# ReactJS – Week 7 Hands-on Lab Exercises

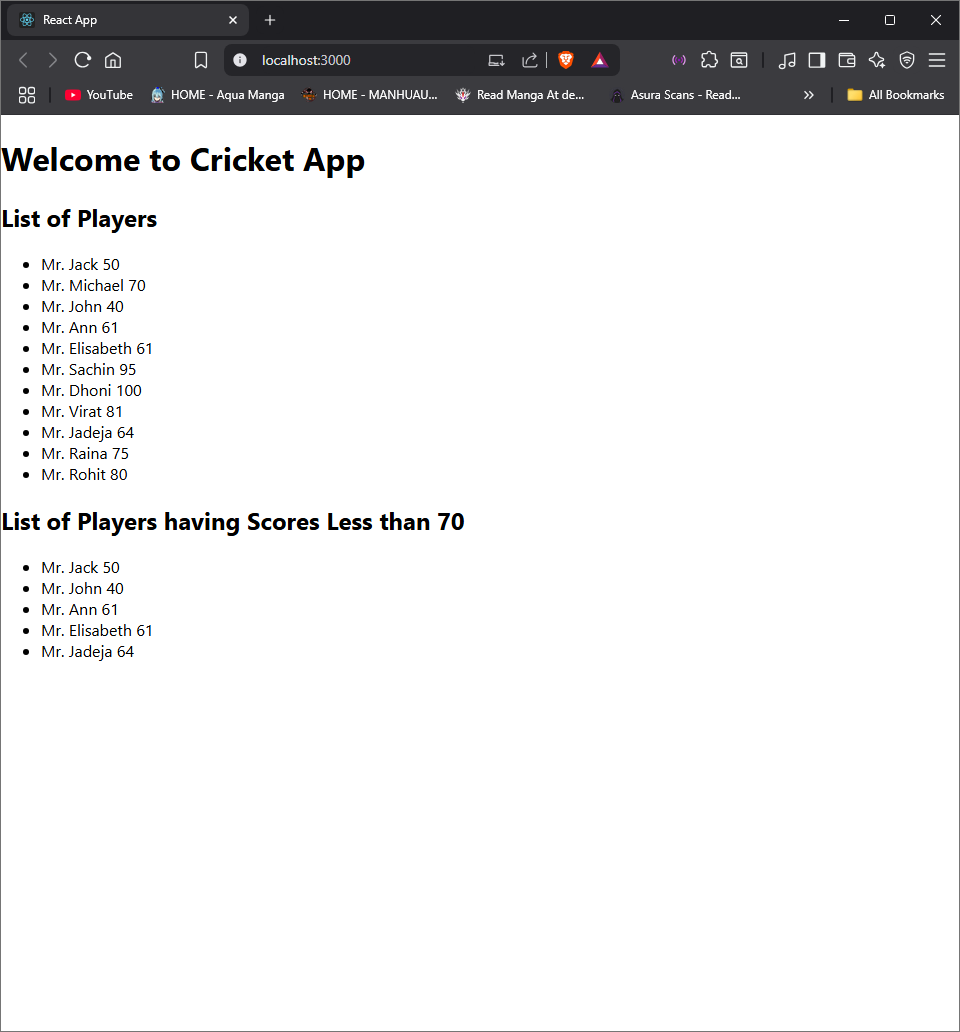
## Exercise 9

Create a React Application named “officespacerentalapp” which uses React JSX to render heading, images, and office details using objects and arrays. Display rent in red if below 60000 and green otherwise.

### Code:

App.js  
import React from 'react';  
import './App.css';  
  
const offices = [  
 { id: 1, name: 'Sky High Towers', rent: 45000, address: 'Bandra, Mumbai', image: 'https://via.placeholder.com/150' },  
 { id: 2, name: 'Tech Park Plaza', rent: 70000, address: 'Whitefield, Bangalore', image: 'https://via.placeholder.com/150' },  
 { id: 3, name: 'Urban Hub', rent: 52000, address: 'Hyderabad', image: 'https://via.placeholder.com/150' }  
];  
  
function App() {  
 return (  
 <div className="App">  
 <h1>Office Space Rental Listings</h1>  
 {offices.map(office => (  
 <div key={office.id} style={{ border: '1px solid gray', margin: '10px', padding: '10px' }}>  
 <img src={office.image} alt={office.name} width="150" />  
 <h2>{office.name}</h2>  
 <p>Address: {office.address}</p>  
 <p style={{ color: office.rent < 60000 ? 'red' : 'green', fontWeight: 'bold' }}>  
 Rent: ₹{office.rent}  
 </p>  
 </div>  
 ))}  
 </div>  
 );  
}  
  
export default App;

