**Name : TEJAS J PATEL**

**ROLL NO : 07**

**CLASS : MSC C.A-1**

**SUBJECT : 102-WEB DEVLOPMENT AND DESIGN**

**File Programs**

**1. Create a new react app using Different Components.**

**1. Components → Header.js**

import React from "react";

function Header() {

return (

<header>

<p>This is a header. - Rendering from Header component. </p>

<img src="<http://placekitten.com/400/300>" height={150} width={150} />

<p>Story</p>

</header> ) }

export default Header

**2. Components → Footer.js**

import React from "react";

function Footer() { return(

<footer>

<p>This is a footer - Rendering from Footer component</p>

<p>Reels</p>

</footer> ) }

export default Footer;

**3. Components → Main.js**

import React from "react";

function Main() { return(

<main>

<p>This is a main body - Rendering from Main component</p>

<p>Post</p>

</main> ) }

export default Main;

**4. App.js**

import './App.css';

import Header from './Components/Header';

import Main from './Components/Main';

import Footer from './Components/Footer';

function App() {

return ( <div>

<Header />

<Main />

<Footer />

</div> ); }

export default App;

**2. Create new react app for Counter using usestate hooks, Greeting using props, Action.**

1. **Counter.js**

import React, { useState } from 'react'

function Counter() {

const [count, setCount] = useState(100);

const [count1, getCount] = useState(1);

function increment() { setCount(count + 20); }

function multiplication() { getCount(count1 \* 5); }

return ( <div> <p>Count: {count}</p> <br /> <p>Product: {count1}</p> <button onClick={increment}>Increment</button> <br /> <br /> <button onClick={multiplication}>Multiplication</button> </div> ) }2.

export default Counter

1. **Greeting.js**

import React from 'react'

function Greeting(props) { return (

<div>

Your full name is {props.firstName} {props.lastName} and you salary is {props.salary}

</div>

) }

export default Greeting

1. **Action.js**

import React from 'react'

function Action() {

function clickButton() { alert('You have clicked the button') }

return ( <div> <button onClick={clickButton}>Click Me !</button> </div> ) }

export default Action;

1. **App.js**

import React from 'react'

import Greeting from './Components/Greeting'

import Action from './Components/Action';

import Counter from './Components/Counter';

function App() { const fN = "Patel"; const lN = "Chirag" const sal = 1000;

return (

<div>

<Greeting firstName={fN} lastName={lN} salary = {sal} />

<br />

<Action />

<br />

<Counter />

</div>

) }

export default App

**3. Create new react app using simpleclass, classConstructor ,classComponents.**

**1.ClassConstructor.js**

import React from "react";

class ClassComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

id : 100,

name : 'Om'

}

};

render() {

const {id, name} = this.state;

return(

<div>

<p>Hi {name}</p>

<p>your id {id}</p>

</div>

);

}

}

export default ClassComponent;

/\*import React from "react";

class ClassComponent extends React.Component { constructor(props) { super(props); this.state = { count: 0, name : 'Jhon', }; };

componentDidMount() {

this.interval = setInterval(() => {

this.setState({

time: new Date().toLocaleTimeString(),

});

}, 1000);

}

render() {

const {count, name} = this.state;

return(

<div>

<p>Class Component Example:</p>

<p>Count: {count}</p>

<p>Name : {name}</p>

<p>Current time: {this.state.time}</p>

</div>

);

}

}

export default ClassComponent;\*/

/\*import React from 'react';

class Timer extends React.Component {

constructor(props) {

super(props);

this.state =

{ time: 0, }; }

componentDidMount() {

this.timerId = setInterval(() => { this.setState(prevState => ({ time: prevState.time + 1, })); }, 1000); }

componentWillUnmount() {

clearInterval(this.timerId); }

render() { return (

<div>

<h2>Timer</h2>

<p>Time: {this.state.time}</p>

</div>

); } }\*/

**2. ClassConstructor.js**

import React from 'react';

class ClassConstructor extends React.Component {

constructor(props) {

super(props);

this.state = { id : 100, name: 'Jhon', } };

render() {

const {name} = this.state;

return ( <div>

<h1>Hello, {name}</h1>

</div>

); } }

export default ClassConstructor;

