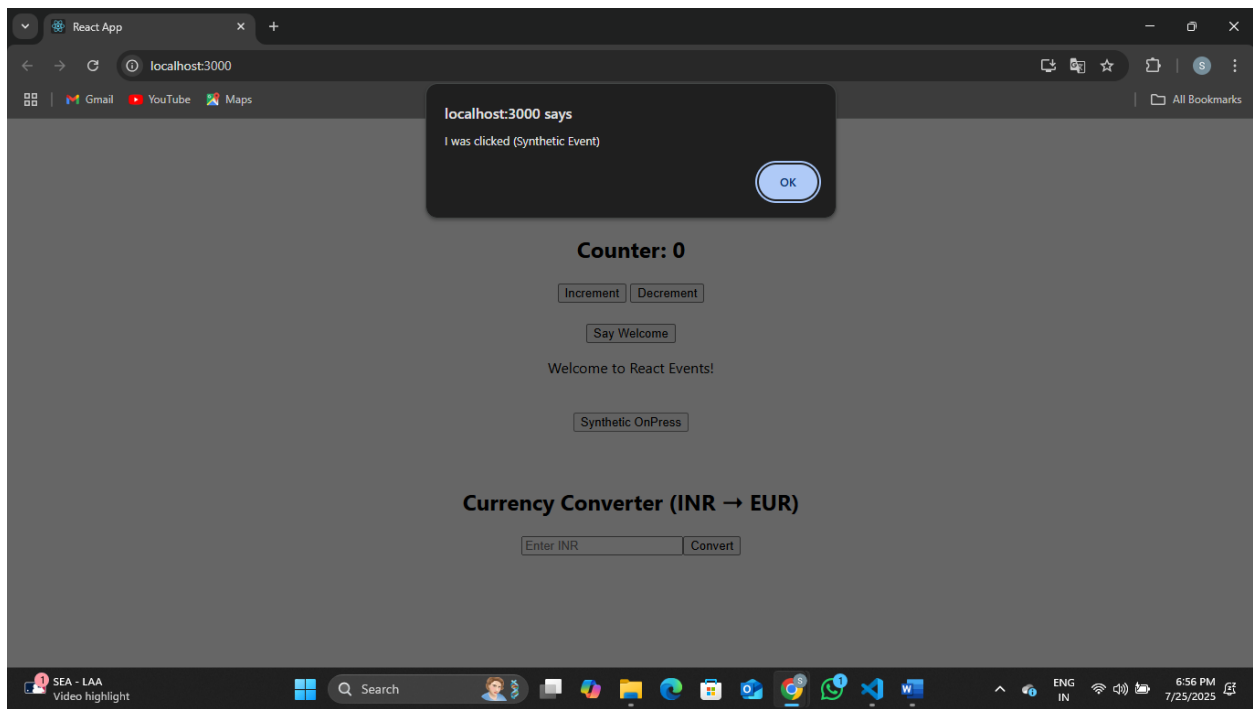
Welcome to React Events!

Synthetic OnPress

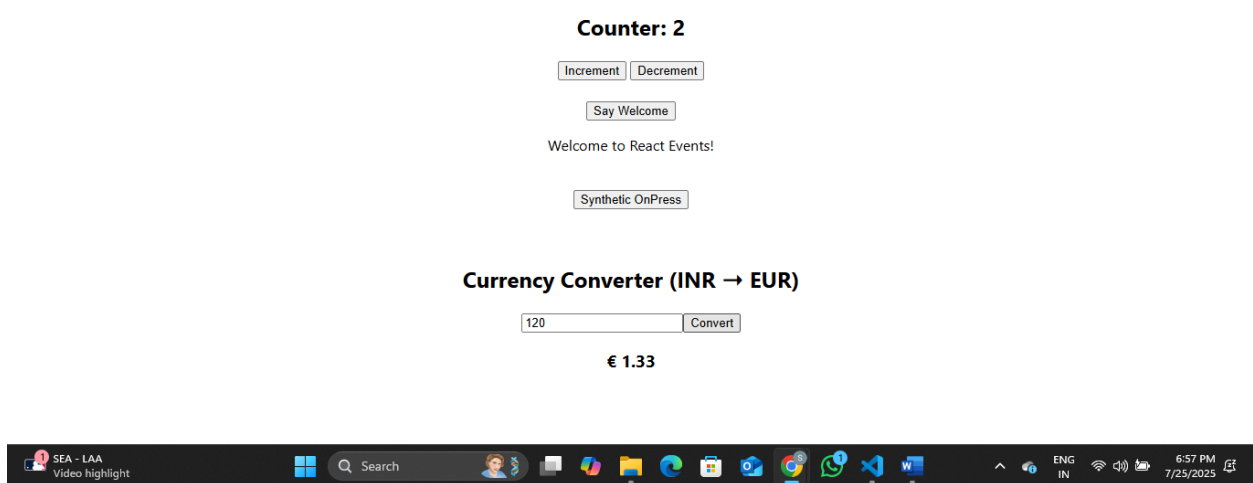
Currency Converter (INR → EUR)

Enter INR Convert



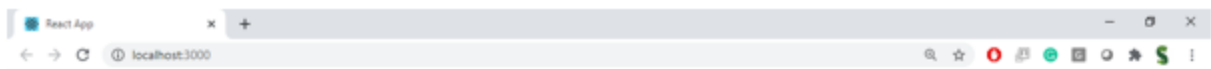
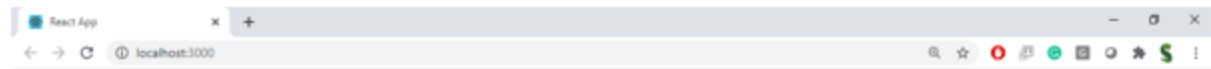


React Events & Currency App



Exercise 4:

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:

```
function LoginButton(props) {  
  return (  
    <button onClick={props.onClick}>  
      Login  
    </button>  
  );  
}
```

```
function LogoutButton(props) {
  return (
    <button onClick={props.onClick}>
      Logout
    </button>
  );
}
```

```
function Greeting(props) {
  const isLoggedIn = props.isLoggedIn;
  if (isLoggedIn) {
    return <UserGreeting />;
  }
  return <GuestGreeting />;
}
```

src/ Guest.js

```
import React from 'react';

function Guest() {
  return (
    <div>
      <h2>Welcome Guest</h2>
      <p>You can browse available flights but need to login to book
tickets.</p>
    </div>
  );
}

export default Guest;
```

src/ User.js

```
import React from 'react';

function User() {
  return (
    <div>
      <h2>Welcome User</h2>
      <p>You are logged in. You can now book tickets.</p>
    </div>
  );
}

export default User;
```

