Digital Nurture 4.0 Deep Skilling

Week 7

Superset id: 6372485

Name: Sangamithra U

Hands on 1

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

1. List of Players

- 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. Indian Players

a. Display the Odd Team Player and Even Team players using the Destructuring features of ES6
b. Declare two arrays T20players and Ranji Trophy 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.

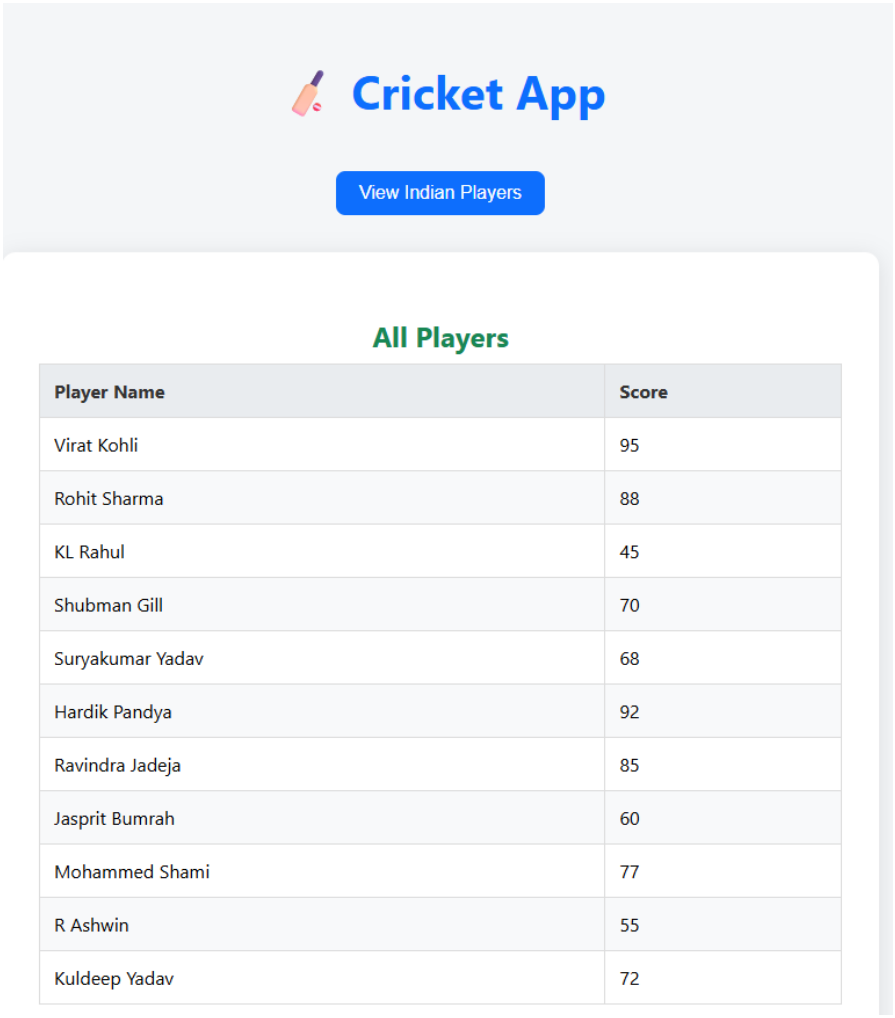
```jsx
import React, { useState } from 'react';
import ListofPlayers from './Components/ListofPlayers';
import IndianPlayers from './Components/IndianPlayers';
import './App.css';
import './Components/ComponentStyles.css';

function App() {
  const [flag, setFlag] = useState(true);

  return (
    <div className="App">
      <h1> Cricket App</h1>
      <button className="toggle-btn" onClick={() => setFlag(!flag)}>
        {flag ? 'View Indian Players' : 'View Player List'}
      </button>

      <div className="component-container">
        {flag ? <ListofPlayers /> : <IndianPlayers />}
      </div>
    </div>
  );
}

export default App;
```

```
const lowScorers = players.filter(player => player.score < 70);
return (
  <>
    <h2>All Players</h2>
    <table className="player-table">
      <thead>
        <tr>
          <th>Player Name</th>
          <th>Score</th>
        </tr>
      </thead>
      <tbody>
        {players.map((player, index) => (
          <tr key={index}>
            <td>{player.name}</td>
            <td>{player.score}</td>
          </tr>
        ))}
      </tbody>
    </table>
```

```
const IndianPlayers = () => {
  const team = ['Virat', 'Rohit', 'Rahul', 'Gill', 'Hardik', 'Jadeja', 'Bumrah', 'Shami']
  const oddPlayers = team.filter((_, index) => index % 2 === 0);
  const evenPlayers = team.filter((_, index) => index % 2 !== 0);
  const T20Players = ['Rohit', 'Gill', 'Surya'];
  const RanjiTrophyPlayers = ['Pujara', 'Rahane', 'Karun'];
  const allPlayers = [...T20Players, ...RanjiTrophyPlayers];
  return (
    <>
      <h2>Odd Team Players</h2>
      <ul className="player-list">
        {oddPlayers.map((player, index) => <li key={index}>{player}</li>)}
      </ul>

      <h2>Even Team Players</h2>
      <ul className="player-list">
        {evenPlayers.map((player, index) => <li key={index}>{player}</li>)}
      </ul>

      <h2>Merged T20 + Ranji Trophy Players</h2>
      <ul className="player-list">
        {allPlayers.map((player, index) => <li key={index}>{player}</li>)}
      </ul>
    </>
  );
};
```