**3.SimpleClass.js**

import React from "react";

class SimpleClass extends React.Component {

render() {

return(

<p>JSX statements</p>

) } }

export default SimpleClass;

/\*class SimpleClass extends React.Component {

render() { return (

<div>

<h1>Hello, Class Component!</h1>

<p>This is a simple class component.</p>

</div> ); } }

export default SimpleClass;\*/

1. **App.js**

import './App.css';

import SimpleClass from './SimpleClass';

import ClassConstructor from './ClassConstructor';

import ClassComponents from './ClassComponents';

function App() {

return (

<div>

<SimpleClass />

<ClassConstructor />

<ClassComponents />

</div> ); }

export default App;

**4. Create new react app for githubuser .**

**1.GithubUser.js**

import React, { Component } from 'react';

class GitHubUser extends Component {

constructor(props) {

super(props);

this.state = {

user: null,

error: null

};

}

componentDidMount() {

fetch('<https://api.github.com/users/codeNinja>')

.then(response => {

if (!response.ok) {

throw new Error('Error fetching user data from GitHub API');

}

return response.json();

})

.then(user => {

this.setState({ user });

})

.catch(error => {

this.setState({ error: error.message });

});

}

render() {

const { user, error } = this.state;

if (error) {

return <p>Error: {error}</p>;

}

if (!user) {

return <p>Loading...</p>;

}

return (

<div>

<h2>GitHub User</h2>

<p>Username: {user.login}</p>

<p>Name: {user.name}</p>

<p>AvatarURL: {user.avatar\_url}</p>

<img src={user.avatar\_url} height={250} width={250} />

<p>Location: {user.location}</p>

<p>Public Repositories: {user.public\_repos}</p>

<p>Type: {user.type}</p>

{/\* Add more desired user information \*/}

</div>

);

}

}

export default GitHubUser;

**2.App.js**

import './App.css'; import GitHubUser from './GitHubUser';

function App() {

return ( <div> <GitHubUser /> </div> ); }

export default App;

**5.Create a new react app for Timer.**

**1.Timer.js**

import './App.css'; import Timer from './Timer';

function App() {

return ( <div> <Timer /> </div> ); }

export default App;

export default Timer

}

componentDidMount() {

this.timerId = setInterval(() => {

this.setState(prevState => ({

time: prevState.time + 1,

}));

}, 1000);

}

componentWillUnmount() {

clearInterval(this.timerId);

}

render() {

return (

<div>

<h2>Timer</h2>

<p>Time: {this.state.time}</p>

</div>

);

}

**2. App.js:**

import './App.css';

import Timer from './Timer ';

function App() {

return (

<div>

<Timer />

</div>

); }

export default App;

1. **Create a new React app for react-simple Portals.**
2. **App.js:**

import logo from './logo.svg';

import './App.css';

import Portal from './Portal';

function App() {

return (

<div className="App">

<h1>Root</h1>

<Portal />

</div>

);

}

export default App;

1. **Index.html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<meta

name="description"

content="Web site created using create-react-app"

/>

<link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

<!--

manifest.json provides metadata used when your web app is installed on a

user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/

-->

<link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

<!--

Notice the use of %PUBLIC\_URL% in the tags above.

It will be replaced with the URL of the `public` folder during the build.

Only files inside the `public` folder can be referenced from the HTML.

Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will

work correctly both with client-side routing and a non-root public URL.

Learn how to configure a non-root public URL by running `npm run build`.

-->

<title>React App</title>

</head>

<body>

<div id="root"></div>

<div id="portal-root"></div>

<!--

This HTML file is a template.

If you open it directly in the browser, you will see an empty page.

You can add webfonts, meta tags, or analytics to this file.

The build step will place the bundled scripts into the <body> tag.

To begin the development, run `npm start` or `yarn start`.

To create a production bundle, use `npm run build` or `yarn build`.