src/ App.js

```
import React, { useState } from 'react';
import './App.css';
import Guest from './Guest';
import User from './User';

function App() {
```

```

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

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

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

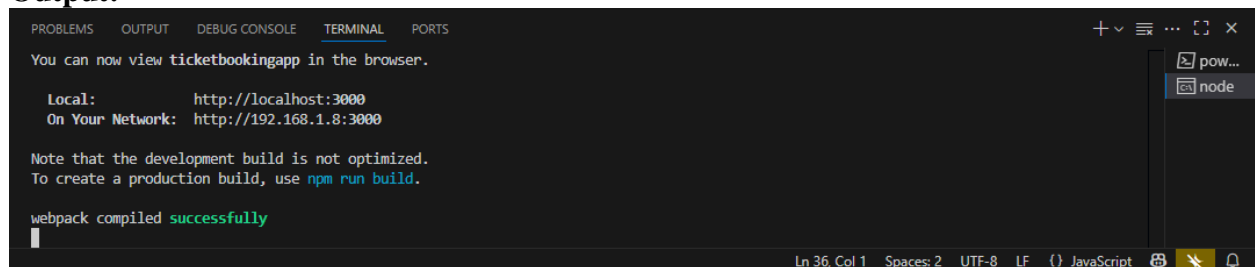
// Element variable example
let button;
if (isLoggedIn) {
  button = <button onClick={handleLogout}>Logout</button>;
} else {
  button = <button onClick={handleLogin}>Login</button>;
}

return (
  <div className="App" style={{ textAlign: 'center', marginTop: '50px' }}>
    <h1>Ticket Booking Portal</h1>
    {button}
    <hr />
    {isLoggedIn ? <User /> : <Guest />}
  </div>
);
}

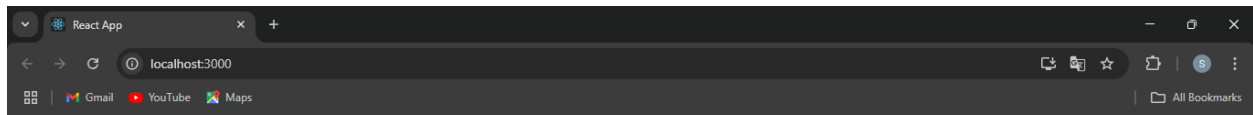
export default App;

```

Output:



<http://localhost:3000/>



Ticket Booking Portal

Login

Welcome Guest

You can browse available flights but need to login to book tickets.



Ticket Booking Portal

Logout

Welcome User

You are logged in. You can now book tickets.

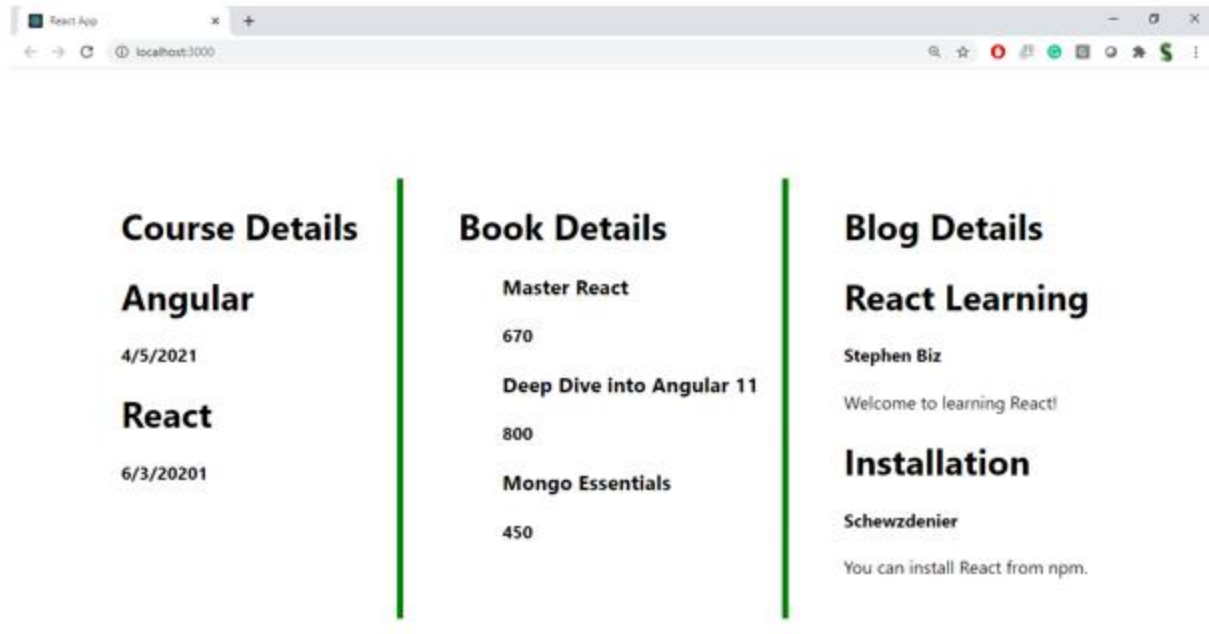


Exercise 5:

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:

```
const bookdet = (  
  <ul>  
    {props.books.map((book) =>  
      <div key={book.id}>  
        <h3> {book.bname}</h3>  
        <h4>{book.price}</h4>  
      </div>  
    )}  
  </ul>  
);
```

```
return (  
  <div>  
    <div>  
      <div className="st2">  
        <h1> Book Details</h1>  
        {bookdet}  
      </div>  
      <div className="v1">  
        <h1> Blog Details</h1>  
        {content}  
      </div>  
      <div className="mystyle1">  
        <h1> Course Details</h1>  
        {coursedet}  
      </div>  
    </div>  
  </div>  
</div>  
);  
}
```

```
export 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},
];
```

src/ BookDetails.js

```
import React from 'react';

function BookDetails({ books }) {
  return (
    <div style={{ textAlign: 'left', marginLeft: '2rem' }}>
      <h2>Book Details</h2>
      <ul>
        {books.map((book, index) => (
          <li key={index}>
            <strong>{book.title}</strong> by {book.author}
          </li>
        ))}
      </ul>
    </div>
  );
}

export default BookDetails;
```

src/ BlogDetails.js

```
import React from 'react';

function BlogDetails({ blogs }) {
  return (
    <div style={{ textAlign: 'left', marginLeft: '2rem' }}>
      <h2>Blog Posts</h2>
      <ul>
        {blogs.map((blog) => (
          <li key={blog.id}>
            <strong>{blog.title}</strong> - {blog.category}
          </li>
        ))}
      </ul>
    </div>
  );
}

export default BlogDetails;
```

src/ CourseDetails.js

```
import React from 'react';

function CourseDetails({ courses }) {
  return (
    <div style={{ textAlign: 'left', marginLeft: '2rem' }}>
      <h2>Course List</h2>
```

```

        <ul>
          {courses.map((course, i) => (
            <li key={i}>
              {course.name} - ₹{course.price}
            </li>
          ))}
        </ul>
      </div>
    );
  }
}

```

```
export default CourseDetails;
```

src/ App.js

```

import React, { useState } from 'react';
import './App.css';
import BookDetails from './BookDetails';
import BlogDetails from './BlogDetails';
import CourseDetails from './CourseDetails';

function App() {
  const [view, setView] = useState("books");

  const books = [
    { title: "You Can Win", author: "Shiv Khera" },
    { title: "Wings of Fire", author: "A.P.J. Abdul Kalam" }
  ];

  const blogs = [
    { id: 1, title: "React Basics", category: "Frontend" },
    { id: 2, title: "REST APIs", category: "Backend" }
  ];

  const courses = [
    { name: "ReactJS", price: 3999 },
    { name: "NodeJS", price: 2999 }
  ];

  let content;

  // Conditional rendering using if-else
  if (view === "books") {
    content = <BookDetails books={books} />;
  } else if (view === "blogs") {
    content = <BlogDetails blogs={blogs} />;
  } else if (view === "courses") {
    content = <CourseDetails courses={courses} />;
  } else {
    content = <p>Please select a view</p>;
  }

  return (
    <div className="App">
      <h1>Blogger Dashboard</h1>
      <button onClick={() => setView("books")}>Books</button>
      <button onClick={() => setView("blogs")}>Blogs</button>
    </div>
  );
}

