

TOPIC:- ReactJS-HOL

Q.9.) 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.**

2. IndianPlayers

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

Solution-

Code:-

App.js

```
import React from 'react';
import ListofPlayers from './ListofPlayers';
import IndianPlayers from './IndianPlayers';
function App() {
  const flag = true;
```

```
return (
  <div className="App">
    {flag ? <ListofPlayers /> : <IndianPlayers />}
  </div>
);
}

export default App;
```

ListofPlayers.js

```
import React from 'react';
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 }
];
function ListofPlayers() {
  const players70 = players.filter(item => item.score <= 70);
  return (
    <div>
      <h2>List of Players</h2>
      <ul>
```

```

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

</ul>

<h2>List of Players having Scores Less than 70</h2>
<ul>
  {players70.map(item => (
    <li key={item.name}>
      Mr. {item.name} <span>{item.score}</span>
    </li>
  ))}

</ul>

</div>

);

}

export default ListofPlayers;

```

IndianPlayers.js

```

import React from 'react';

import { IndianPlayers } from './playersData';

function OddPlayers([first, , third, , fifth]) {
  return (
    <div>
      <h2>Odd Players</h2>
      <ul>
        <li>First : {first}</li>
        <li>Third : {third}</li>
      </ul>
    </div>
  );
}

export default OddPlayers;

```

```
<li>Fifth : {fifth}</li>
</ul>
</div>
);
}

function EvenPlayers([, second, , fourth, , sixth]) {
  return (
    <div>
      <h2>Even Players</h2>
      <ul>
        <li>Second : {second}</li>
        <li>Fourth : {fourth}</li>
        <li>Sixth : {sixth}</li>
      </ul>
    </div>
  );
}

function IndianPlayersComponent() {
  return (
    <div>
      <OddPlayers players={IndianPlayers} />
      <EvenPlayers players={IndianPlayers} />
      <h2>List of Indian Players Merged:</h2>
      <ul>
        {IndianPlayers.map((player, index) => (
          <li key={index}>Mr. {player}</li>
        )))
      </ul>
    </div>
  );
}
```

```
}

export default IndianPlayersComponent;
```

playersData.js

```
const T20Players = ['First Player', 'Second Player', 'Third Player'];

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

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

OUTPUT:-

The screenshot shows a browser window with two tabs: "React cricketapp components" and "React App". The "React App" tab is active, displaying the output of the IndianPlayers component. The page title is "List of Players" and it contains a list of 12 players. Below this, there is a section titled "List of Players having Scores Less than 70" which lists 6 players. The page then branches into two sections: "Odd Players" (listing 3 players) and "Even Players" (listing 3 players). Finally, there is a section titled "List of Indian Players Merged:" which lists all 12 players from the initial list.

List of Players

- Mr. Jack 50
- Mr. Michael 70
- Mr. John 40
- Mr. Ann 61
- Mr. Elisabeth 61
- Mr. Sachin 95
- Mr. Dhoni 100
- Mr. Virat 84
- Mr. Jadeja 64
- Mr. Raina 75
- Mr. Rohit 80

List of Players having Scores Less than 70

- Mr. Jack 50
- Mr. Michael 70
- Mr. John 40
- Mr. Ann 61
- Mr. Elisabeth 61
- Mr. Jadeja 64

Odd Players

- First : Sachin1
- Third : Virat3
- Fifth : Yuvraj5

Even Players

- Second : Dhoni2
- Fourth : Rohit4
- Sixth : Raina6

List of Indian Players Merged:

- Mr. First Player
- Mr. Second Player
- Mr. Third Player
- Mr. Fourth Player
- Mr. Fifth Player
- Mr. Sixth Player

Q.10.) Create a React Application named “officespacentalapp” 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.

Solution:-

Code:-

