1. Metode untuk pengiriman value ke store besar dan memanggil component yang di perlukan disebut dengan?

Jawab: Redux, adalah sebuah aplikasi state management. State management adalah cara untuk memfasilitasi komunikasi dan berbagai data lintas komponen.

1. Sebutkan fungsi reducer dan store pada redux?

Jawab:

Reducer adalah fungsi untuk mengambil status pada aplikasi. Reducer berfungsi untuk meng-action dan mengambil status baru. Status ini disimpan menjadi objek, dan menentukan status aplikasi berubah sebagai respons terhadap tindakan yang dikirim oleh store.

Store adalah fungsi untuk menyimpan status aplikasi. Disarankan untuk menyimpan satu store di aplikasi redux. Kita dapat mengakses status yang disimpan dan mengupdate.

1. Buatlah Langkah Langkah untuk membuat redux?

Jawab:

1. Membuat aplikasi React

Kita bisa membuat aplikasi react menggunakan command berikut:

***“npx create-react-app my-react-redux-app”***

Sebelumnya, pastikan sudah menginstall versi terbaru dari **node** terlebih dahulu. Kemudian masuk ke directory yang sudah dibuat dan jalankan command berikut:

***“npm start”***

Maka akan muncul tampilan awal dari React di browser.

1. Install dependency redux dan react-redux

Install redux ke aplikasi React, menggunakan command berikut:

***“npm install redux react-redux”***

Package react-redux akan langsung terhubung ke aplikasi React yang telah kita buat dengan Redux.

1. Setup Redux Store

Kita bisa melakukan setup redux store pada berkas yang akan menjadi parent dari semua component yang ada, bisa melalui App.js ataupun index.js.

Pada code, kita dapat mengimpor “Provider” dari react-redux. Provider merupakan component yang disediakan oleh library react-redux.

Component Provider memiliki props berupa store yang akan membungkus semua component yang ada, sehingga semua component dapat mengakses informasi yang ada pada redux store.

Kita dapat menggunakan function “createStore()” yang disediakan oleh redux untuk membuat store. Kita dapat membuat file reducer dan kemudian mengimport rootReducer di reducer kita.

Kita bisa mengimport “combineReducers()” untuk mengkombinasikan beberapa reducers yang berbeda manjadi satu reducer.

Pada redux kita dapat membuat banyak component reducer. Karena fungsi “createStore()” di store.js hanya dapat menggunakan satu reducer, maka “combineReducer()” dapat mengonversi beberapa reducernya menjadi satu.

Dengan ini, kita berhasil membuat rootReducer yang akan digunakan.

1. Setup reducer, actions, types

Pada langkah ini, kita akan membuat component reducer, actions, dan types.

Types adalah konstanta yang berisi nilai string yang akan mengidentifikasi jenis actions yang ada pada aplikasi. Kita dapat membuat types yang berbeda pada setiap actions yang akan kita gunakan.

Sederhananya sebuah action merupakan sebuah object yang memiliki property type.

Reducer adalah bagian redux yang merubah state menjadi respon yang terjadi ketika Action di dispatch(). Sederhananya, untuk merubah state hanya bisa dilakukan di Reducer, dan Reducer hanya merubah state jika ada action yang di dispatch(). Yang terjadi adalah ketika action yang di dispatch() type nya adalah ADD\_DATA maka akan mengembalikan sebuah object baru yang isinya adalah state lama di tambah dengan data baru yang di passing melalui Action. Maka hasil itu akan menjadi state yang baru.

1. Menghubungkan redux dengan JSX

Kita dapat mengimpor function connect dari library react-redux, kemudian mengimpor actions login dari actions. Function connect merupakan sebuah Higher Order Component (HOC) yang berfungsi untuk memanipulasi component yang diterima dengan menambahkan props ke component tersebut dan kemudian mereturnnya.

1. Buatlah contoh program menggunakan redux?
2. Buatlah web UI data table CRUD dengan hit rest API dengan method GET, POST, DELET. Dengan tampilan seperti dibawah ini untuk request API menggunakan

GET <https://jsonplaceholder.typicode.com/users>,

POST <https://jsonplaceholder.typicode.com/users>

DELET [https://jsonplaceholder.typicode.com/posts/${id}](https://jsonplaceholder.typicode.com/posts/$%7bid%7d)