```

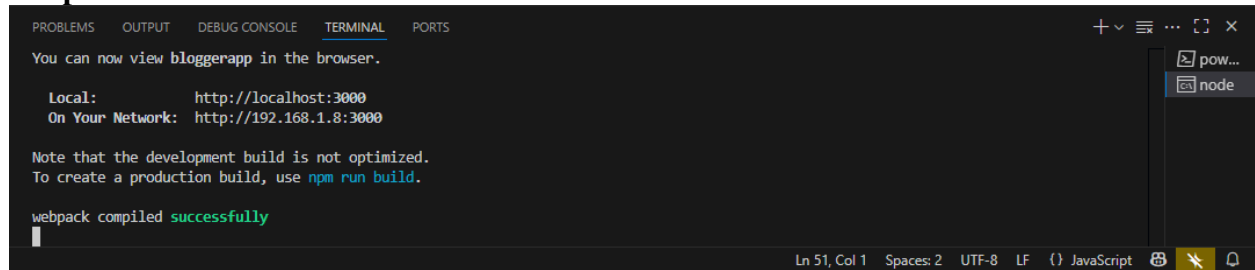
```

        <button onClick={() => setView("courses")}>Courses</button>
        <hr />
        {content}
      </div>
    );
  }
}

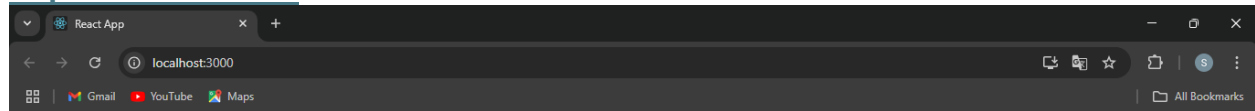
```

export default App;

Output:



<http://localhost:3000/>



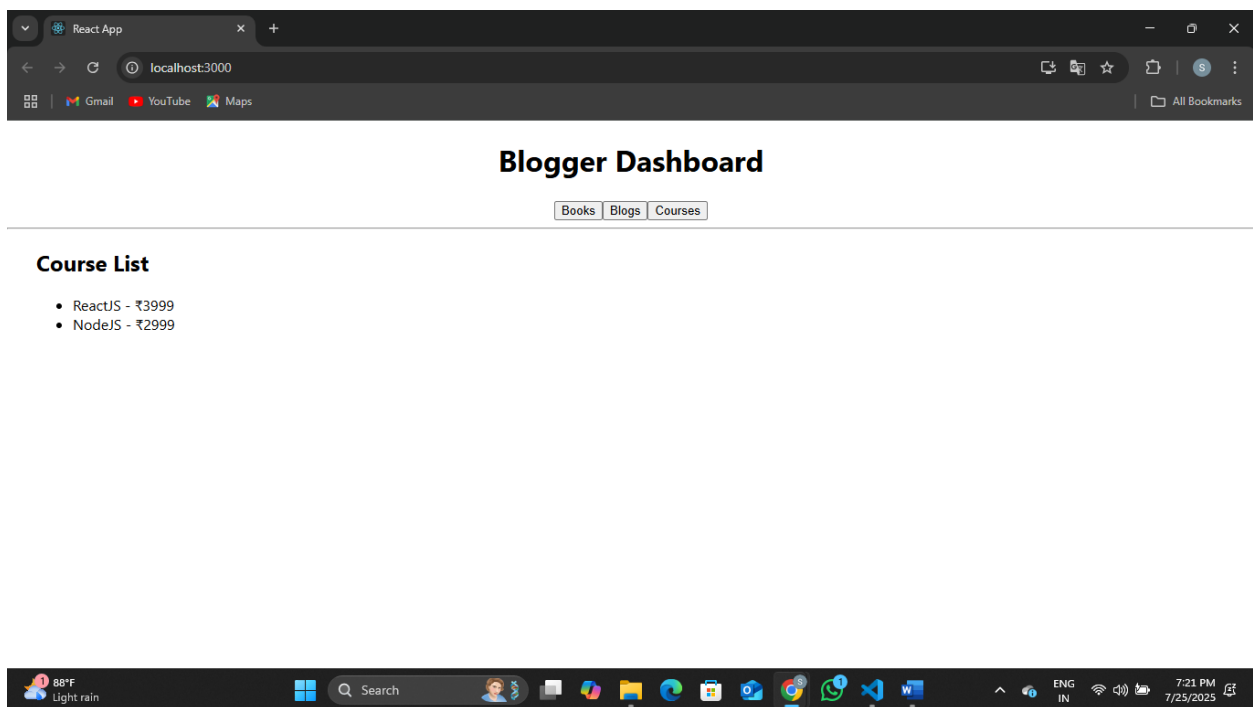
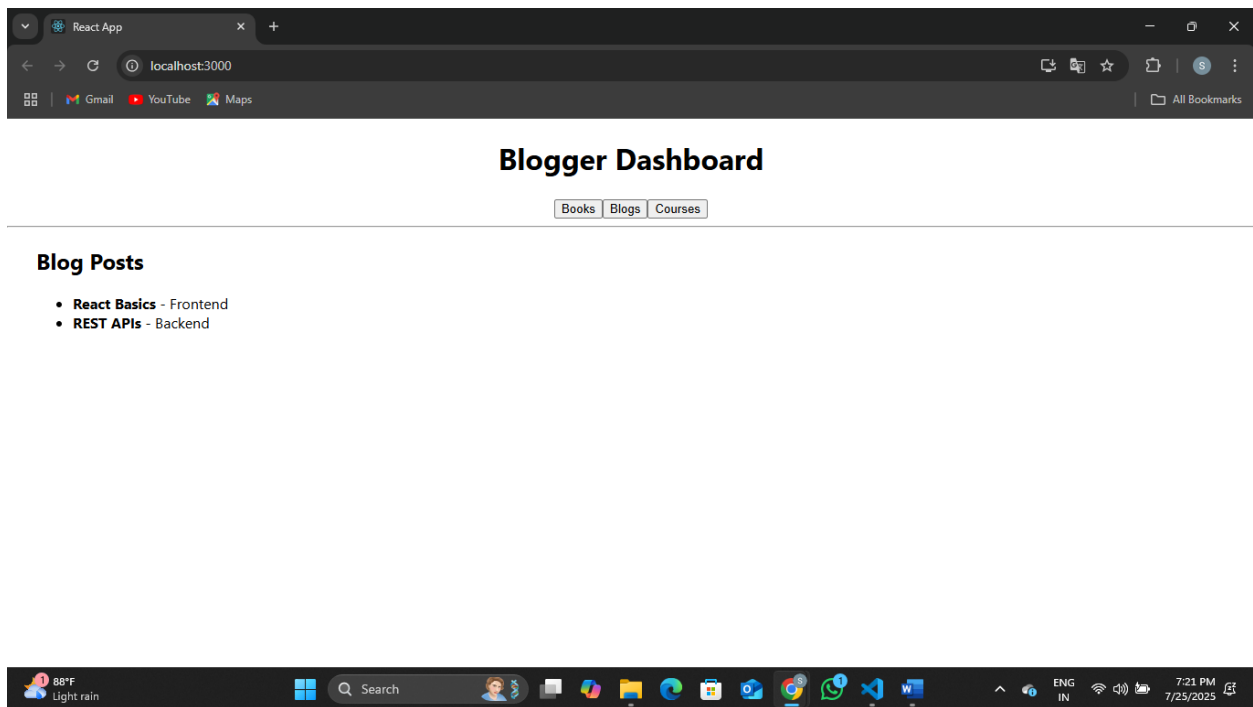
Blogger Dashboard