```
npx create-react-app officespacentalapp
```

```
cd officespacentalapp
```

src/App.js

```
import React from 'react';
import './App.css';
function App() {
  const office = {
    name: "Skyview Tower",
    rent: 55000,
    address: "123 Business Street, Bengaluru",
    image: "https://via.placeholder.com/300x200?text=Office+Space"
  };
  const officeList = [
    {
      name: "Cloud Nine Office",
      rent: 70000,
      address: "456 Business Street, Bengaluru",
      image: "https://via.placeholder.com/300x200?text=Cloud+Nine+Office"
    },
    {
      name: "Tech Hub Office",
      rent: 65000,
      address: "567 Business Street, Bengaluru",
      image: "https://via.placeholder.com/300x200?text=Tech+Hub+Office"
    }
  ];
  return (
    <div>
      <h1>Office Space Rental App</h1>
      <p>This app displays a list of office spaces available for rental. Each item shows the office name, rent amount, address, and a placeholder image.</p>
      <table border="1">
        <thead>
          <tr>
            <th>Office Name</th>
            <th>Rent</th>
            <th>Address</th>
            <th>Image</th>
          </tr>
        </thead>
        <tbody>
          {officeList.map(item => (
            <tr>
              <td>{item.name}</td>
              <td>${item.rent}</td>
              <td>{item.address}</td>
              <td><img alt={item.image} /></td>
            </tr>
          ))}
        </tbody>
      </table>
    </div>
  );
}

export default App;
```

```
        name: "Downtown Heights",
        rent: 75000,
        address: "45 MG Road, Bengaluru",
        image: "https://via.placeholder.com/300x200?text=Office+1"
    },
    {
        name: "TechPark Square",
        rent: 58000,
        address: "88 IT Hub, Hyderabad",
        image: "https://via.placeholder.com/300x200?text=Office+2"
    },
    {
        name: "Business Bay",
        rent: 92000,
        address: "12 Marina Drive, Mumbai",
        image: "https://via.placeholder.com/300x200?text=Office+3"
    }
];
return (
<div className="App">
    <h1>Office Space Rental App</h1>
    <div className="office-card">
        <img src={office.image} alt="Office" />
        <h2>{office.name}</h2>
        <p>Address: {office.address}</p>
        <p style={{ color: office.rent < 60000 ? 'red' : 'green' }}>
            Rent: ₹{office.rent}
        </p>
    </div>
    <hr />
```

```

<h2>Available Office Listings:</h2>
<div className="office-list">
  {officeList.map((item, index) => (
    <div key={index} className="office-card">
      <img src={item.image} alt={item.name} />
      <h2>{item.name}</h2>
      <p>Address: {item.address}</p>
      <p style={{ color: item.rent < 60000 ? 'red' : 'green' }}>
        Rent: ₹{item.rent}
      </p>
    </div>
  )));
</div>
</div>
);
}

export default App;

```

src/App.css

```

.App {
  font-family: Arial, sans-serif;
  padding: 20px;
}

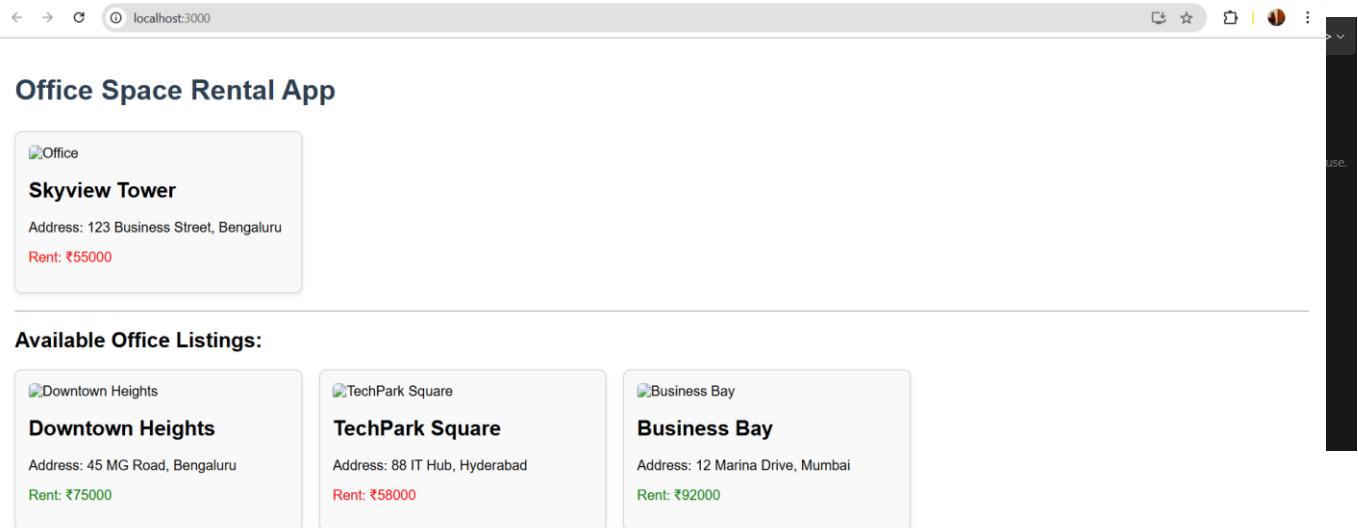
h1 {
  color: #2c3e50;
  margin-bottom: 30px;
}