-->

</body>

</html>

1. **Portal.js**

import React from 'react'

import ReactDOM from 'react-dom'

function Portal() {

return ReactDOM.createPortal(

<div style={{border:"1px solid red"}}>Portal</div>

, document.getElementById('portal-root'))

}

export default Portal

/\*

import React from 'react'

import ReactDOM from 'react-dom'

function Portal() {

return ReactDOM.createPortal(

<div>

Portal

</div>

, document.getElementById('portal-root')

)

}

export default Portal\*/

1. **Create A new react app for render-list.**

**1.App.js:**

import logo from './logo.svg';

import './App.css';

import Render from './Components/Render';

function App() {

return (

<div className="app">

<Render />

</div>

);

}

export default App;

**2. Render.js:**

import React from "react";

import ReactDOM from "react-dom";

class Render extends React.Component {

country = ['India', 'USA', 'Australia', 'Japan', 'Germany']

render() {

return(

<div>

{this.country.map((country, index) => {

return <h2 key={index}>{country}</h2>

})}

</div>

)

}

}

export default Render;

/\*

function Render(){

const countries = ['India', 'Austraila', 'Sri Lanka', 'Nepal' ];

return(

<div>

{countries.map((country, index) => {

return <h1 key={index}>{country}</h1>

})}

</div>

)

}

export default Render

import React from "react";

\*/

1. **Create a new react app for Model.**

**1.App.js:**

import logo from './logo.svg';

import './App.css';

import Modal from './Modal';

import React, { Children } from 'react';

import ReactDOM from 'react-dom';

class App extends React.Component {

constructor(props) {

super(props);

this.state = {

showModal : false,

}

}

openModal = () => {

this.setState({showModal : true})

}

closeModal = () => {

this.setState({showModal : false})

}

render() {

const { showModal } = this.state;

let modalContent = null;

if(showModal) {

modalContent = (

<Modal children={modalContent}>

<h2>Child Modal</h2>

<p>This is child modal of React Portal Example</p>

<p>This is child modal of React Portal Example</p>

<p>This is child modal of React Portal Example</p>

<input type='checkbox' />

<p>This is checkbox</p>

<button onClick={this.closeModal}>Close Modal</button>

</Modal>

)

}

return(

<div className="App">

<h1>React Portal Example</h1>

<button onMouseOver={this.openModal}>Open Modal</button>

{modalContent}

</div>

)

}

}

export default App

**2. 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);

}

}

1. **Index.js:**

import React from 'react';

import ReactDOM from 'react-dom';

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>

);

// If you want to start measuring performance in your app, pass a function

// to log results (for example: reportWebVitals(console.log))

// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals

reportWebVitals();

1. **Index .html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<meta

name="description"

content="Web site created using create-react-app"

/>

<link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

<!--

manifest.json provides metadata used when your web app is installed on a

user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/

-->

<link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

<!--

Notice the use of %PUBLIC\_URL% in the tags above.

It will be replaced with the URL of the `public` folder during the build.

Only files inside the `public` folder can be referenced from the HTML.

Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will

work correctly both with client-side routing and a non-root public URL.

Learn how to configure a non-root public URL by running `npm run build`.

-->

<title>React App</title>

</head>

<body>

<noscript>You need to enable JavaScript to run this app.</noscript>

<div id="root"></div>

<div id="modal-root"></div>

<!--

This HTML file is a template.

If you open it directly in the browser, you will see an empty page.

You can add webfonts, meta tags, or analytics to this file.

The build step will place the bundled scripts into the <body> tag.

To begin the development, run `npm start` or `yarn start`.

To create a production bundle, use `npm run build` or `yarn build`.