Books Blogs Courses

Book Details

- **You Can Win** by Shiv Kherra
- **Wings of Fire** by A.P.J. Abdul Kalam





Exercise 6:

Developers of Apps Centric Solutions have created an employee management application which supports light and dark themes for the buttons. The current solution uses the react state and props to provide the theme name to be used from App component to Employee List component and from there to Employee Card component. Quality assurance team analyzed the solutions and found the technique being used to be a substandard one. React architect suggested to use the

react context API to share the theme name with nested child components instead of passing them down using props from the parent component.

You are assigned the task of converting the application from props only to React Context API.

Application can be downloaded from below



employeesapp.zip

1. Unzip the application and open it using VS Code
2. Go to terminal and execute `npm install` command to restore all the node modules

```
TERMINAL  PROBLEMS  OUTPUT  DEBUG CONSOLE

C:\CTS-NewHandsOns\ReactHandsOns\employeesapp>npm install
```

Figure 1: Restore node modules

3. Run the application once to see the output. Use `npm start` command.

```
TERMINAL  PROBLEMS  OUTPUT  DEBUG CONSOLE

C:\CTS-NewHandsOns\ReactHandsOns\employeesapp>npm start
```

Figure 2: Starting application

4. Explore the components present in **App.js**, **EmployeesList.js** and **EmployeeCard.js** files.
5. Create a new file with the name as **ThemeContext.js**. Define a new context in the file with the name as **ThemeContext** and assign it a default value of 'light' and export it as default from the module.
6. Open App component present in **App.js** file.
 - a. Import the **ThemeContext** in App component.
 - b. Define the theme context provider to be the entire JSX of the App component.
 - c. Assign the value for the theme provider from the state of the component.
 - d. Modify the call to **EmployeeList** component so that theme name is no longer passed as props.
7. Go to **EmployeeList** component present in **EmployeeList.js** file and modify it so that theme name is not passed explicitly to its child component.
8. Go to **EmployeeCard** component inside **EmployeeCard.js** file
 - a. Import the **ThemeContext** into the component file
 - b. Retrieve the value of the context with the help of **useContext()** and store it in a variable
 - c. Use the variable to pass the **className** for the buttons.

src/ ThemeContext.js

```
import React from 'react';

const ThemeContext = React.createContext('light'); // default is 'light'

export default ThemeContext;
```

src/ App.js

```
import React, { useState } from 'react';
import EmployeesList from './EmployeesList';
import ThemeContext from './ThemeContext';

function App() {
  const [theme, setTheme] = useState('light');

  return (
    <ThemeContext.Provider value={theme}>
      <div className="App">
        <h1>Employee Management</h1>
        <button onClick={() => setTheme(theme === 'light' ? 'dark' :
'light')}>
          Toggle Theme
        </button>
        <EmployeesList /> {/* Removed theme prop */}
      </div>
    </ThemeContext.Provider>
  );
}

export default App;
```

src/ EmployeesList.js

```
import React from 'react';
import EmployeeCard from './EmployeeCard';

function EmployeesList() {
  const employees = [
    { id: 1, name: 'John Doe', role: 'Developer' },
    { id: 2, name: 'Jane Smith', role: 'Designer' },
    { id: 3, name: 'Sam Wilson', role: 'Tester' }
  ];

  return (
    <div>
      <h2>Employee List</h2>
      {employees.map(emp => (
        <EmployeeCard key={emp.id} employee={emp} />
      ))}
    </div>
  );
}

export default EmployeesList;
```

src/ EmployeeCard.js

```
import React, { useContext } from 'react';
import ThemeContext from './ThemeContext';

function EmployeeCard({ employee }) {
  const theme = useContext(ThemeContext);

  return (
    <div className="employee-card" style={{
```

```

        margin: '1rem',
        padding: '1rem',
        border: '1px solid gray',
        borderRadius: '6px'
    })>
    <p><strong>Name:</strong> {employee.name}</p>
    <p><strong>Role:</strong> {employee.role}</p>
    <button className={`btn-${theme}`}>
      {theme === 'light' ? 'Light Mode' : 'Dark Mode'}
    </button>
  </div>
);
}

export default EmployeeCard;

```

src/ App.css

```

a {
  display: inline-block;
  margin: 3px 5px;
  padding: 5px 5px;
  border-radius: 3px;
  text-decoration: none;
}

a.dark {
  background-color: #333;
  color: #eee;
}

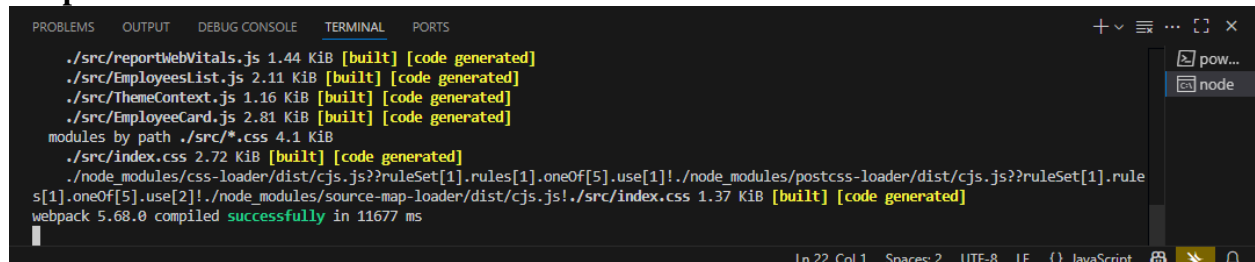
a.light {
  background-color: #eee;
  color: #333;
}

.btn-light {
  background-color: white;
  color: black;
  border: 1px solid black;
}

.btn-dark {
  background-color: black;
  color: white;
  border: 1px solid white;
}

```

Output:

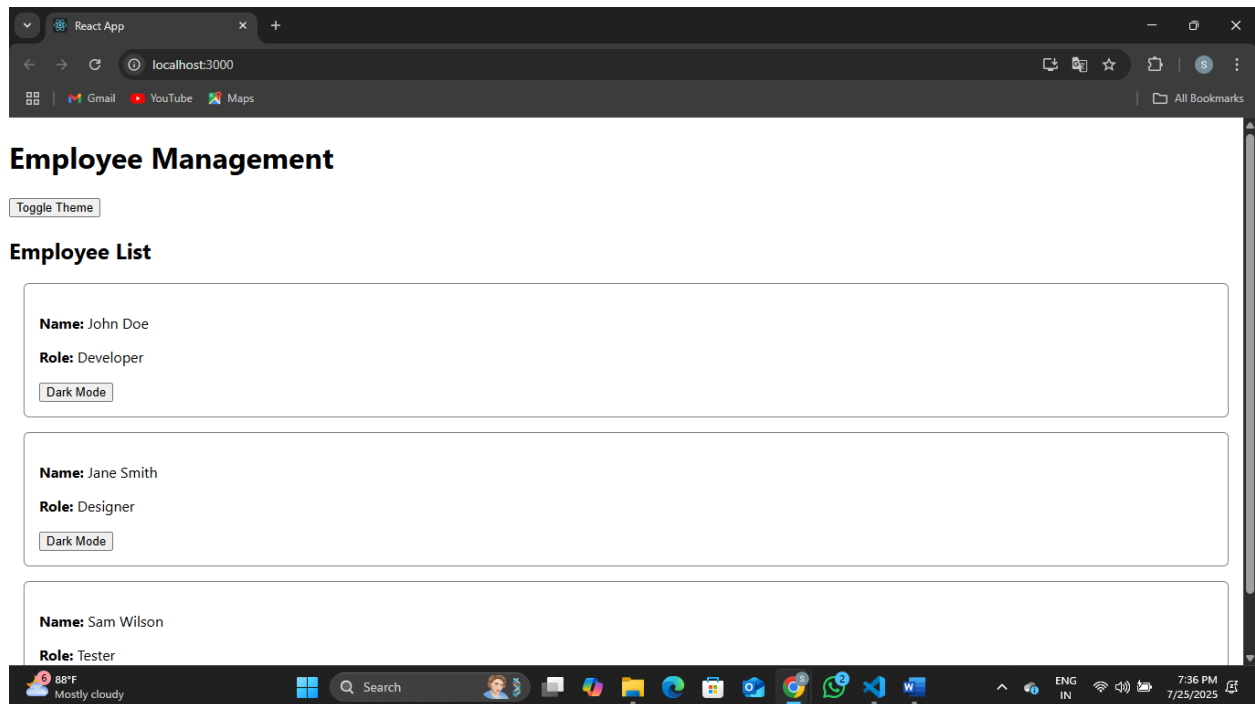
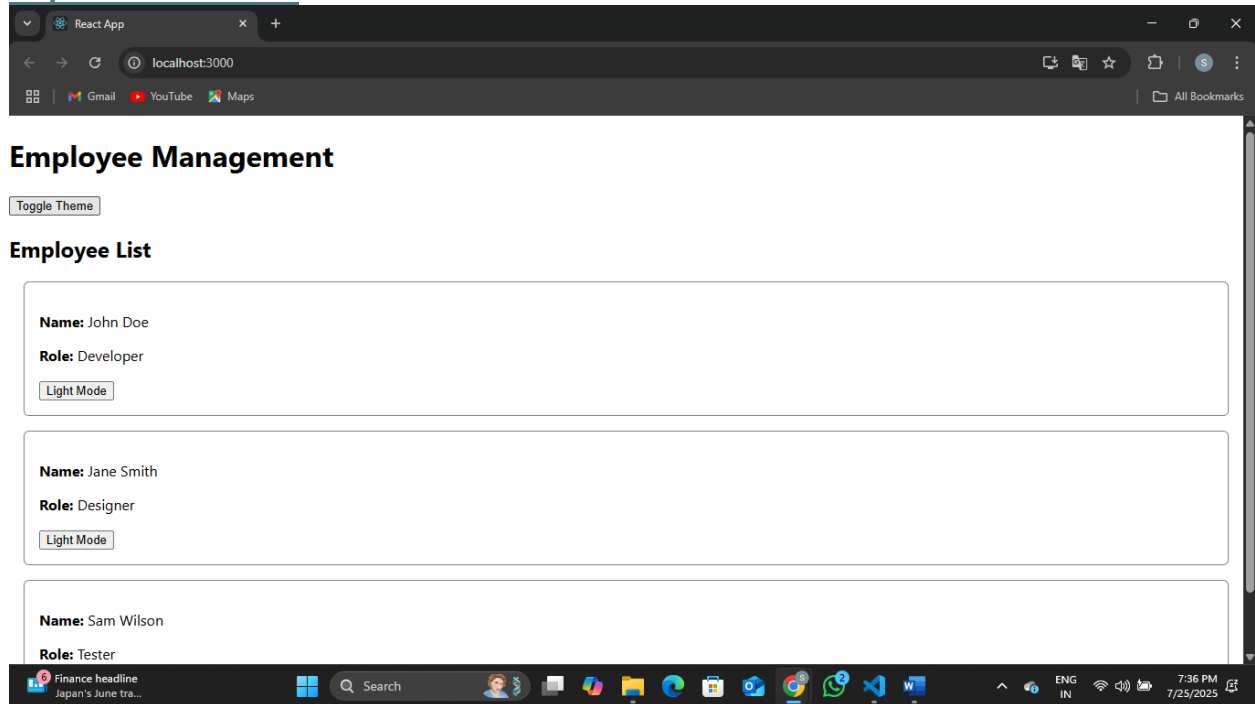


```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
./src/reportWebVitals.js 1.44 KiB [built] [code generated]
./src/EmployeesList.js 2.11 KiB [built] [code generated]
./src/ThemeContext.js 1.16 KiB [built] [code generated]
./src/EmployeeCard.js 2.81 KiB [built] [code generated]
modules by path ./src/*.css 4.1 KiB
./src/index.css 2.72 KiB [built] [code generated]
./node_modules/css-loader/dist/cjs.js??ruleSet[1].rules[1].oneOf[5].use[1]!./node_modules/postcss-loader/dist/cjs.js??ruleSet[1].rule
s[1].oneOf[5].use[2]!./node_modules/source-map-loader/dist/cjs.js!./src/index.css 1.37 KiB [built] [code generated]
webpack 5.68.0 compiled successfully in 11677 ms

```

<http://localhost:3000/>

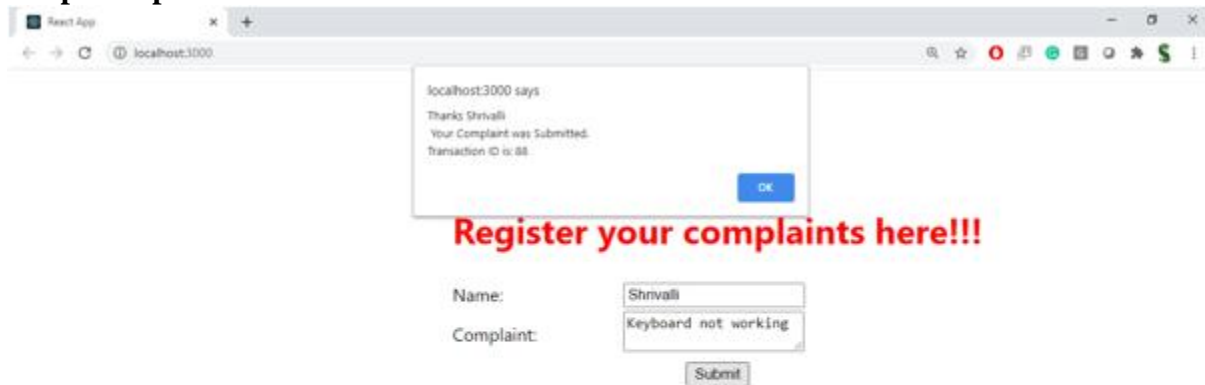


Exercise 7:

Create a React App named “ticketraisingapp” which will help to raise a complaint and get it resolved.