### Expected Output:



Exercise 10

Create a React Application eventexamplesapp to handle various events including increment, decrement, welcome message, synthetic events, and currency conversion.

Code:

App.js:

// App.js

import React, { Component } from 'react';

import CurrencyConverter from './components/CurrencyConverter';

class App extends Component {

constructor(props) {

super(props);

this.state = { count: 0 };

}

increment = () => {

this.setState({ count: this.state.count + 1 });

this.sayHello();

};

sayHello = () => {

alert("Hello! This is a static message.");

};

decrement = () => {

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

};

sayWelcome = (msg) => {

alert(`Welcome Message: ${msg}`);

};

handleClick = () => {

alert('I was clicked!');

};

render() {

return (

<div>

<h1>Counter: {this.state.count}</h1>

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

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

<br />

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

<br />

<button onClick={this.handleClick}>Synthetic Event</button>

<br /><br />

<CurrencyConverter />

</div>

);

}

}

export default App;

CurrencyConverter.js:

// components/CurrencyConverter.js

import React, { Component } from 'react';

class CurrencyConverter extends Component {

constructor(props) {

super(props);

this.state = { rupees: '', euro: '' };

}

handleChange = (e) => {

this.setState({ rupees: e.target.value });

};

handleSubmit = () => {

const euro = (this.state.rupees / 90).toFixed(2);

this.setState({ euro });

};

render() {

return (

<div>

<h2>Currency Converter</h2>

<input type="number" onChange={this.handleChange} placeholder="INR" />

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

<p>Euro: {this.state.euro}</p>

</div>

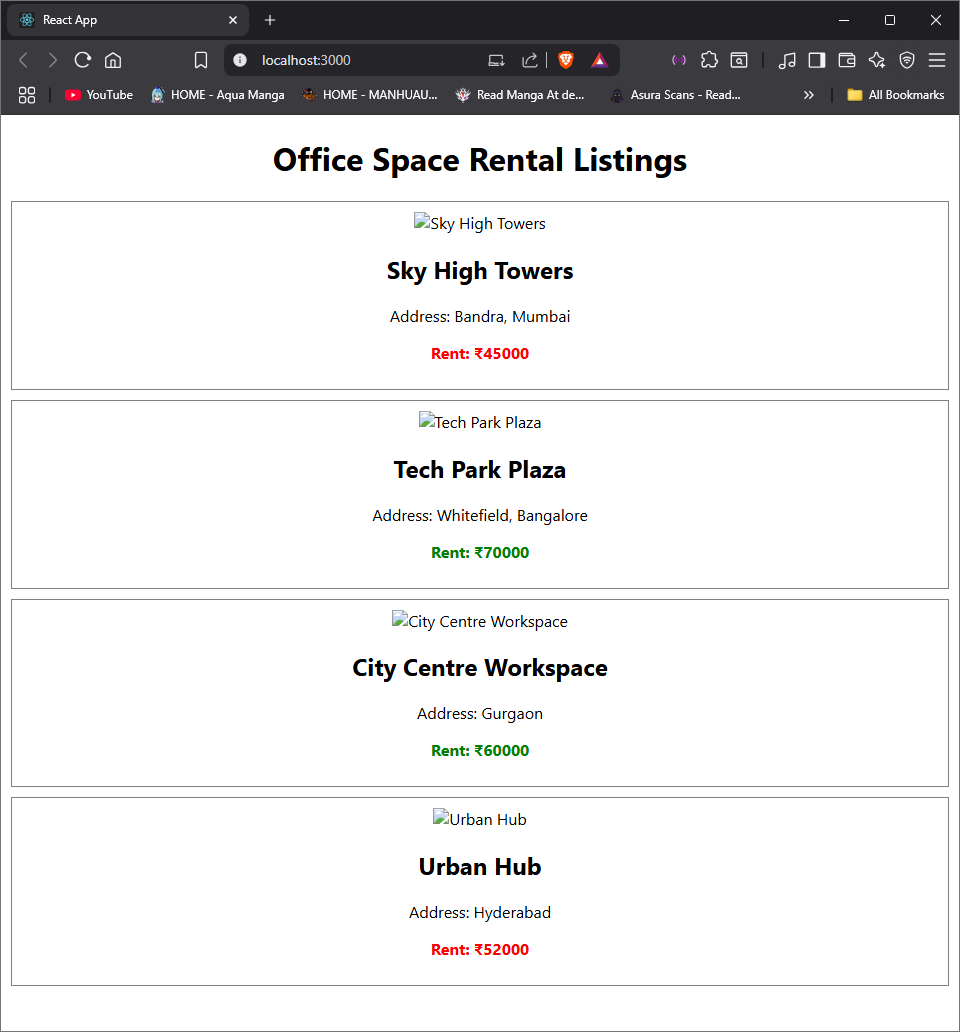
);