-->

</body>

</html>

1. **Index.css:**

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

}

.app {

text-align: center;

}

1. **Model.js:**

import React, { Children } from 'react'

import ReactDOM from 'react-dom'

import './Modal.css'

function Modal({children}) {

const modalRoot = document.getElementById('modal-root');

return ReactDOM.createPortal(

<div className='modal-overlay'>

<div className='modal'>

<div className='modal-content'>

{children}

</div>

</div>

</div>

, modalRoot

)

}

export default Modal;

**7.Model.css:**

.modal-overlay {

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

background-color: rgba(0, 0, 0, 0.5);

display: flex;

align-items: center;

justify-content: center;

}

.modal {

background-color: #12d2dd;

padding: 20px;

border-radius: 5px;

}

.modal-content {

margin-bottom: 10px;

}

button {

cursor: pointer;

}

1. **Create a new react app for react-router and recat-nested-router.**
2. **About.css:**

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

/\* Apply some basic styles to the body \*/

body {

margin: 0;

padding: 0;

font-family: Arial, sans-serif;

line-height: 1.6;

background-color: #f9f9f9;

}

/\* Style the header and navigation \*/

header {

background-color: #1a78c4;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

.login-menu {

display: flex;

gap: 10px;

}

.login-item {

color: white;

cursor: pointer;

text-decoration: none;

}

nav {

display: flex;

justify-content: space-between;

align-items: center;

}

nav a {

color: #fff;

text-decoration: none;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

/\* Style the main content area \*/

.about-us {

max-width: 800px;

margin: 20px auto;

padding: 20px;

background-color: #fff;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);

}

.about-content h1 {

font-size: 36px;

margin-bottom: 10px;

}

.about-content p {

margin-bottom: 20px;

}

/\* Style the footer \*/

footer {

background-color: #1a78c4;

color: #fff;

padding: 10px;

text-align: center;

}

footer p {

margin: 0;

}

/\* Responsive styles \*/

@media screen and (max-width: 600px) {

.about-us {

padding: 10px;

}

}

1. **About.js:**

import React from 'react'

import { Link } from 'react-router-dom'

import './About.css'

function About() {

return (

<div>

<header>

<div className='login-menu'>

<Link className='login-item' to='/about/abouthistory' >About History</Link>

<Link className='login-item' to='/about/aboutoverview' >About Overview</Link>

<Link className='login-item' to='/about/aboutteam' >About Team</Link>

<Link className='login-item' to='/' >Back to Home page</Link>

</div>

</header>

<main class="about-us">

<section class="about-content">

<h1>About Us</h1>

<p>

Welcome to our application! We are a team of passionate developers dedicated to creating awesome web applications that make a difference. Our mission is to provide seamless and user-friendly experiences to our users.

</p>

<p>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur posuere ultrices sapien ut venenatis. Nulla ut malesuada nisl. Sed finibus nulla sed odio iaculis, et tincidunt mi consectetur. Maecenas congue aliquam ligula non eleifend. Quisque rhoncus risus vel elit fermentum dignissim.

</p>

<p>

Integer non mi id elit ullamcorper venenatis. Proin ultricies id lectus nec semper. Sed dictum odio non nunc luctus, non luctus massa rhoncus. Nullam congue euismod felis vel tincidunt. Cras at neque ut odio feugiat pharetra.

</p>

</section>

</main>

<footer>

<div>

1. **AboutHistory.js**

import React from 'react'

function AboutHistory() {

return (

<div>

<h1>AboutHistory</h1>

</div>

)

}

export default AboutHistory

1. **AboutOverview.js**

import React from 'react'

function AboutOverview() {

return (

<div>

<h1>AboutOverview</h1>

</div>

)

}

export default AboutOverview

1. **AboutTeam.js:**

import React from 'react'

function AboutTeam() {

return (

<div>

<h1>AboutTeam</h1>

</div>

)

}

export default AboutTeam

1. **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);

}

}

1. **App.js**

import './App.css';

import { Routes, Route } from 'react-router-dom'

/\*

It used for defining and rendering routes in a React application

with client-side routing.

\*/

import Home from './Home';

import About from './About';

import Contact from './Contact';

import AboutHistory from './AboutHistory';

import AboutOverview from './AboutOverview';

import AboutTeam from './AboutTeam';

function App() {

return (

/\*

The below code defines routes for different paths

and their corresponding components to be rendered using

client-side routing in a React application.