Create a component named “ComplaintRegister” with a form containing a textbox to enter the employee name and a textarea to enter the complaint. Use “handleSubmit” event of the button to submit the complaint and generate a Reference number for further follow ups in the alert box.

Output Expected:



Hint:

```
handleChange(event) {  
  this.setState({ [event.target.name] : event.target.value});  
}  
  
handleSubmit(event) {  
  var msg='Thanks ' + this.state.ename + ' \n Your Complaint was Submitted  
ID is: ' + this.state.NumberHolder;  
  alert(msg);  
  event.preventDefault();  
}
```

src/ ComplaintRegister.js

```
import React, { useState } from 'react';  
  
function ComplaintRegister() {  
  const [employeeName, setEmployeeName] = useState('');  
  const [complaint, setComplaint] = useState('');  
  
  const handleSubmit = (e) => {  
    e.preventDefault();  
  
    if (employeeName.trim() === '' || complaint.trim() === '') {  
      alert('Please fill out both fields.');      return;  
    }  
  }  
}
```

```

    const referenceNo = 'REF' + Math.floor(Math.random() * 1000000);
    alert(`Complaint submitted successfully!\nReference No: ${referenceNo}`);

    // Clear form after submission
    setEmployeeName('');
    setComplaint('');
  };

  return (
    <div style={{ maxWidth: '500px', margin: '2rem auto', padding: '20px',
border: '1px solid gray', borderRadius: '8px' }}>
      <h2>Complaint Registration Form</h2>
      <form onSubmit={handleSubmit}>
        <div style={{ marginBottom: '1rem' }}>
          <label>Employee Name: </label><br />
          <input
            type="text"
            value={employeeName}
            onChange={(e) => setEmployeeName(e.target.value)}
            placeholder="Enter your name"
            style={{ width: '100%', padding: '8px' }}
          />
        </div>

        <div style={{ marginBottom: '1rem' }}>
          <label>Complaint:</label><br />
          <textarea
            value={complaint}
            onChange={(e) => setComplaint(e.target.value)}
            placeholder="Describe your complaint"
            rows="4"
            style={{ width: '100%', padding: '8px' }}
          />
        </div>

        <button type="submit" style={{ padding: '10px 20px' }}>
          Submit Complaint
        </button>
      </form>
    </div>
  );
}

export default ComplaintRegister;

```

src/ App.js

```

import React from 'react';
import './App.css';
import ComplaintRegister from './ComplaintRegister';

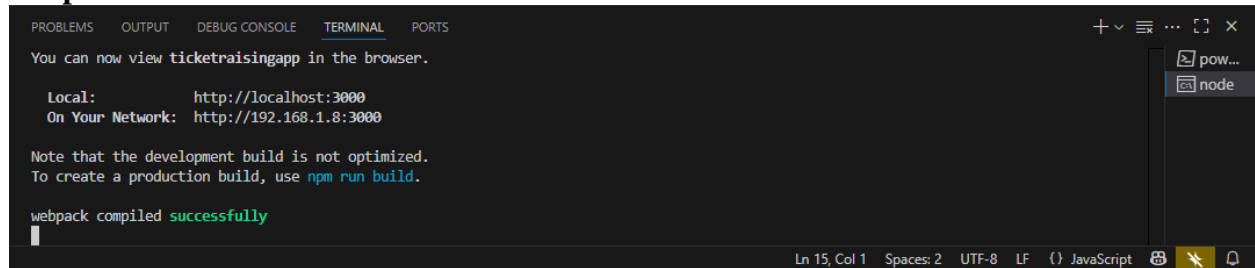
function App() {
  return (
    <div className="App">
      <h1>Ticket Raising Application</h1>
      <ComplaintRegister />
    </div>
  );
}

```

```
    </div>
  );
}

export default App;
```

Output:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
You can now view ticketraisingapp in the browser.

Local:      http://localhost:3000
On Your Network: http://192.168.1.8:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

<http://localhost:3000/>



Ticket Raising Application

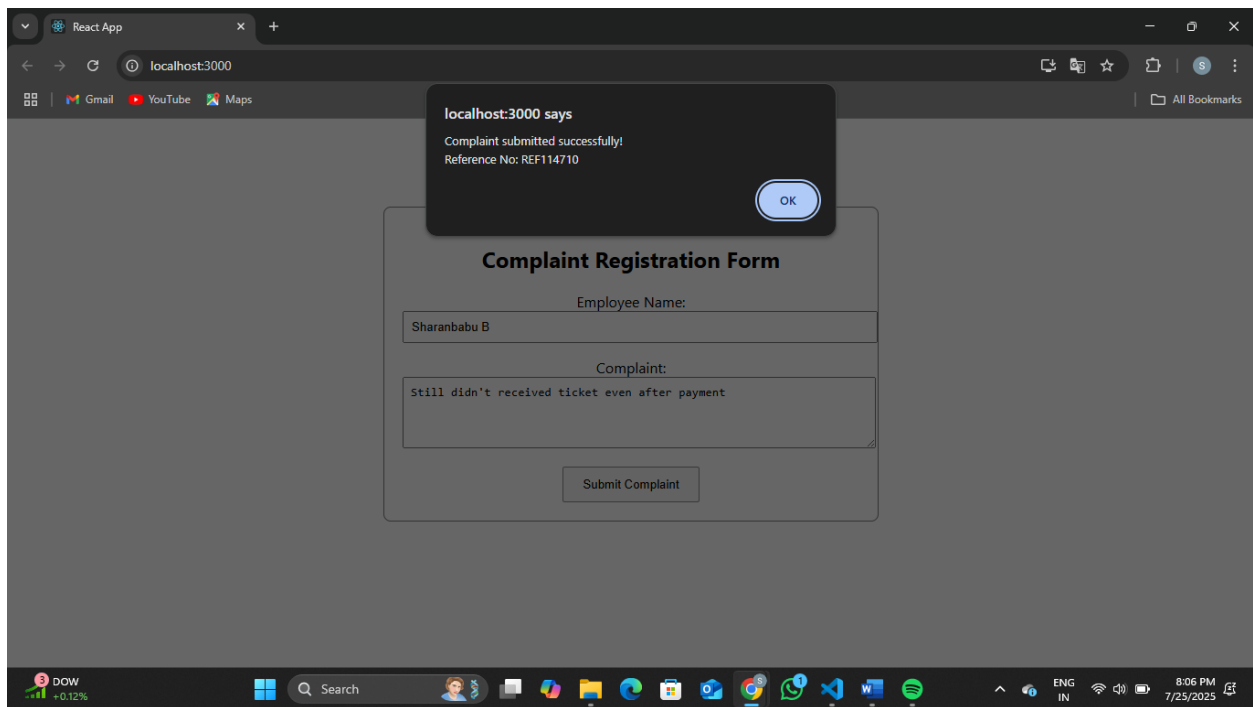
Complaint Registration Form

Employee Name:

Complaint:

Submit Complaint





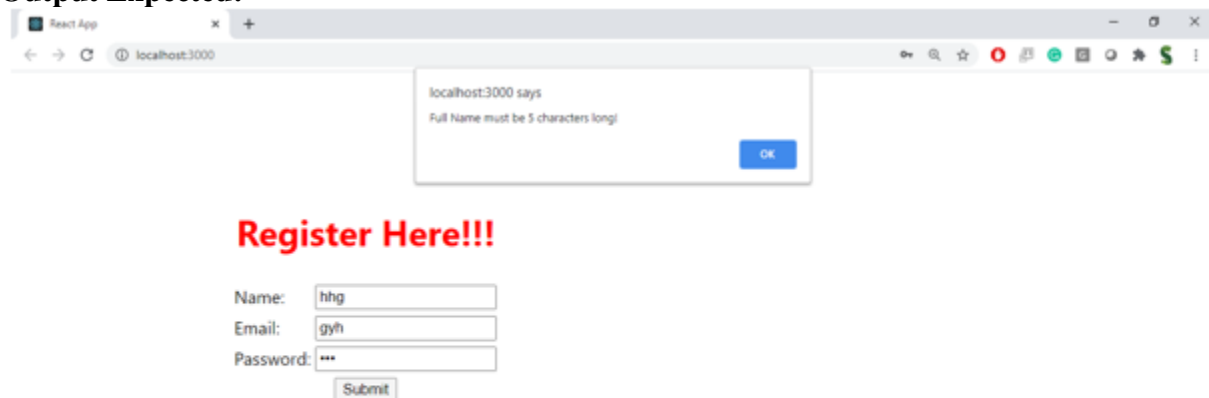
Exercise 8:

Create a React App named “mailregisterapp” which will have a component named “register.js”. Create a form which accepts the name, email and password and validate the fields as per the following:

1. Name should have atleast 5 characters
2. Email should have @ and .
3. Password should have atleast 8 characters.

Ensure that validations are implemented through eventhandle and eventsubmit of a form.

Output Expected:





Register Here!!!

Name:

Email:

Password:



Register Here!!!

Name:

Email:

Password:

Hint:

```
handleSubmit = (event) => {  
  event.preventDefault();  
  if(validateForm(this.state.errors)) {  
    alert('Valid Form')  
  }else{  
    if(this.state.errors.fullName!="")  
    {alert(this.state.errors.fullName)}  
    if(this.state.errors.email != "")  
    {alert(this.state.errors.email)}  
    if(this.state.errors.password != "")  
    {alert(this.state.errors.password)}  
  }  
}
```

```

switch (name) {
  case 'fullName':
    errors.fullName =
      value.length < 5
        ? 'Full Name must be 5 characters long!'
        : '';
    break;

  case 'email':
    const validEmailRegex =
      RegExp(/^([^\<>()\\[\]{};:\s@'"]+(\.[^\<>()\\[\]{};:\s@'"]+)*)(\.[^\<>()\\[\]{};:\s@'"]+)+([^\<>()\\[\]{};:\s@'"]{2,})$/i);

    errors.email =
      validEmailRegex.test(value)
        ? ''
        : 'Email is not valid!';
    break;

```

src/ Register.js

```

import React, { useState } from 'react';

function Register() {
  const [form, setForm] = useState({
    name: '',
    email: '',
    password: ''
  });

  const [errors, setErrors] = useState({});

  const validate = () => {
    const newErrors = {};

    if (form.name.length < 5) {
      newErrors.name = 'Name must be at least 5 characters long.';
    }

    if (!form.email.includes('@') || !form.email.includes('.')) {
      newErrors.email = 'Email must contain "@" and "."';
    }

    if (form.password.length < 8) {
      newErrors.password = 'Password must be at least 8 characters long.';
    }

    setErrors(newErrors);
    return Object.keys(newErrors).length === 0; // No errors
  };

  const handleChange = (e) => {
    setForm({ ...form, [e.target.name]: e.target.value });
  };

```

```

const handleSubmit = (e) => {
  e.preventDefault();

  if (validate()) {
    alert(`Registration Successful!\nName: ${form.name}`);
    setForm({ name: '', email: '', password: '' });
    setErrors({});
  }
};

return (
  <div style={{ maxWidth: '400px', margin: '2rem auto', padding: '20px',
border: '1px solid #ccc', borderRadius: '8px' }}>
    <h2>Mail Registration Form</h2>
    <form onSubmit={handleSubmit}>
      <div>
        <label>Name:</label><br />
        <input
          type="text"
          name="name"
          value={form.name}
          onChange={handleChange}
          style={{ width: '100%', padding: '8px' }}
        />
        {errors.name && <p style={{ color: 'red' }}>{errors.name}</p>}
      </div>

      <div>
        <label>Email:</label><br />
        <input
          type="text"
          name="email"
          value={form.email}
          onChange={handleChange}
          style={{ width: '100%', padding: '8px' }}
        />
        {errors.email && <p style={{ color: 'red' }}>{errors.email}</p>}
      </div>

      <div>
        <label>Password:</label><br />
        <input
          type="password"
          name="password"
          value={form.password}
          onChange={handleChange}
          style={{ width: '100%', padding: '8px' }}
        />
        {errors.password && <p style={{ color: 'red' }}>{errors.password}</p>}
      </div>

      <button type="submit" style={{ marginTop: '1rem', padding: '10px
20px' }}>
        Register
      </button>
    </form>
  </div>
);

```

```

        </form>
      </div>
    );
  }

export default Register;

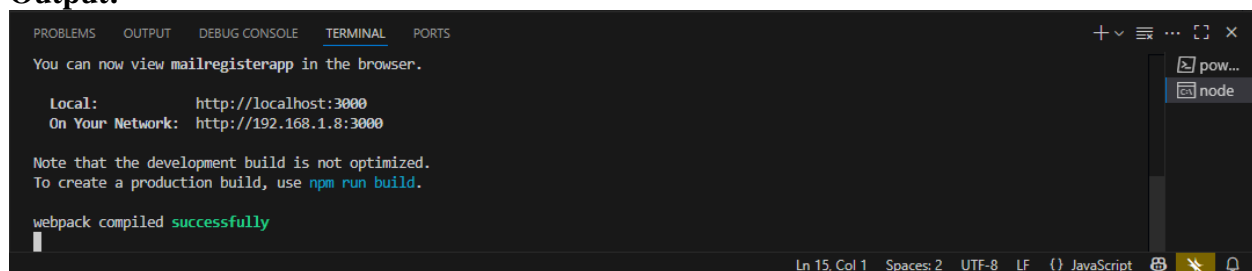
src/ App.js
import React from 'react';
import './App.css';
import Register from './Register';

function App() {
  return (
    <div className="App">
      <h1>Mail Registration App</h1>
      <Register />
    </div>
  );
}

export default App;

```

Output:



<http://localhost:3000/>



Mail Registration App

Mail Registration Form

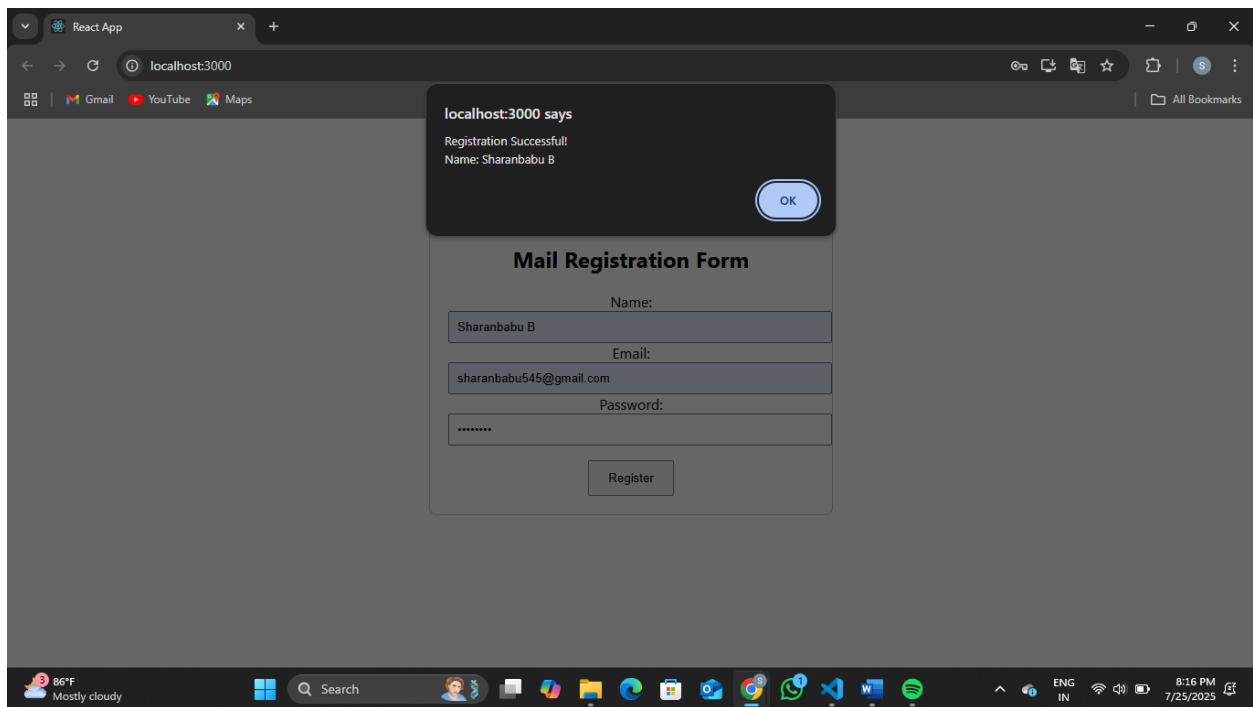
Name:

Email:

Password:

Register





Mail Registration App

Mail Registration Form

Name:

Sharanbabu B

Email:

sharanbabu545

Email must contain "@" and ".".

Password:

....

Password must be at least 8 characters long.

Register

Exercise 9:

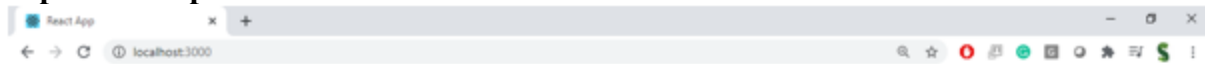
Create a React Application “fetchuserapp” which will retrieve the user details from <https://api.randomuser.me/> and display the title, firstname and image of a user.

Create a component named “Getuser” and in the asynchronous method “ComponentDidMount ()” invoke the URL using fetch method and the response can be displayed in the render method of the component.

Code Snippet in Getuser Component:

```
async componentDidMount()
{
  const url="https://api.randomuser.me/";
  const response=await fetch(url);
  const data=await response.json();
  this.setState({person:data.results[0],loading:false});
  console.log(data.results[0]);
}
```

Expected Output:



Mr Donato Nunes



src/ Getuser.js

```
import React, { Component } from 'react';

class Getuser extends Component {
  constructor(props) {
    super(props);
    this.state = {
      user: null,
      loading: true
    };
  }

  componentDidMount() {
    fetch('https://api.randomuser.me/')
      .then((response) => response.json())
      .then((data) => {
        this.setState({
          user: data.results[0],
          loading: false
        });
      });
  }
}
```

```

        .catch((error) => {
            console.error('Error fetching data:', error);
            this.setState({ loading: false });
        });
    }

    render() {
        const { user, loading } = this.state;

        if (loading) {
            return <h2>Loading user...</h2>;
        }

        if (!user) {
            return <h2>No user data found</h2>;
        }

        return (
            <div style={{ textAlign: 'center', marginTop: '30px' }}>
                <h2>User Information</h2>
                <p><strong>Title:</strong> {user.name.title}</p>
                <p><strong>First Name:</strong> {user.name.first}</p>
                <img src={user.picture.large} alt="User" style={{ borderRadius: '50%'
            </div>
        );
    }
}

export default Getuser;

```

src/ App.js

```

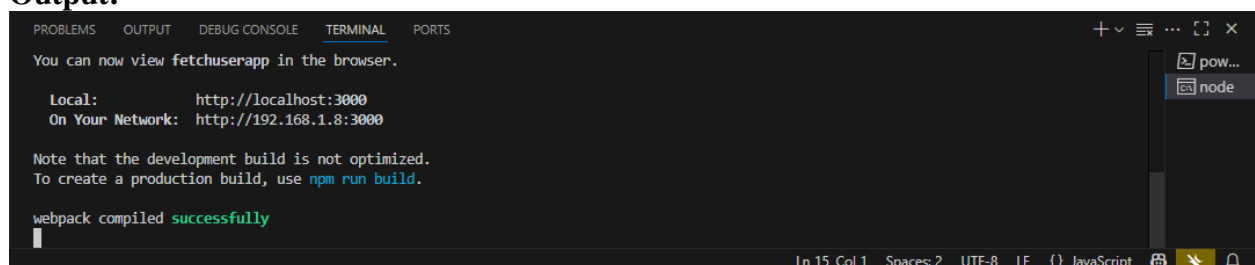
import React from 'react';
import './App.css';
import Getuser from './Getuser';

function App() {
    return (
        <div className="App">
            <h1>Fetch Random User</h1>
            <Getuser />
        </div>
    );
}

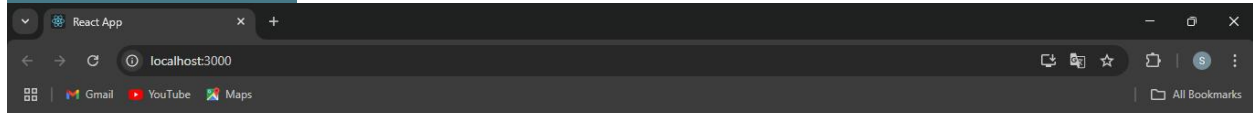
export default App;

```

Output:



<http://localhost:3000/>

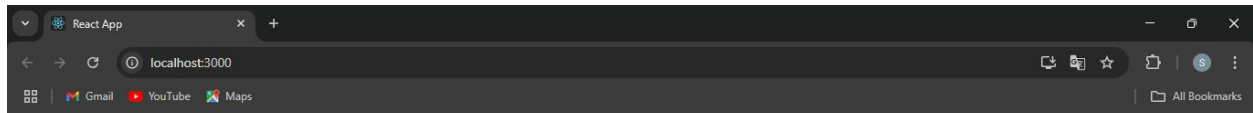


Fetch Random User

User Information

Title: Mr

First Name: Rodoljub



Fetch Random User

User Information

Title: Mrs

First Name: Elya