}

}

export default CurrencyConverter

### Expected Output:



Exercise 11:

Create a React Application ticketbookingapp to demonstrate conditional rendering. Show guest view by default, user view after login, and switch back on logout.

Code:

App.js:

// App.js

import React, { Component } from 'react';

import GuestPage from './components/GuestPage';

import UserPage from './components/UserPage';

class App extends Component {

constructor(props) {

super(props);

this.state = { isLoggedIn: false };

}

toggleLogin = () => {

this.setState({ isLoggedIn: !this.state.isLoggedIn });

};

render() {

return (

<div>

<h1>Flight Ticket Booking App</h1>

<button onClick={this.toggleLogin}>

{this.state.isLoggedIn ? "Logout" : "Login"}

</button>

{this.state.isLoggedIn ? <UserPage /> : <GuestPage />}

</div>

);

}

}

export default App;

UserPage.js:

// components/UserPage.js

import React from 'react';

const UserPage = () => (

<div>

<h2>Welcome, User!</h2>

<p>You can book your ticket now.</p>

<p>Flight: Air India | From: Mumbai | To: Delhi | ₹5500</p>

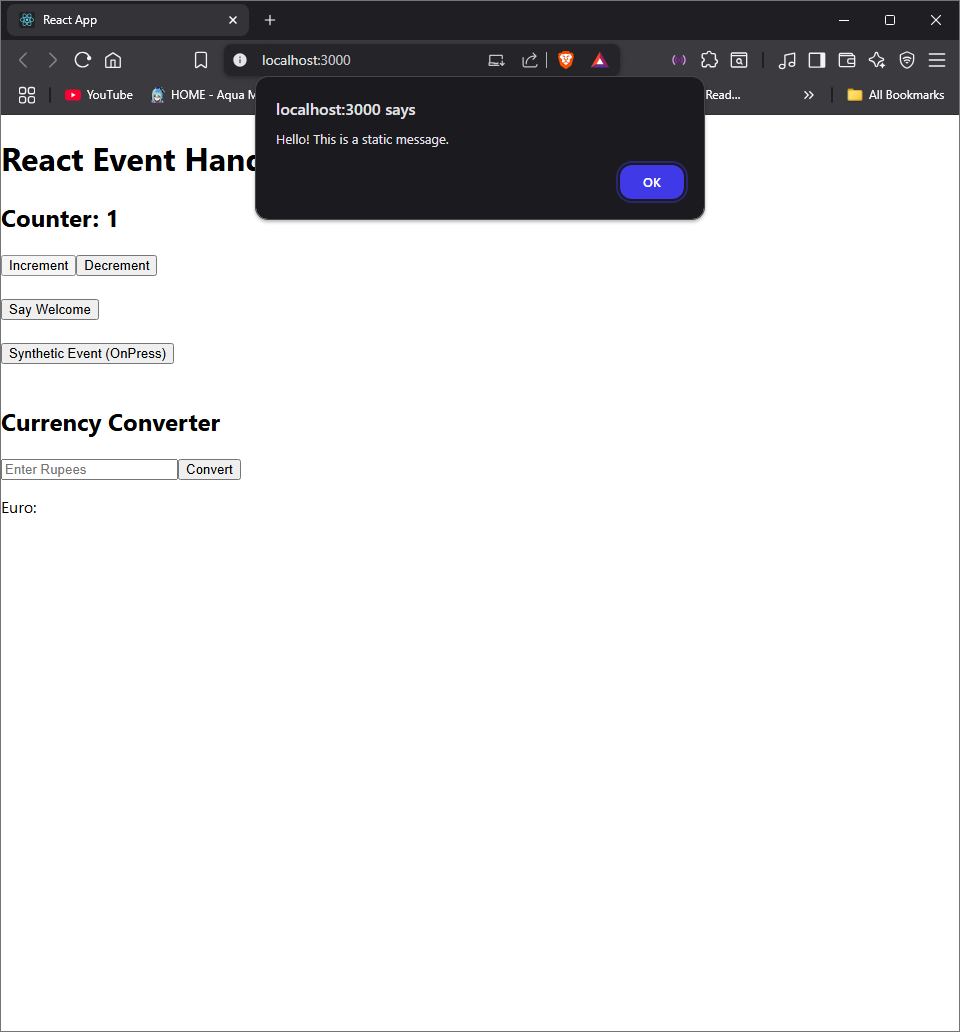
<button>Book Now</button>

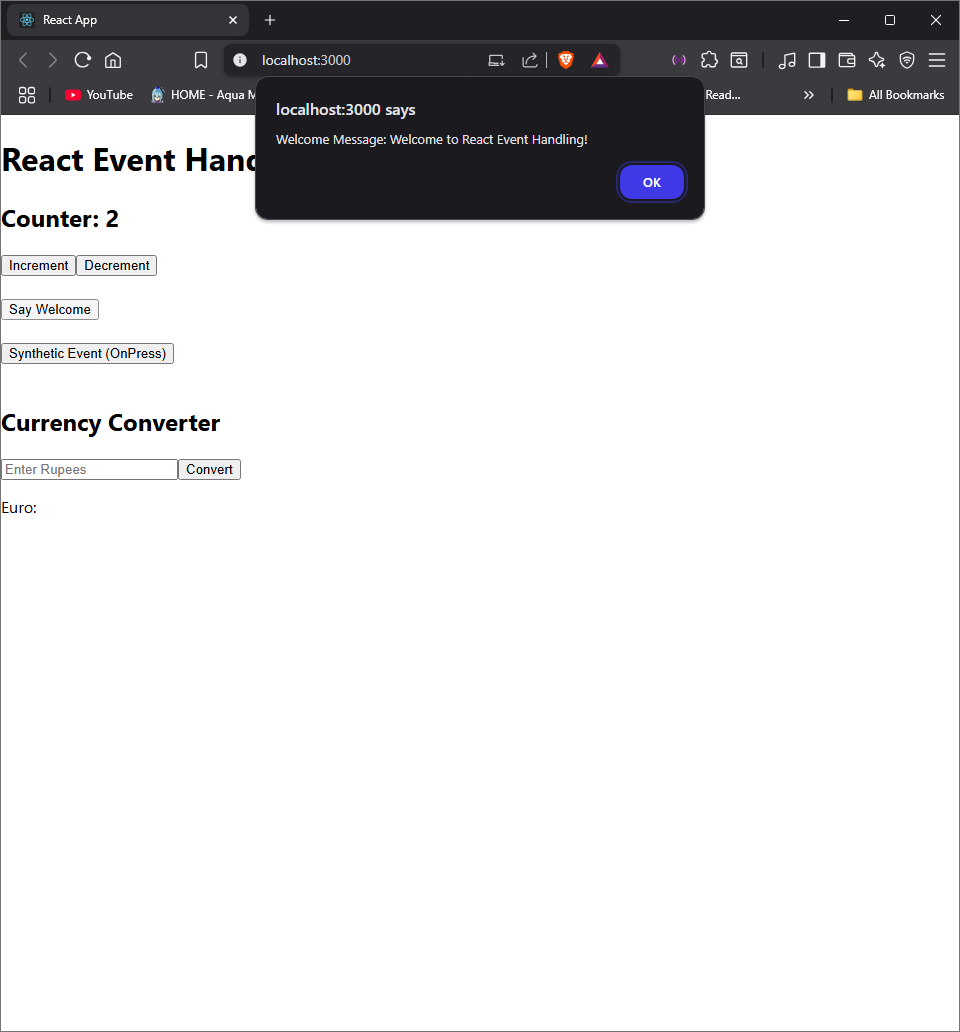
</div>

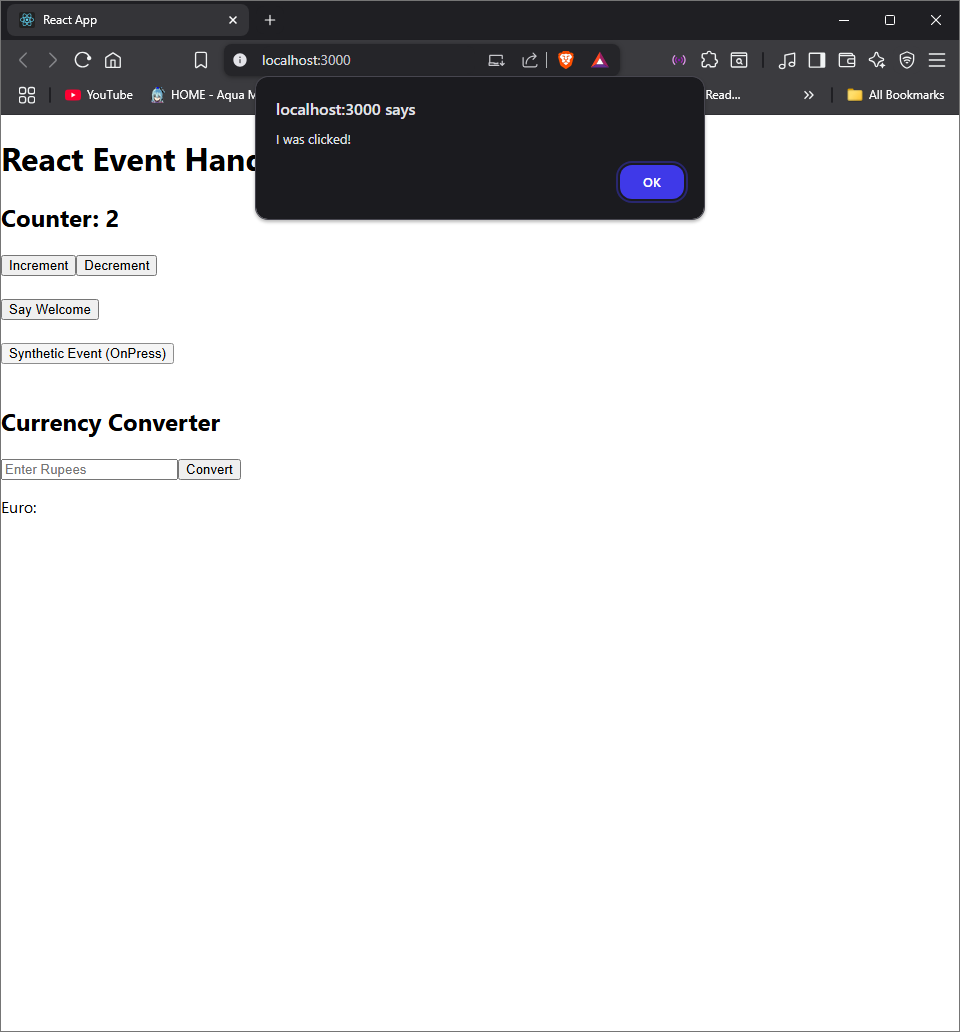
);