\*/

<Routes>

<Route path='/' Component={Home} />

<Route path='/about' Component={About} />

<Route path='/contact' Component={Contact} />

<Route path='/about/abouthistory' Component={AboutHistory} />

<Route path='/about/aboutoverview' Component={AboutOverview} />

<Route path='/about/aboutteam' Component={AboutTeam} />

</Routes>

)

}

export default App;

1. **Contact.css:**

.con {

max-width: 600px;

margin: 0 auto;

padding: 20px;

}

.h1 {

text-align: center;

}

.form-group {

margin-bottom: 20px;

}

label {

display: block;

font-weight: bold;

}

input[type="text"],

input[type="email"],

textarea {

width: 100%;

padding: 8px;

font-size: 16px;

border: 1px solid #ccc;

border-radius: 4px;

}

textarea {

resize: vertical;

}

button {

background-color: #4CAF50;

color: white;

padding: 10px 20px;

border: none;

border-radius: 4px;

cursor: pointer;

font-size: 16px;

display: block;

margin: 0 auto;

}

1. **Contact.js**

import React from "react";

import './Contact.css';

import { Link } from 'react-router-dom'

function Contact() {

return(

<div>

<header>

<div className='login-menu'>

<Link className='login-item' to='/' >Back to Home page</Link>

</div>

</header>

<div class="con">

<center><h1>Contact Us</h1></center>

<form>

<div class="form-group">

<label for="name">Name</label>

<input type="text" id="name" name="name" required />

</div>

<div class="form-group">

<label for="email">Email</label>

<input type="email" id="email" name="email" required />

</div>

<div class="form-group">

<label for="message">Message</label>

<textarea id="message" name="message" rows="5" required></textarea>

</div>

<button type="submit">Submit</button>

</form>

</div>

</div>

)

}

export default Contact;

1. **Home.css**

body {

margin: 0;

padding: 0;

font-family: Arial, sans-serif;

}

header {

background-color: #1a78c4;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

.logo {

color: white;

font-size: 24px;

font-weight: bold;

display: flex;

justify-content: center;

align-items: center;

}

.login-menu {

display: flex;

gap: 10px;

}

.login-item {

color: white;

cursor: pointer;

text-decoration: none;

}

.footer {

background-color: #1a78c4;

color: #fff;

padding: 10px;

text-align: center;

}

.container {

display: flex;

flex-wrap: wrap;

justify-content: center;

}

.box {

width: 350px;

height: 250px;

margin: 5px;

display: flex;

justify-content: center;

align-items: center;

background-color: #2cf546;

border: 1px solid #ccc;

font-weight: bold;

color: white;

font-size: 24px;

}

1. **Home.js:**

import React , { Component } from 'react';

import { Link } from 'react-router-dom';

import './Home.css';

import Modal from './Modal';

import { Children } from 'react';

import ReactDOM from 'react-dom';

class Home extends React.Component {

constructor(props) {

super(props);

this.state = {

showModal : false,

user: null,

error: null

}

}

componentDidMount() {

fetch('https://api.github.com/users/codeNinja')

.then(response => {

if (!response.ok) {

throw new Error('Error fetching user data from GitHub API');

}

return response.json();

})

.then(user => {

this.setState({ user });

})

.catch(error => {

this.setState({ error: error.message });

});

}

componentDidMount() {

fetch('https://api.github.com/users/mojombo')

.then(response => {

if (!response.ok) {

throw new Error('Error fetching user data from GitHub API');

}

return response.json();

})

.then(user => {

this.setState({ user });

})

.catch(error => {

this.setState({ error: error.message });

});

}

openModal = () => {

this.setState({showModal : true})

}

closeModal = () => {

this.setState({showModal : false})

}

render() {

const { user, error } = this.state;

if (error) {

return <p>Error: {error}</p>;

}

if (!user) {

return <p>Loading...</p>;

}

const { showModal } = this.state;

let modalContent = null;

if(showModal) {

modalContent = (

<Modal children={modalContent}>

<h2>GitHub User</h2>

<p>Username: {user.login}</p>

<p>Name: {user.name}</p>

<p>AvatarURL: {user.avatar\_url}</p>

<img src={user.avatar\_url} height={250} width={250} />

<p>Location: {user.location}</p>

<p>Public Repositories: {user.public\_repos}</p>

<p>Type: {user.type}</p>

<button onClick={this.closeModal}>Close Modal</button>

</Modal>

)

