npm install redux react-redux redux-think

Redux

1. Redux создаёт и запускает экшн “redux init”
2. Наш initial state записывается в хранилище store в редакс.
3. С помощью метода mapStateToProps мы получаем доступ к state, и берём необходимые данные оттуда и перкидываем в компонент через props.
4. Когда мы вызываем action с помощью mapDispatchToProps, этот метод передаёт тип и данные в reducer.
5. Reducer отбирает данные согласно типу, и делает копию state с новыми полученными данными.
6. Новые данные приходят в компонент и компонент перерисовывается.

Create reducer

const initialState = {

  likes: 0,

};

export const likesReducer = (state = initialState, action) => {

  switch (action.type) {

    case INCREMENT:

      return { ...state, likes: state.likes + 1 };

    case DECREMENT:

      if (state.likes === 0) return state;

      return { ...state, likes: state.likes - 1 };

    default:

      return state;

  }

};

Create Root reducer

import { combineReducers } from "redux";

import { likesReducer } from "./likesReducer";

import { messagesReducer } from "./messagesReducer";

export const rootReducer = combineReducers({

  likes: likesReducer,

  messages: messagesReducer,

});

Create store

import { Provider } from "react-redux";

const store = createStore(rootReducer);

<Provider> // to be able to access store from any Component

  <App store={store} />

</Provider>

Create store with redux (for async operations)

import { Provider } from "react-redux";

import { createStore, compose, applyMiddleware } from "redux";

import reduxThunk from "redux-thunk";

const composeEnhancers = window.\_\_REDUX\_DEVTOOLS\_EXTENSION\_COMPOSE\_\_ || compose;

const store = createStore(reducers, /\* preloadedState, \*/ composeEnhancers(applyMiddleware(reduxThunk)));

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

root.render(

  <React.StrictMode>

    <Provider store={store}>

      <App />

    </Provider>

  </React.StrictMode>

);

To connect state with a Class component

class Likes extends React.Component {

  constructor(props) {

    super(props);

  }

  render() {

    return (

      <div></div>

    );

  }

}

function mapStateToProps(state) {

  return { likes: state.likesReducer.likes };

}

export default connect(mapStateToProps, { actionCreator1, actionCreator2 })(Likes);

To Connect state with a Functional component

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

function Comments(props) {

  const dispatch = useDispatch();

  const comments = useSelector((state) => state);

  return (

    <div className="card-comments"></div>

  );

}