Output :

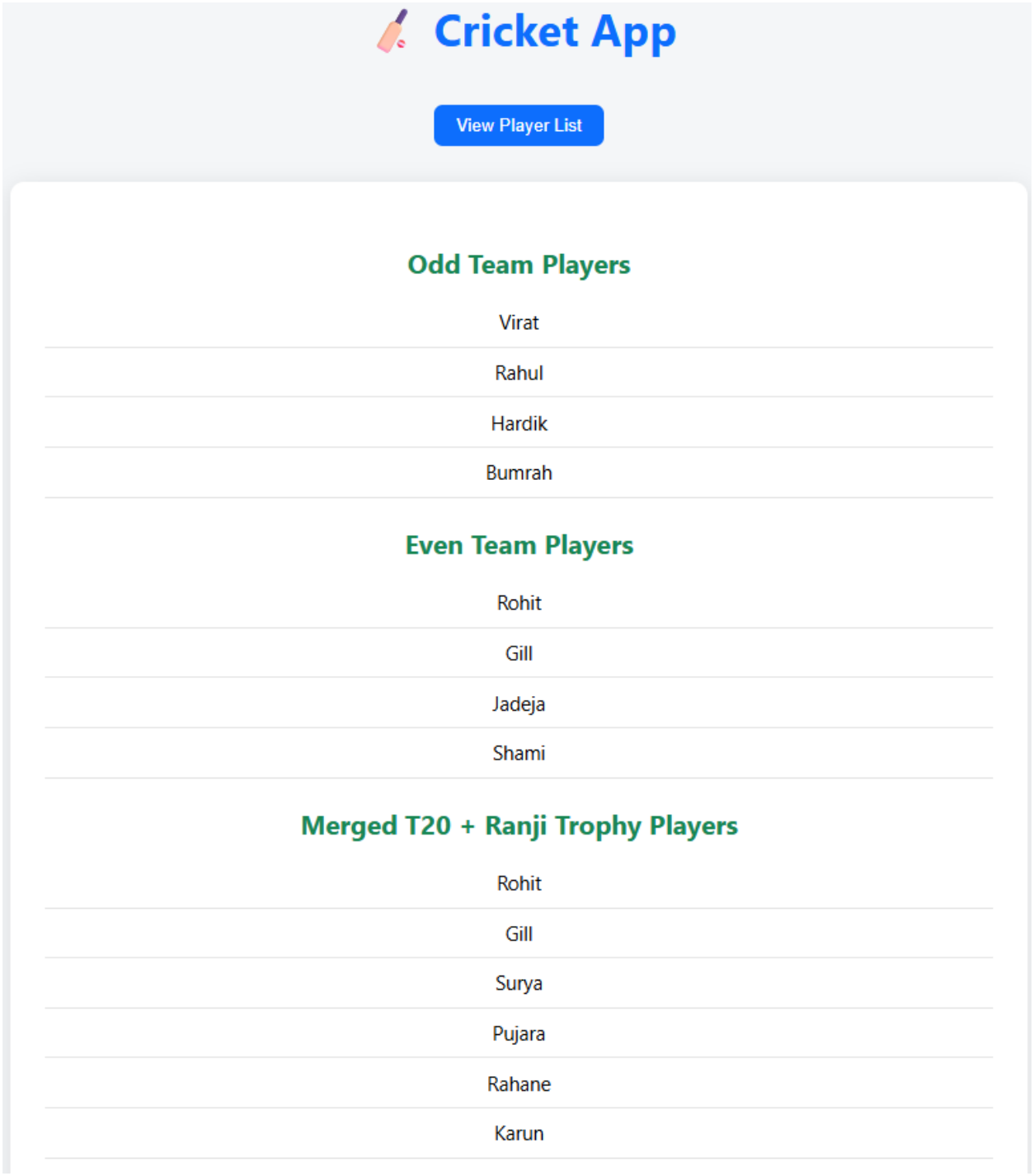
When flag = true



Players with Score Below 70

Player Name	Score
KL Rahul	45
Suryakumar Yadav	68
Jasprit Bumrah	60
R Ashwin	55

When flag=false



Hands on 2

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

a. Create an element to display the heading of the page.
b. Attribute to display the image of the office space
c. Create an object of office to display the details like Name, Rent and Address.
d. Create a list of Object and loop through the office space item to display more data.
e. To apply css, Display the color of the Rent in Red if it’s below 60000 and in Green if it’s above 60000.

```
const getRentStyle = (rent) => ({
  color: rent < 60000 ? 'red' : 'green',
  fontWeight: 'bold'
});

return (
  <div className="App">
    <h1>🏢 Office Space Rental App</h1>

    <h2>Available Spaces</h2>
    <div className="card-container">
      {officeList.map((item, index) => (
        <div className="card" key={index}>
          <img src={item.image} alt={item.name} />
          <h3>{item.name}</h3>
          <p><strong>Address:</strong> {item.address}</p>
          <p>
            <strong>Rent:</strong>{' '}
            <span style={getRentStyle(item.rent)}>₹{item.rent}</span>
          </p>
        </div>
      ))}
    </div>
  </div>
);
```

Output :

Office Space Rental App

Available Spaces

shutterstock.com · 1722517036

Innovate Space

Address: Koramangala, Bengaluru

Rent: ₹45000

Urban Office

Address: Andheri, Mumbai

Rent: ₹75000

Hands on 3

Create a React application named **"event examples app"** to handle various form events.

- Include an **"Increment"** button to increase a counter and a **"Decrement"** button to decrease it.

- The "Increment" button should invoke multiple methods:

  - One method to increment the counter value.
  - Another method to display a static message like **"Hello Member1"**.

- Create a **"Say Welcome"** button that invokes a function taking **"Welcome"** as an argument and displays it.

- Add a **"Click on me"** button that demonstrates a **synthetic event**, showing the message **"I was clicked!"** when triggered.

- Create a separate component named **"Currency Convertor"**.

  - This component should include two input fields: one for **amount in Indian Rupees** and another for the **target currency** (e.g., Euro).
  - Include a **"Submit"** button that, when clicked, triggers the **handle Submit** event.
  - The event handler should simulate or alert a conversion of the given amount from rupees to the specified currency.