.office-card {
  border: 1px solid #ccc;
}

```

```
padding: 15px;  
margin-bottom: 20px;  
border-radius: 8px;  
box-shadow: 0 2px 5px rgba(0,0,0,0.1);  
width: 300px;  
background-color: #f9f9f9;  
}  
  
.office-card img {  
width: 100%;  
height: auto;  
border-radius: 5px;  
}  
  
.office-list {  
display: flex;  
flex-wrap: wrap;  
gap: 20px;  
}
```

OUTPUT:-



The screenshot shows a web browser window with the URL `localhost:3000` in the address bar. The page title is "Office Space Rental App". Below the title, there is a card for "Skyview Tower" with a placeholder image, address, and rent information. A horizontal line separates this from the "Available Office Listings" section. In this section, there are three cards: "Downtown Heights", "TechPark Square", and "Business Bay", each with a placeholder image, address, and rent information.

Office Listing	Address	Rent
Skyview Tower	123 Business Street, Bengaluru	₹55000
Downtown Heights	45 MG Road, Bengaluru	₹75000
TechPark Square	88 IT Hub, Hyderabad	₹58000
Business Bay	12 Marina Drive, Mumbai	₹92000

Q.11.) 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 “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.

Solution:-

Code:-

src/App.js

```
import React, { useState } from "react";
import "./App.css";
import CurrencyConvertor from "./CurrencyConvertor";
function App() {
  const [count, setCount] = useState(0);
  const handleIncrement = () => {
```

```
incrementValue();
sayHello();
};

const incrementValue = () => {
  setCount(count + 1);
};

const decrementValue = () => {
  setCount(count - 1);
};

const sayHello = () => {
  console.log("Hello! This is a static message.");
};

const sayWelcome = (message) => {
  alert(message);
};

const handlePress = () => {
  alert("I was clicked");
};

return (
  <div className="App">
    <h1>Event Examples App</h1>
    <h2>Counter: {count}</h2>
    <button onClick={handleIncrement}>Increment</button>
    <button onClick={decrementValue}>Decrement</button>
    <hr />
    <button onClick={() => sayWelcome("Welcome!")}>Say Welcome</button>
    <hr />
    <button onClick={handlePress}>OnPress</button>
    <hr />
    <CurrencyConvertor />
  </div>
);
```

```
</div>
);
}

export default App;
```

src/CurrencyConvertor.js

```
import React, { useState } from "react";
function CurrencyConvertor() {
  const [inr, setInr] = useState("");
  const [euro, setEuro] = useState(null);
  const conversionRate = 0.011; // Example conversion rate
  const handleSubmit = (e) => {
    e.preventDefault();
    const converted = parseFloat(inr) * conversionRate;
    setEuro(converted.toFixed(2));
  };
  return (
    <div>
      <h2>Currency Convertor (INR → EURO)</h2>
      <form onSubmit={handleSubmit}>
        <label>Enter amount in INR:</label>
        <input
          type="number"
          value={inr}
          onChange={(e) => setInr(e.target.value)}
          required
        />
        <button type="submit">Convert</button>
      </form>
    </div>
  );
}

