# COUNTER

REDUX Folder

1.reducer.js

2.store.jsx

Main.jsx

import ReactDOM from "react-dom/client";

import App from "./App.jsx";

import "./index.css";

ReactDOM.createRoot(document.getElementById("root")).render(<App />);

App.jsx

import { useState } from "react";

import "./App.css";

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

function App() {

const reduxstore = store;

console.log(reduxstore);

const [value, setValue] = useState(1);

const { getState, dispatch, subscribe } = store;

console.log(getState().counter);

const handleAdd = () => {

dispatch({ type: "ADD", payload: 1 });

console.log(getState().counter);

};

const handleSub = () => {

dispatch({ type: "SUB", payload: 1 });

console.log(getState().counter);

};

subscribe(() => {

console.log("hii");

console.log(value);

setValue((prev) => prev + 1);

});

return (

<>

<h1>{getState().counter}</h1>

<button onClick={handleAdd}>+</button>

<button onClick={handleSub}>-</button>

</>

);

}

export default App;

REDUX FOLDER

Reducer.jsx

export const reducer = (state, action) => {

console.log(action);

switch (action.type) {

case "ADD":

return { counter: state.counter + action.payload };

case "SUB":

return { counter: state.counter - action.payload };

default:

return state;

}

};

Store.js

import { legacy\_createStore } from "redux";

import { reducer } from "./reducer";

const initialstate = {

counter: 0,

};

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