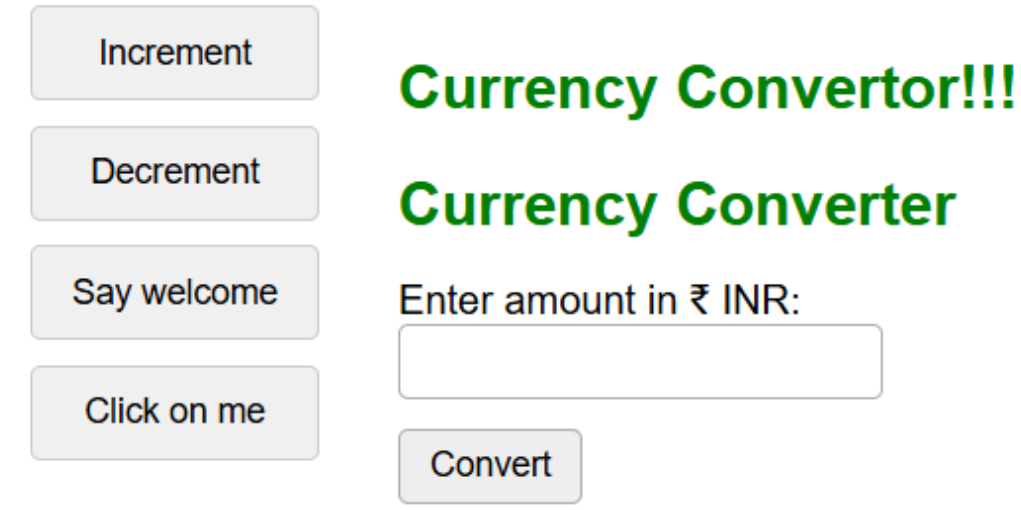
Code :

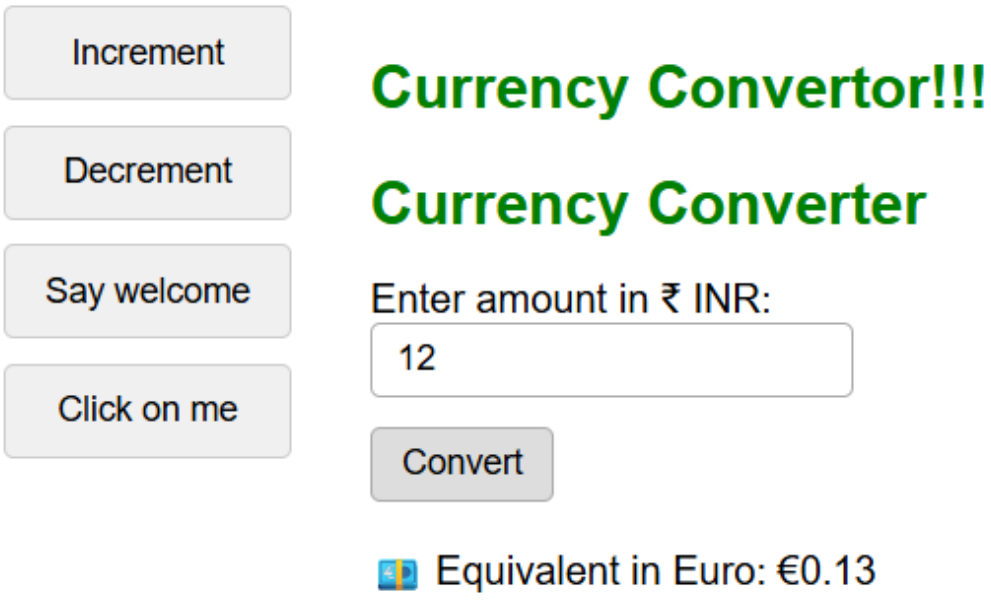
```
import React, { useState } from 'react';
import './App.css';
import CurrencyConverter from './CurrencyConverter';

function App() {
  const [count, setCount] = useState(0);

  const handleIncrement = () => {
    incrementValue();
    sayHello();
  };
  const incrementValue = () => {
    setCount(count + 1);
  };
  const sayHello = () => {
    alert("Hello Member1!");
  };
  const handleDecrement = () => {
    setCount(count - 1);
  };
  const sayWelcome = (msg) => {
    alert(`Hello! ${msg}`);
  };
  const handlePress = () => {
    alert("I was clicked!");
  };
```