export default CurrencyConvertor;
```

```
{euro !== null && (  
    <p>  
        ₹{inr} INR = €{euro} Euro  
    </p>  
)  
</div>  
);  
}  
export default CurrencyConvertor;
```

src/App.css

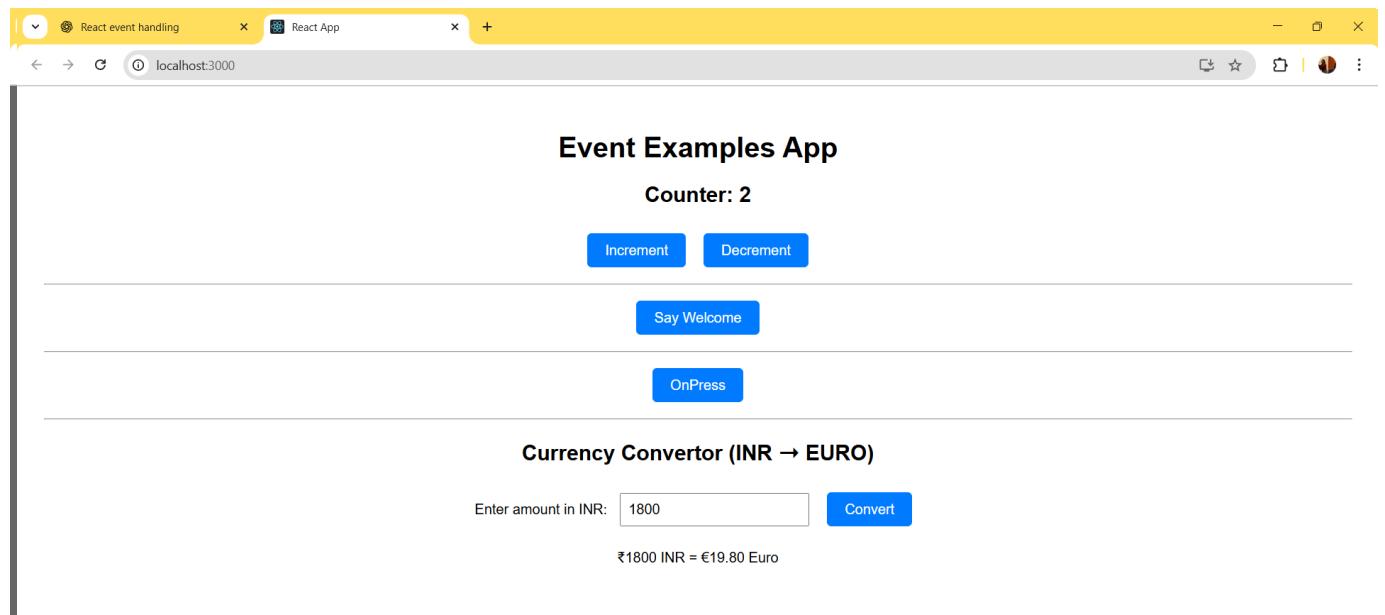
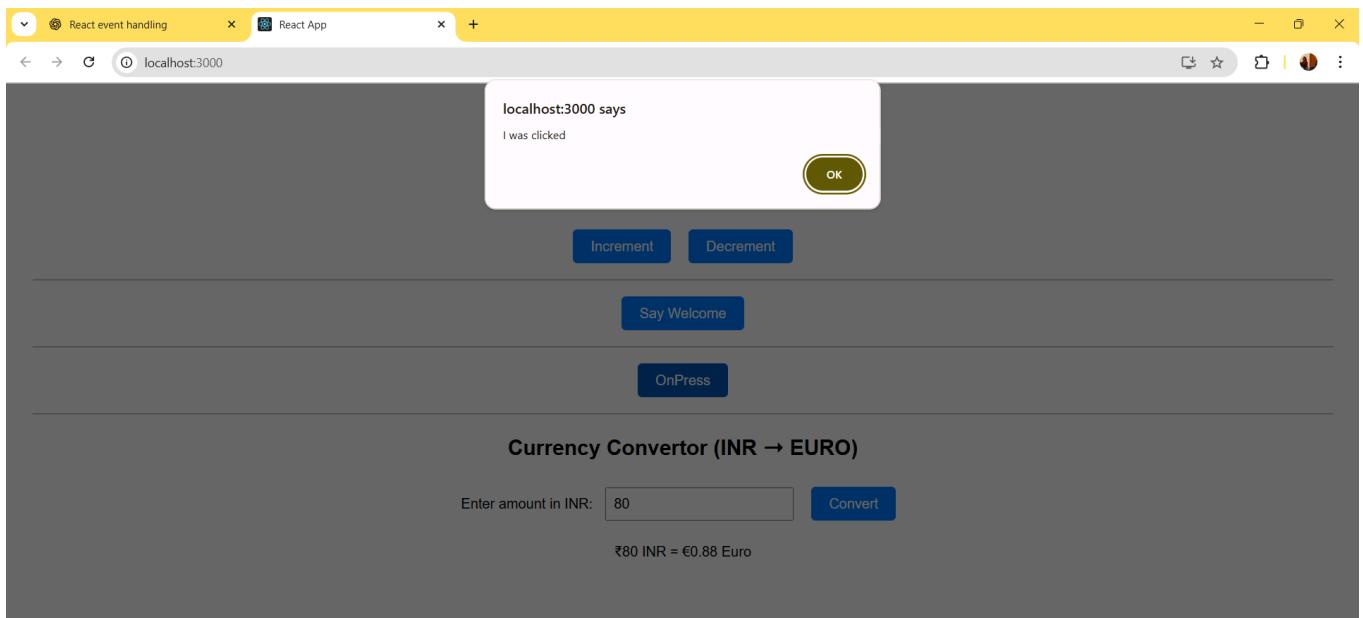
```
.App {  
    text-align: center;  
    padding: 30px;  
    font-family: Arial, sans-serif;  
}  
button {  
    margin: 10px;  
    padding: 10px 20px;  
    font-size: 16px;  
    cursor: pointer;  
    border-radius: 5px;  
    background-color: #007bff;  
    color: white;  
    border: none;  
}  
button:hover {  
    background-color: #0056b3;  
}
```