}

return (

<div>

<header>

<div className='logo'><img src="logo512.png" width={50} height={50} /> React Design</div>

<div className='login-menu'>

<div><Link className='login-item' to="/contact">Contact</Link></div>

<div><Link className='login-item' to="/about">About</Link></div>

<div><Link className='login-item' to="/login">Login</Link></div>

</div>

</header>

<main class="container">

<div class="box"><button onClick={this.openModal}>CodeNinja</button>

{modalContent}</div>

<div class="box"><button onClick={this.openModal}>Mojombo</button>

{modalContent}</div>

<div class="box"> Pjhyett</div>

<div class="box">Wycats</div>

<div class="box">Ezmobius</div>

<div class="box">lvey</div>

</main>

<footer className='footer'>

<div>

<h4>@copywrite M.Sc C.A. Batch 2023</h4>

</div>

</footer>

</div>

)

}

}

export default Home;

1. **index.css**

body {

margin: 0;

font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',

'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',

sans-serif;

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

}

code {

font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',

monospace;

}

.app {

text-align: center;

}

1. **index.js:**

import React from 'react';

import ReactDOM from 'react-dom';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

import { BrowserRouter } from 'react-router-dom';

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

root.render(

<BrowserRouter>

<App />

</BrowserRouter>

);

// If you want to start measuring performance in your app, pass a function

// to log results (for example: reportWebVitals(console.log))

// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals

reportWebVitals();

1. **Model.js:**

import React, { Children } from 'react'

import ReactDOM from 'react-dom'

import './Modal.css'

function Modal({children}) {

const modalRoot = document.getElementById('modal-root');

return ReactDOM.createPortal(

<div className='modal-overlay'>

<div className='modal'>

<div className='modal-content'>

{children}

</div>

</div>

</div>

, modalRoot

)

}

export default Modal

1. **Model.css**

.modal-overlay {

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

background-color: rgba(0, 0, 0, 0.5);

display: flex;

align-items: center;

justify-content: center;

}

.modal {

background-color: #12d2dd;

padding: 20px;

border-radius: 5px;

}

.modal-content {

margin-bottom: 10px;

}

button {

cursor: pointer;

}

1. **Create A new react app for react-reducer.**
2. **App.js:**

import React from 'react';

import {Provider} from 'react-redux';

import store from './Store';

import Counter from './Counter';

import Display from './Display';

function App() {

return (

<Provider store = {store}>

<div>

<center>

<h1>Redux Counter App</h1>

<Counter />

<Display />

</center>

</div>

</Provider>

);

}

export default App;

1. **Counter.js:**

import React from "react";

import { connect } from "react-redux";

import { increment , decrement } from "./Store";

import './App.css';

const Counter = ({increment,decrement}) => {

return (

<div>

<h2>Counter</h2>

<button class="button-container" onClick = { increment }>Increment</button>

<button class="button-container" onClick = { decrement} >Decrement</button>

</div>

);

};

const mapStateToProps = (state) => {

return {

count : state.count,

};

};

const mapDispatchToProps = {

increment,

decrement,

};

export default

connect(mapStateToProps,mapDispatchToProps)(Counter);

1. **Store.js:**

import {createStore} from 'redux';

const initialState = {

count : 0,

}

const INCREMENT = 'INCREMENT';

const DECREMENT = 'DECREMENT';

export const increment = ( ) => (

{ type : INCREMENT }

);

export const decrement = ( ) => (

{ type : DECREMENT}

);

const CounterReducer = (state = initialState , action) => {

switch (action.type) {

case INCREMENT :

return {...state ,

count : state.count + 1,

};

case DECREMENT :

return {...state,

count : state.count - 1,

};

default :

return state;

}

}

const store = createStore (

CounterReducer

);

export default store;

1. **Display.js:**

import React from "react";

import { connect } from "react-redux";

const Display = ( { count }) => {

return (

<div>

<h2>Display</h2>

<p>Count : {count}</p>

</div>

);

}

const mapStateToProps = (state) => {

return {

count : state.count,

}

}

export default connect(mapStateToProps)(Display);