## Page:1

# React-Redux COUNTER

REDUX Folder

1. action.js

2. actiontype.js

3. reducer.js

4. store.js

# store.js

import { legacy\_createStore } from "redux";

import { reducer } from "./reducer";

const initialstate = {

counter: 0,

};

export const store = legacy\_createStore(reducer, initialstate);

# reducer.js

import { ADD, SUB } from "./actiontype";

export const reducer = (state, { type, payload }) => {

switch (type) {

case ADD:

return { counter: state.counter + payload };

case SUB:

return { counter: state.counter - payload };

default:

return state;

}};

## Page:2

# action.js

import { ADD, SUB } from "./actiontype";

export function AddToCounter(val) {

return { type: ADD, payload: val };

}

export function SubToCounter(val) {

return { type: SUB, payload: val };

}

# actiontype.js

export const ADD = "ADD";

export const SUB = "SUB";

# main.jsx

import ReactDOM from "react-dom/client";

import App from "./App.jsx";

import "./index.css";

import { Provider } from "react-redux";

import { store } from "./Redux/store.js";

ReactDOM.createRoot(document.getElementById("root")).render(

<Provider store={store}>

<App />

</Provider>

## Page:3

);

# App.jsx

import { useDispatch, useSelector } from "react-redux";

import "./App.css";

import { AddToCounter, SubToCounter } from "./Redux/action";

import { store } from "./Redux/store";

function App() {

const reduxstore = store;

console.log(reduxstore);

// const { dispatch, getState, subscribe } = store;

// console.log(getState().counter);

const dispatch = useDispatch(); //in place of dispatch

const counter = useSelector((s) => s.counter); // in place of getState ,as a parameter anythink you can pass

console.log(counter);

return (

<>

{/\* <h2>{getState().counter}</h2> \*/}

<h2>{counter}</h2>

<button

onClick={() => {

dispatch(AddToCounter(1)); //type: ADD, payload: 1

// console.log(getState().counter);

}}

## Page:4

>

+

</button>

<button

onClick={() => {

dispatch(SubToCounter(1)); //type: SUB, payload: 1

// console.log(getState().counter);

}}

>

-

</button>

</>

);

}

export default App;