```
input {  
  padding: 8px;  
  margin: 10px;  
  font-size: 16px;  
}
```

src/index.js

```
import React from "react";  
import ReactDOM from "react-dom/client";  
import "./index.css";  
import App from "./App";  
const root = ReactDOM.createRoot(document.getElementById("root"));  
root.render(  
  <React.StrictMode>  
    <App />  
  </React.StrictMode>  
);
```

OUTPUT-



Q.)12.) 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.

Solution:

Code:

src/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', fontFamily: 'Arial' }}>
      <h1>Ticket Booking App</h1>
      <div>
        {!isLoggedIn ? (
          <button onClick={handleLogin}>Login</button>
        ) : (
          <button onClick={handleLogout}>Logout</button>
        )}
      </div>
      <hr />
      {isLoggedIn ? <UserPage /> : <GuestPage />}
    </div>
  );
}
```

```
}

export default App;
```

src/GuestPage.js

```
import React from 'react';

function GuestPage() {
  const flights = [
    { id: 1, from: 'Delhi', to: 'Mumbai', time: '10:00 AM', price: '₹4500' },
    { id: 2, from: 'Bangalore', to: 'Chennai', time: '1:30 PM', price: '₹3200' },
    { id: 3, from: 'Kolkata', to: 'Hyderabad', time: '5:15 PM', price: '₹3900' },
  ];

  return (
    <div>
      <h2>Welcome Guest</h2>
      <p>Please login to book tickets.</p>
      <h3>Available Flights</h3>
      <ul style={{ listStyleType: 'none', padding: 0 }}>
        {flights.map((flight) => (
          <li key={flight.id} style={{ marginBottom: '10px' }}>
             {flight.from} → {flight.to} at {flight.time} – {flight.price}
          </li>
        )));
      </ul>
    </div>
  );
}

export default GuestPage;
```

src/UserPage.js

```
import React from 'react';

function UserPage() {
  const handleBook = (flight) => {
    alert(` ✅ Ticket booked from ${flight.from} to ${flight.to}`);
  };

  const flights = [
    { id: 1, from: 'Delhi', to: 'Mumbai', time: '10:00 AM', price: '₹4500' },
    { id: 2, from: 'Bangalore', to: 'Chennai', time: '1:30 PM', price: '₹3200' },
    { id: 3, from: 'Kolkata', to: 'Hyderabad', time: '5:15 PM', price: '₹3900' },
  ];
  return (
    <div>
      <h2>Welcome User</h2>
      <h3>Book Your Flight</h3>
      <ul style={{ listStyleType: 'none', padding: 0 }}>
        {flights.map((flight) => (
          <li key={flight.id} style={{ marginBottom: '10px' }}>
            <img alt="Flight icon" style={{ verticalAlign: 'middle' }} /> {flight.from} → {flight.to} at {flight.time} – {flight.price}
            <button
              style={{ marginLeft: '10px' }}
              onClick={() => handleBook(flight)}
            >
              Book
            </button>
          </li>
        ))}
      </ul>
    </div>
```

```
 );
}

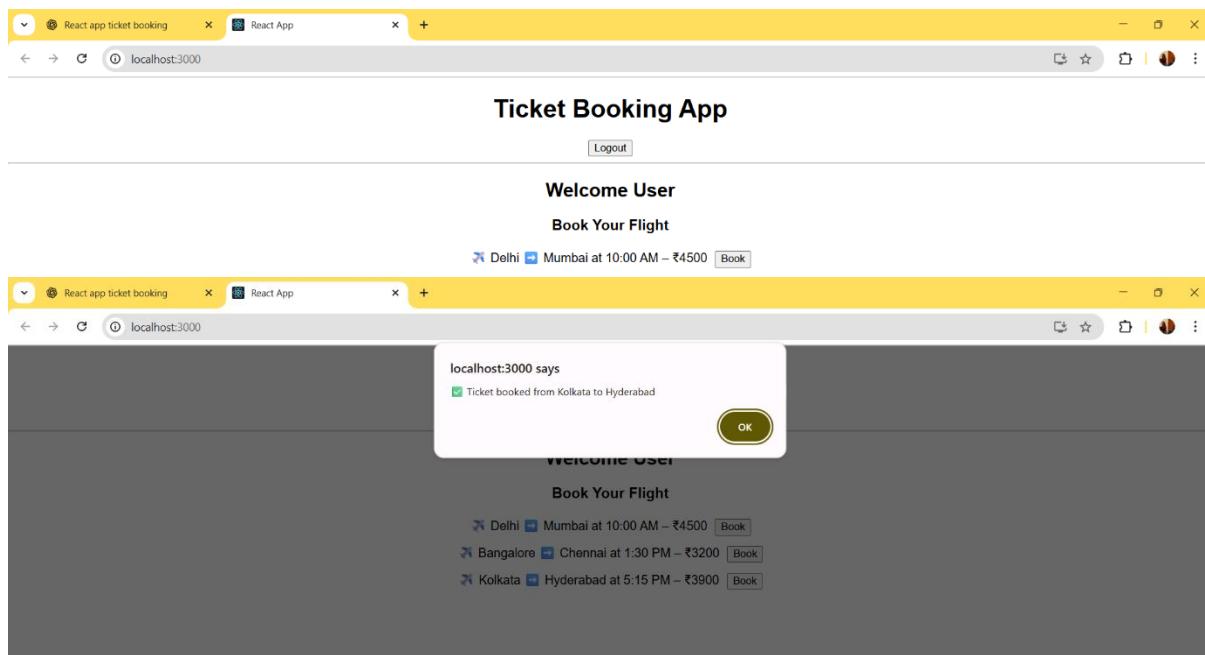
export default UserPage;
```

src/index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

OUTPUT-



Q.13.) 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.

Solution:

Code:

src/data.js

```
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 }  
];  
  
export const blogs = [  
  {  
    id: 1,  
    title: 'React Learning',  
    author: 'Stephen Biz',  
    content: 'Welcome to learning React!'  
  },  
  {  
    id: 2,  
    title: 'Installation',  
    author: 'Schwezdenier',  
    content: 'You can install React from npm.'  
  }]
```

```
}

];

export const courses = [
  { id: 1, cname: 'Angular', date: '4/5/2021' },
  { id: 2, cname: 'React', date: '6/3/20201' }
];
```

src/App.js

```
import React from 'react';
import './App.css';
import { books, blogs, courses } from './data';
function App() {
  const bookdet = (
    <ul>
      {books.map((book) => (
        <div key={book.id}>
          <h3>{book.bname}</h3>
          <h4>{book.price}</h4>
        </div>
      )))
    </ul>
  );
  const content = (
    <ul>
      {blogs.map((blog) => (
        <div key={blog.id}>
          <h3>{blog.title}</h3>
          <h5><strong>{blog.author}</strong></h5>
          <p>{blog.content}</p>
      )));
    </ul>
  );
  return (
    <div>
      {content}
    </div>
  );
}

export default App;
```

```
</div>
))}

</ul>

);

const coursedet = (
<ul>
{courses.map((course) => (
<div key={course.id}>
<h3>{course cname}</h3>
<h4>{course.date}</h4>
</div>
))}

</ul>

);

return (
<div className="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 App;
```

src/App.css

```
.container {
  display: flex;
  justify-content: space-around;
  padding: 40px;
}

.mystyle1, .st2, .v1 {
  padding: 10px;
  width: 30%;
  font-family: Arial, sans-serif;
}

.st2, .v1 {
  border-left: 5px solid green;
}
```

OUTPUT-

The screenshot shows a web browser window with a yellow header bar. The address bar displays "localhost:3002". The main content area contains three sections: "Course Details", "Book Details", and "Blog Details".

Course Details

- Angular**
- 4/5/2021**

React

- 6/3/20201**

Book Details

- Master React**
- 670**

- Deep Dive into Angular 11**
- 800**

- Mongo Essentials**
- 450**

Blog Details

- React Learning**
- Stephen Biz**
- Welcome to learning React!

- Installation**
- Schwezdenier**
- You can install React from npm.