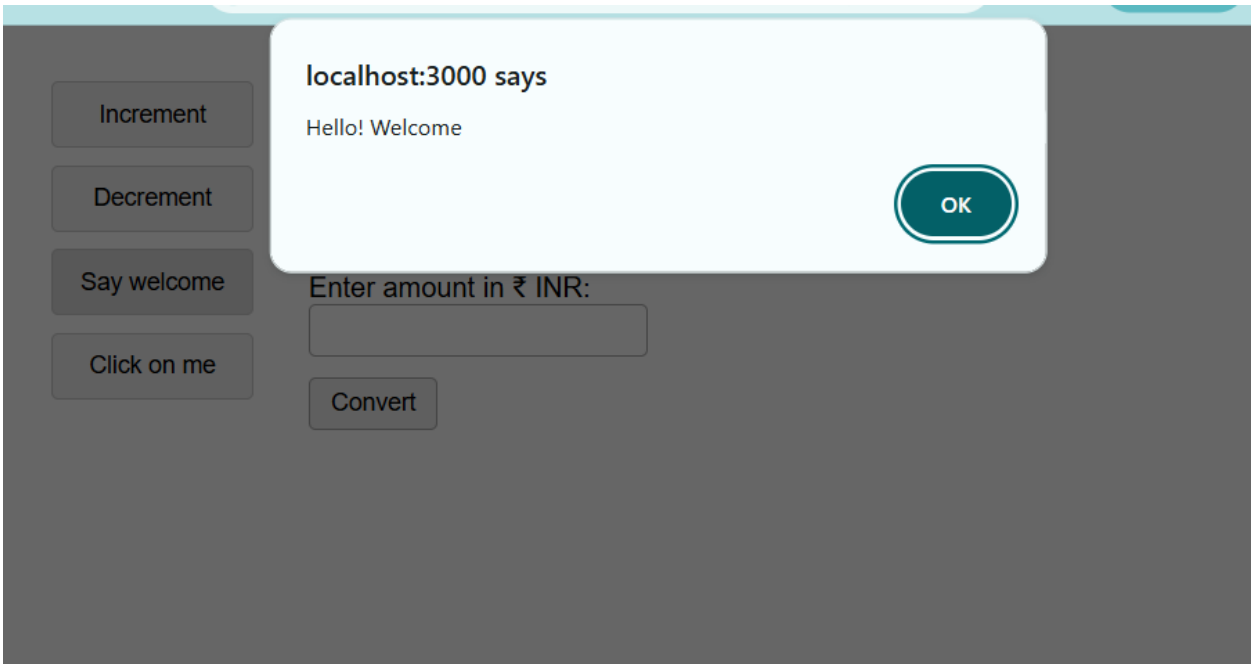
```
  return (
    <div className="App">
      <div className="left-panel">
        <button onClick={handleIncrement}>Increment</button>
        <button onClick={handleDecrement}>Decrement</button>
        <button onClick={() => sayWelcome("Welcome")}>Say welcome</button>
        <button onClick={handlePress}>Click on me</button>
      </div>

      <div className="currency-section">
        <h2>Currency Convertor!!!</h2>
        <CurrencyConverter />
      </div>
    </div>
  );
}

export default App;
```