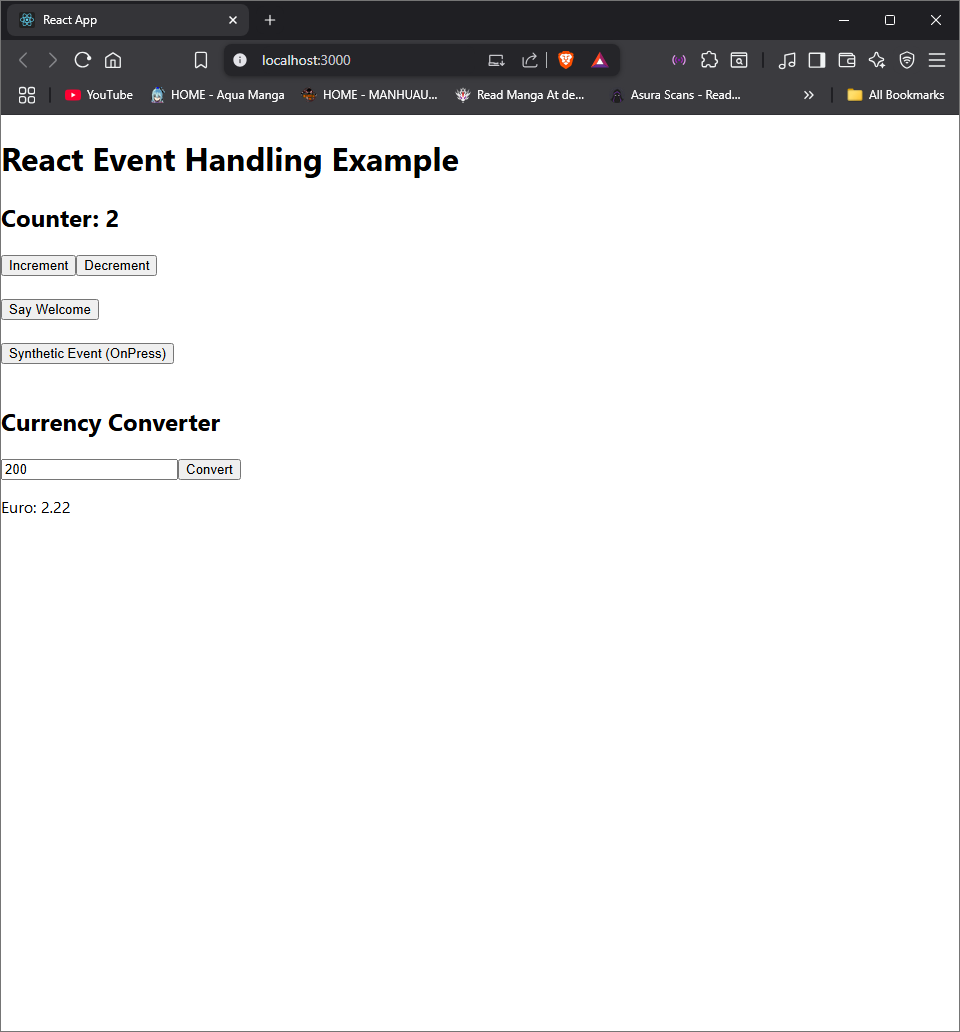
export default UserPage;

### Expected Output:









Exercise 12:

Create a React Application named **ticketbookingapp** to demonstrate **conditional rendering**. Show **guest view** by default, **user view** after login, and switch back on logout.

Code:

App.js:

// App.js

import React, { Component } from 'react';

import GuestPage from './components/GuestPage';

import UserPage from './components/UserPage';

class App extends Component {

constructor(props) {

super(props);

this.state = { isLoggedIn: false };

}

toggleLogin = () => {

this.setState({ isLoggedIn: !this.state.isLoggedIn });

};

render() {

return (

<div>

<h1>Flight Ticket Booking App</h1>

<button onClick={this.toggleLogin}>

{this.state.isLoggedIn ? "Logout" : "Login"}

</button>

{this.state.isLoggedIn ? <UserPage /> : <GuestPage />}

</div>

);

}

}

export default App;

GuestPage.js:

// components/GuestPage.js

import React from 'react';

const GuestPage = () => (

<div>

<h2>Welcome, Guest!</h2>

<p>Flight: Air India | From: Mumbai | To: Delhi | ₹5500</p>

</div>

);

export default GuestPage;

UserPage.js:

// components/UserPage.js

import React from 'react';

const UserPage = () => (

<div>

<h2>Welcome, User!</h2>

<p>You can book your ticket now.</p>

<p>Flight: Air India | From: Mumbai | To: Delhi | ₹5500</p>

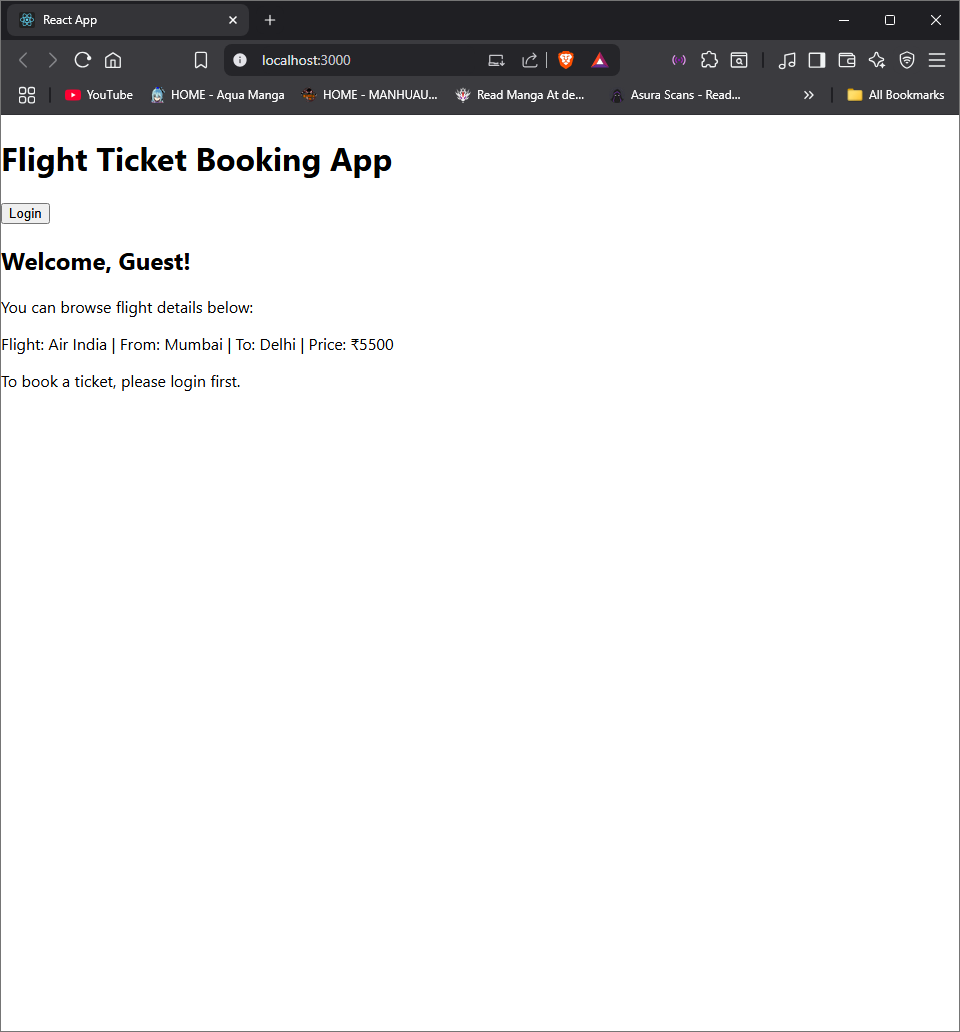
<button>Book Now</button>

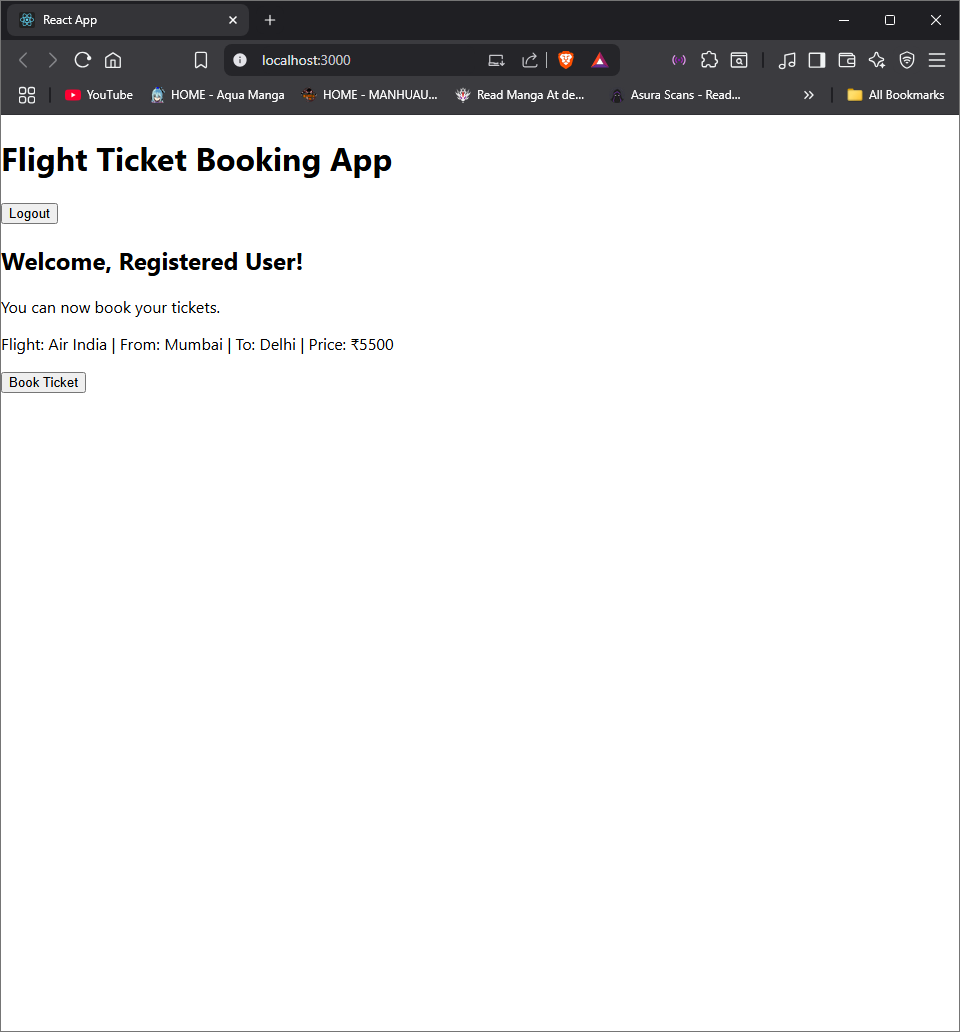
</div>

);

export default UserPage;

### Expected Output:





Exercise 13:

Create a React app bloggerapp to display a list of posts using .map(), key prop, conditional rendering, and reusable components. Include a toggle button to show/hide posts.

Code:

App.js:

import React, { Component } from 'react';

import BookDetails from './components/BookDetails';

import BlogDetails from './components/BlogDetails';

import CourseDetails from './components/CourseDetails';

class App extends Component {

constructor(props) {

super(props);

this.state = { view: 'book' };

}

renderComponent = () => {

switch (this.state.view) {

case 'book': return <BookDetails />;

case 'blog': return <BlogDetails />;

case 'course': return <CourseDetails />;

default: return <h3>Select View</h3>;

}

};

render() {

return (

<div>

<h1>Blogger Dashboard</h1>

<button onClick={() => this.setState({ view: 'book' })}>Book</button>

<button onClick={() => this.setState({ view: 'blog' })}>Blog</button>

<button onClick={() => this.setState({ view: 'course' })}>Course</button>

{this.renderComponent()}

</div>

);

}

}

export default App;

BookDetails.js:

import React from 'react';

const BookDetails = () => (

<div>

<h2>Books</h2>

<ul>

<li>React Basics - Dan</li>

<li>React Hooks - Kent</li>

</ul>

</div>

);

export default BookDetails;

BlogDetails.js:

import React from 'react';

const BlogDetails = () => (

<div>

<h2>Blogs</h2>

<p>Understanding JSX - Nina</p>

<p>React Performance - Max</p>

</div>

);

export default BlogDetails;

CourseDetails.js:

import React from 'react';

const CourseDetails = () => (

<div>

<h2>Courses</h2>

<ul>

<li>ReactJS – 3 weeks</li>

<li>React with Redux – 2 weeks</li>

</ul>

</div>

);

export default CourseDetails;

### Expected Output:

