REACT

Commands to install in ubuntu:-

- sudo apt update
- sudo apt upgrade
- sudo apt install nodejs npm
- node -v
- npm -v
- curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
- sudo apt install nodejs
- sudo apt install build-essential'
- npx create-react-app hello-world

Flow of React Application:-

1. index.html (Entry Point)

📌 **Location:** public/index.html

- This is the **base HTML file** that loads when you run the app in a browser.

Key Part of index.html:-

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>React App</title>
</head>
<body>
  <div id="root"></div> <!-- React app will be injected here -->
</body>
</html>
```

- The acts as a **container** for the React app.
- React **does not modify the entire HTML file**, it just injects components inside #root.

---

2. index.js (Entry Point of React)

📌 **Location:** src/index.js

This is the main JavaScript file that **connects React to the browser**.

Code in index.js

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

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

reportWebVitals();
```

Imports React & ReactDOM:

- React: Enables React functionality.
- ReactDOM: Renders React components into the browser.

Gets the #root Element:

```
const root = ReactDOM.createRoot(document.getElementById('root'));
```

- Finds in index.html.

Renders the <App /> Component:

```
root.render(
   <React.StrictMode>
     <App />
   </React.StrictMode>
 );
```

- **Injects** the App component inside .
- **Uses React.StrictMode** (for catching potential issues in development).

3. App.js (Main Component)

📌 **Location:** src/App.js

This file defines the **main UI structure** of the application.

Code in App.js

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

function App() {
  return (
    <div className="App">
      <h1>Welcome to React!</h1>
    </div>
  );
}

export default App;
```

Defines a Component:
function App() { return (...); }

- This is a **functional component** that returns JSX (HTML-like syntax).

Exports the Component:
export default App;

- This allows App to be imported in index.js.

Flow of Execution

- **Browser loads index.html**
    - It contains .

- **index.js runs**
    - It finds the #root element.

It injects the App component into #root using:
root.render(<App />);

- **App.js renders UI**

The App component returns:

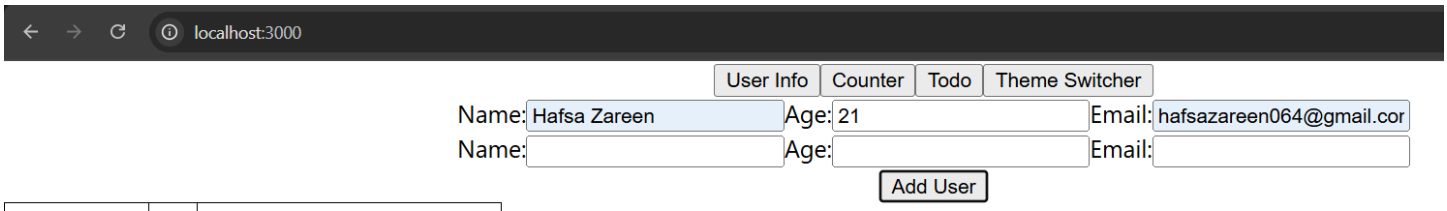
  <h1>Welcome to React!</h1>

Diagram Representation 🖼️

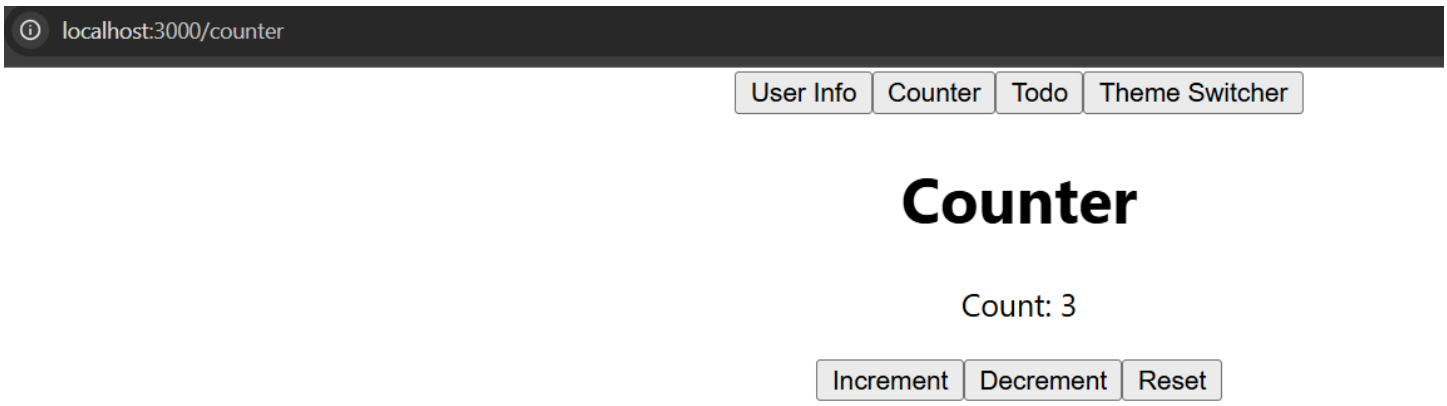
1. index.html
   └──   (Empty at start)

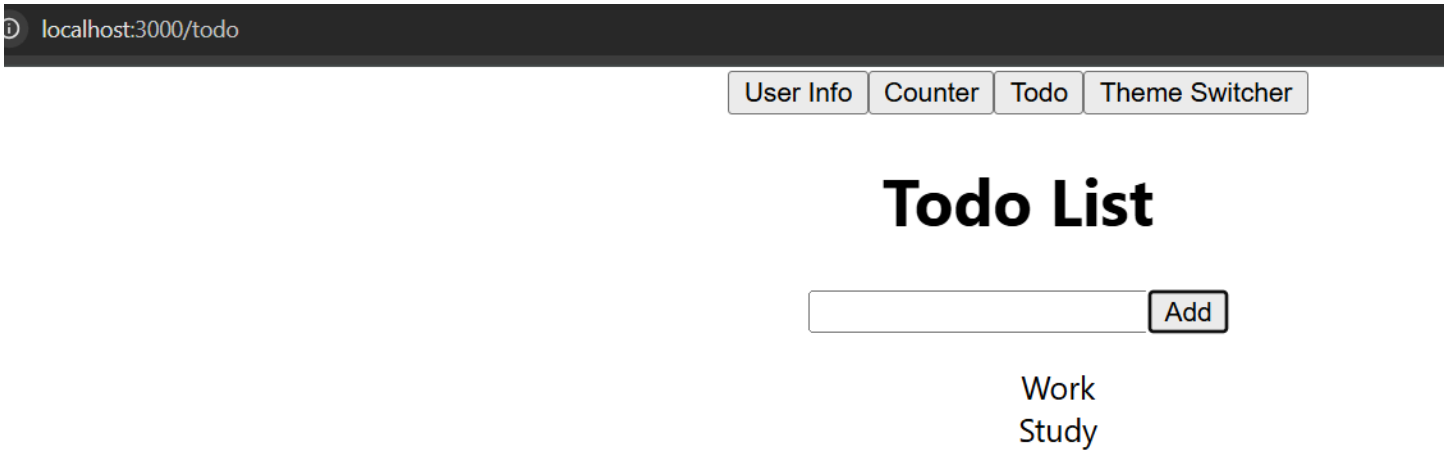
2. index.js
   ├── Gets #root
   ├── Renders <App /> inside #root

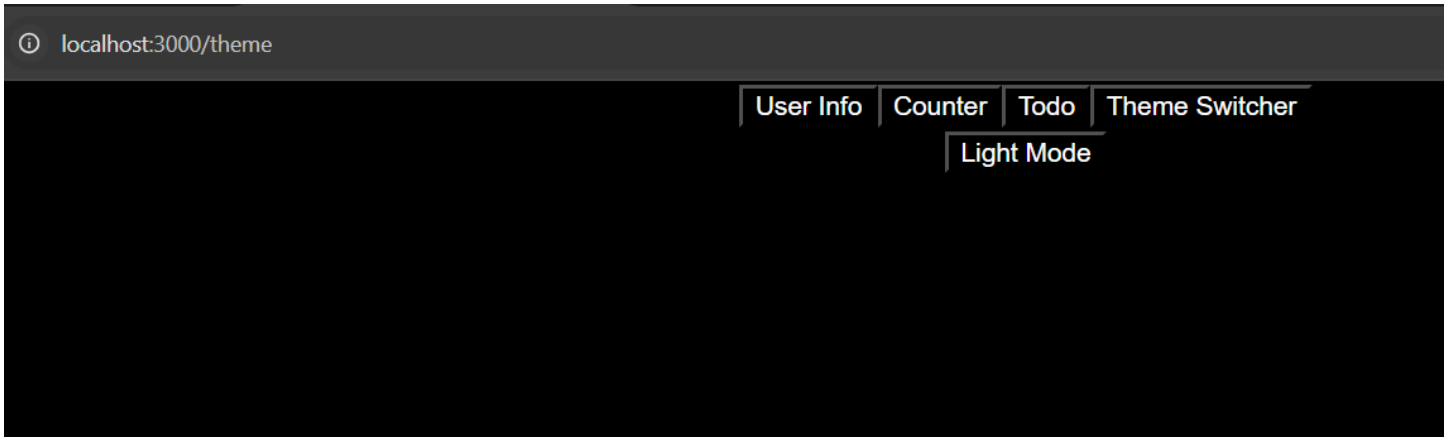
3. App.js
   ├── Defines UI
   ├── Returns JSX (<h1>Welcome to React</h1>)
   ├── Displayed in the browser



Name	Age	Email
Hafsa Zareen	21	hafsazareen064@gmail.com







App.js:-

```
import './App.css';
import { BrowserRouter as Router, Route, Routes, Link } from 'react-router-dom';
import ThemeSwitcher from './Components/ThemeSwitcher';
import Counter from './Components/Counter';
import Todo from './Components/todo';
import UserInfo from './Components/UserInfo';

function App() {
  return (
    <Router>
      <div className="App">
        {/* Navigation Buttons */}
        <nav>
          <Link to="/">
            <button>User Info</button>
          </Link>
          <Link to="/counter">
            <button>Counter</button>
          </Link>
          <Link to="/todo">
            <button>Todo</button>
          </Link>
          <Link to="/theme">
            <button>Theme Switcher</button>
          </Link>
        </nav>

        {/* Routes */}
        <Routes>
          <Route path="/" element={<UserInfo />} />
          <Route path="/counter" element={<Counter />} />
          <Route path="/todo" element={<Todo />} />
          <Route path="/theme" element={<ThemeSwitcher />} />
        </Routes>
      </div>
    </Router>
  );
}

export default App;
```

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

function UserInfo() {
 const [users, setUsers] = useState([
  { name: '', age: '', email: '' },
 ]);

 const [tableStyle, setTableStyle] = useState({
  border: '1px solid black',
  borderColor: 'black',
 });

 const handleChange = (index, e) => {
  const { name, value } = e.target;
  const updatedUsers = [...users];
  updatedUsers[index][name] = value;
  setUsers(updatedUsers);
 };

 const handleStyleChange = (e) => {
  const { name, value } = e.target;
  setTableStyle({ ...tableStyle, [name]: value });
 };

 const addUser = () => {
  setUsers([...users, { name: '', age: '', email: '' }]);
 };


 return (
  <div>

   {users.map((user, index) => (
    <div key={index}>
     <label>Name:</label>
     <input
      type="text"
      name="name"
      value={user.name}
      onChange={(e) => handleChange(index, e)}
     />
     <label>Age:</label>
```

```
        <input
          type="number"
          name="age"
          value={user.age}
          onChange={(e) => handleChange(index, e)}
        />
        <label>Email:</label>
        <input
          type="email"
          name="email"
          value={user.email}
          onChange={(e) => handleChange(index, e)}
        />
      </div>
    ))}

    <button onClick={addUser}>Add User</button>

    <table
      style={{
        border: tableStyle.border,
        borderColor: tableStyle.borderColor,
        borderCollapse: 'collapse',
      }}
    >
      <thead>
        <tr>
          <th style={{border: tableStyle.border}}>Name</th>
          <th style={{border: tableStyle.border}}>Age</th>
          <th style={{border: tableStyle.border}}>Email</th>
        </tr>
      </thead>
      <tbody>
        {users.map((user, index) => (
          <tr key={index}>
            <td style={{border: tableStyle.border}}>{user.name}</td>
            <td style={{border: tableStyle.border}}>{user.age}</td>
            <td style={{border: tableStyle.border}}>{user.email}</td>
          </tr>
        ))}
      </tbody>
    </table>
  </div>
 );
}
```

```
export default UserInfo;
```

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

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

  useEffect(() => {
    // Access the document body to change the background color
    document.body.style.backgroundColor = theme === 'light' ? 'white' : 'black';
    document.body.style.color = theme === 'light' ? 'black' : 'white';
  }, [theme]); // Re-run effect when theme changes

  const toggleTheme = () => {
    setTheme(theme === 'light' ? 'dark' : 'light');
  };

  return (
    <button onClick={toggleTheme}>
      {theme === 'light' ? 'Dark Mode' : 'Light Mode'}
      <style jsx>{`
        button {
          background-color: ${theme === 'light' ? 'white' : 'black'};
          color: ${theme === 'light' ? 'black' : 'white'};
        }
      `}</style>
    </button>
  );
}

export default ThemeSwitcher;
```

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

const Todo = () => {
    const [todos, setTodos] = useState([]);
    const [input, setInput] = useState('');

    const addTodo = () => {
        // Spreads (...todos) the existing todos array and adds the new input value at the end
        setTodos([...todos, input]);
        // Clears the input field after adding a todo.
        setInput('');
    }
```

```
    return (
        <div>
            <h1>Todo List</h1>
            <input type="text" value={input} onChange={(e) => setInput(e.target.value)} />
            {/* onChange={(e) => setInput(e.target.value) Updates input state whenever the user types. */}
            <button onClick={addTodo}>Add</button>
            <ul>
                {/* <ul> → Creates an unordered list. */}
                {todos.map((todo, index) => (
                    <li key={index}>{todo}</li>
                ))}
                {/*
Loops through the todos array and creates <li> elements for each todo.
key={index} → React requires a unique key for each list item. */}

            </ul>
        </div>
    )

}

export default Todo;
```

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

const Counter = () => {
//     useState(0) → Initializes a state variable named count with an initial value of 0.
// count → Holds the current value of the counter.
// setCount → A function that updates the count state.
    const [count, setCount] = useState(0);
    return (
        <div>
            <h1>Counter</h1>
            <p>Count: {count}</p>
            <button onClick={() => setCount(count + 1)}>Increment</button>
            <button onClick={() => setCount(count - 1)}>Decrement</button>
            <button onClick={() => setCount(0)}>Reset</button>

        </div>
    )
}
```

```
export default Counter;
```

- In React, **state** is an object that stores **dynamic data** and determines how a component behaves. When the state changes, the component **re-renders** automatically.

- It captures the **change event** when the user types in the input field.