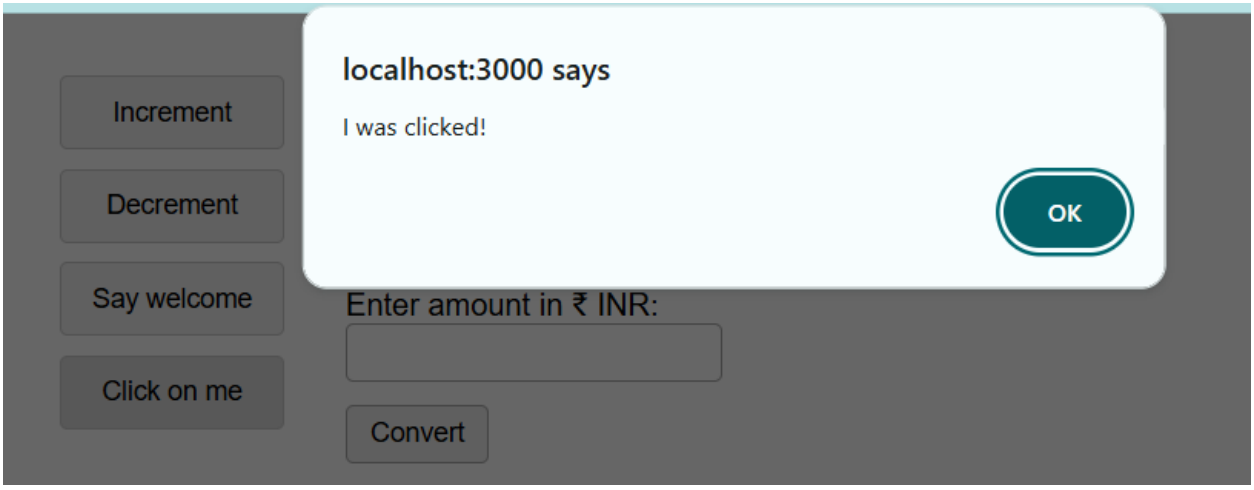
```
function CurrencyConverter() {
  const [rupees, setRupees] = useState('');
  const [euro, setEuro] = useState(null);
  const handleSubmit = (e) => {
    e.preventDefault();
    const rate = 0.011; // 1 INR = 0.011 Euro (example rate)
    const converted = (parseFloat(rupees) * rate).toFixed(2);
    setEuro(converted);
  };
  return (
    <div className="currency">
      <h2>Currency Converter</h2>
      <form onSubmit={handleSubmit}>
        <label>
          Enter amount in ₹ INR:
          <input
            type="number"
            value={rupees}
            onChange={(e) => setRupees(e.target.value)}
            required
          />
        </label>
        <button type="submit">Convert</button>
      </form>
      {euro && (
        <p>💶 Equivalent in Euro: €{euro}</p>
      )}
    </div>
```

Output :


Increment
Decrement
Say welcome
Click on me
Currency Convertor!!!
Currency Converter
Enter amount in ₹ INR:
Convert


Increment
Decrement
Say welcome
Click on me
Currency Convertor!!!
Currency Converter
Enter amount in ₹ INR:
12
Convert
Equivalent in Euro: €0.13


localhost:3000 says
Hello! Welcome
OK
Increment
Decrement
Say welcome
Click on me
Enter amount in ₹ INR:
Convert


localhost:3000 says
I was clicked!
OK
Increment
Decrement
Say welcome
Click on me
Enter amount in ₹ INR:
Convert

Hands on 4

Create a React Application named “ticket booking app” 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.

Code:

App.js M | LoginControl.js U | App.css M

src > App.js > ...

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

function App() {
  return (
    <div className="App">
      <LoginControl />
    </div>
  );
}

export default App;
```

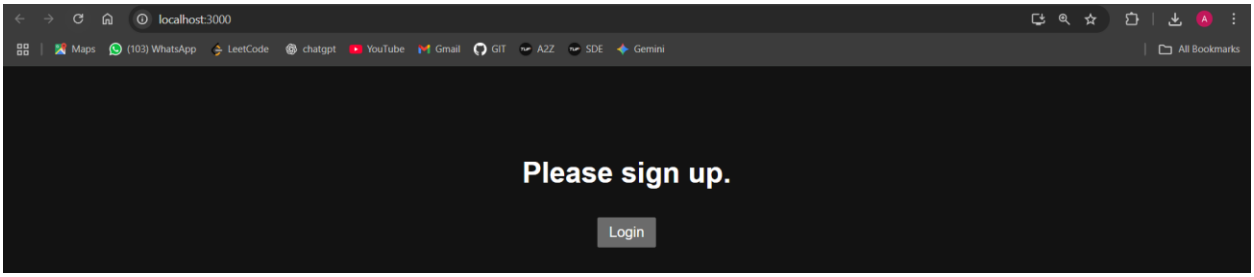
src > App.css > ...

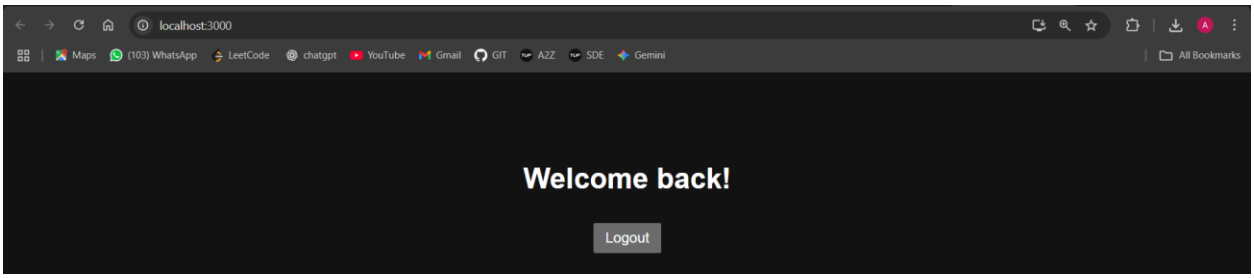
```
.App {
  text-align: center;
  margin-top: 100px;
  font-family: Arial, sans-serif;
}

button {
  font-size: 16px;
  padding: 6px 12px;
  margin-top: 10px;
  cursor: pointer;
}
```

src > JS LoginControl.js > LoginControl

```jsx
import React, { useState } from 'react';
function UserGreeting() {
  return <h1>Welcome back!</h1>;
}
function GuestGreeting() {
  return <h1>Please sign up.</h1>;
}
function Greeting({ isLoggedIn }) {
  if (isLoggedIn) {
    return <UserGreeting />;
  }
  return <GuestGreeting />;
}
function LoginButton({ onClick }) {
  return <button onClick={onClick}>Login</button>;
}
function LogoutButton({ onClick }) {
  return <button onClick={onClick}>Logout</button>;
}
function LoginControl() {
  const [isLoggedIn, setIsLoggedIn] = useState(false);
  const handleLoginClick = () => setIsLoggedIn(true);
  const handleLogoutClick = () => setIsLoggedIn(false);
  let button;
  if (isLoggedIn) {
    button = <LogoutButton onClick={handleLogoutClick} />;
  } else {
    button = <LoginButton onClick={handleLoginClick} />;
  }
  return (
    <div>
      <Greeting isLoggedIn={isLoggedIn} />
      {button}
    </div>
  );
}
```

Output :


localhost:3000
Maps
(103) WhatsApp
LeetCode
chatgpt
YouTube
Gmail
GIT
A2Z
SDE
Gemini
All Bookmarks
Please sign up.
Login


localhost:3000
Maps
(103) WhatsApp
LeetCode
chatgpt
YouTube
Gmail
GIT
A2Z
SDE
Gemini
All Bookmarks
Welcome back!
Logout

Hands on 5

Estimated time to complete this lab: **60 minutes.**

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.

Code :

```
import React from 'react';
import './App.css';
import CourseDetails from './components/CourseDetails';
import BookDetails from './components/BookDetails';
import BlogDetails from './components/BlogDetails';

function App() {
  return (
    <div className="App">
      <div className="container">
        <div className="column">
          <CourseDetails />
        </div>

        <div className="divider"></div>

        <div className="column">
          <BookDetails />
        </div>

        <div className="divider"></div>

        <div className="column">
          <BlogDetails />
        </div>
      </div>
    </div>
  );
}

export default App;
```

src > # App.css > ...

```css
.App {
  font-family: Arial, sans-serif;
  padding: 40px;
}

.container {
  display: flex;
  justify-content: space-between;
  align-items: flex-start;
  text-align: left;
}

.column {
  flex: 1;
  padding: 0 20px;
}

.divider {
  width: 4px;
  background-color: green;
  height: auto;
}

```

src > components > JS CourseDetails.js > ...

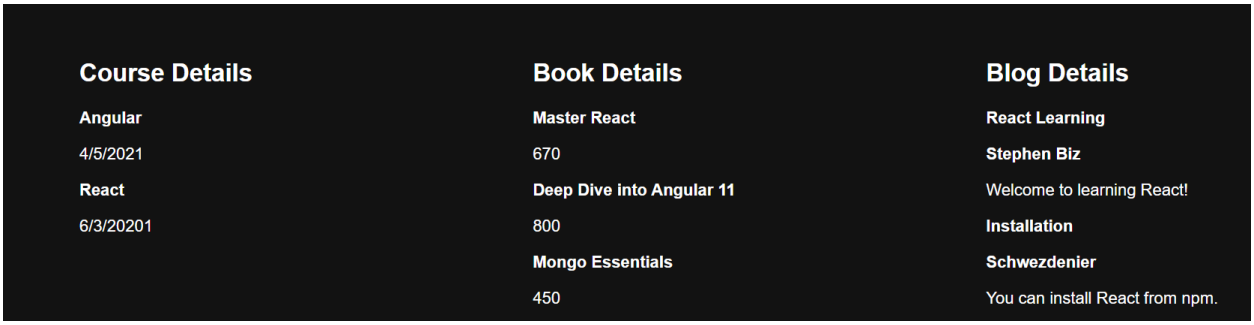
```jsx
function CourseDetails() {
  return (
    <div className="column">
      <h2>Course Details</h2>
      {courses.map((course, index) => (
        <div key={index}>
          <strong>{course.name}</strong>
          <p>{course.date}</p>
        </div>
      ))}
    </div>
  );
}

export default CourseDetails;
```

```jsx
import React from 'react';

function BookDetails() {
  const books = [
    { title: "Master React", price: 670 },
    { title: "Deep Dive into Angular 11", price: 800 },
    { title: "Mongo Essentials", price: 450 }
  ];

  return (
    <div className="column">
      <h2>Book Details</h2>
      {books.map((book, index) => (
        <div key={index}>
          <strong>{book.title}</strong>
          <p>{book.price}</p>
        </div>
      ))}
    </div>
  );
}

export default BookDetails;
```

src > components > JS BlogDetails.js > ...

```jsx
import React from 'react';

function BlogDetails() {
  const blogs = [
    {
      title: "React Learning",
      author: "Stephen Biz",
      content: "Welcome to learning React!"
    },
    {
      title: "Installation",
      author: "Schwezdenier",
      content: "You can install React from npm."
    }
  ];

  return (
    <div className="column">
      <h2>Blog Details</h2>
      {blogs.map((blog, index) => (
        <div key={index}>
          <strong>{blog.title}</strong>
          <p><b>{blog.author}</b></p>
          <p>{blog.content}</p>
        </div>
      ))}
    </div>
  );
}

export default BlogDetails;
```

Output :


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.