|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| React JS vid 55 s/d 58 | 90 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Redirect adalah cara untuk mengarahkan seseorang yang membuka halaman web yang kemudian akan diarahkan ke halaman web yang diperbolehkan untuk diakses dengan memasukkan data yang diperlukan untuk melindungi data sehingga tidak dapat semua orang dapat mengakses data di dalam web. Jadi, jika seseorang tersebut tidak terdaftar dalam database yang kita miliki seseorang tersebut tidak dapat mengakses halaman yang kita ingin lindungi.

2. useHistory berfungsi untuk membantu mengakses objek histori, yang digunakan untuk menavigasi secara terprogram ke rute lain menggunakan method push atau replace. Cara menggunakannya adalah dengan mengimport useHistory dari react-router-dom/import { useHistory } from "react-router-dom" ke dalam komponen yang ingin digunakan.

3. sessionStorage adalah jenis penyimpanan web yang memungkinkan penyimpanan data secara lokal dalam browser pengguna dan hilang ketika sesi halaman berakhir. Data ini akan dihapus saat browser ditutup dan tidak akan tersedia saat browser dibuka kembali.

4. Agar halaman web teredirect adalah dengan membuat konstanta untuk useHistory(const history = useHistory()). Kemudian, setItem untuk token menggunakan sessionStorage(sessionStorage.setItem("token", token)). Kemudian, menggunakan if untuk mengarahkan token yang dimasukkan akan terarah ke halaman admin dengan menggunakan method push dari useHistory(if (gettoken() != "undefined") { history.push("/admin") }). Kemudian, membuat konstanta lagi untuk mendapatkan token dengan menggunakan arrow function dengan mengunakan getItem dari sessionStorage(const gettoken = () => sessionStorage.getItem("token")). Kemudian, pada komponen back impor untuk redirect dari react-router-dom(import { Redirect } from "react-router-dom"). Lalu, menggunakan if untuk mengarahkan tokennya jika data yang dimasukkan tidak terdaftar maka user akan diarahkan ke halaman login terus (if (sessionStorage.getItem("token") === "undefined") {

return <Redirect to="/login" /> }). Jadi, caranya adalah dengan mengambil data user yang terdaftar dengan menggunakan tokennya, tokennya lah yang nantinya akan digunakan untuk mengambil data user yang ada didalamnya, apakah sudah sesuai apa belum jika tidak sesuai maka akan diarahkan ke dalam halaman login terus menerus jika data yang dimasukkan user sudah sesuai maka akan diarahkan ke halaman admin.

5. Penggunaan tanda '||' dalah untuk function or.

6. Untuk meredirect halaman ketika di logout akan mengarah ke halaman login lagi adalah dengan membuat nav terlebih dahulu. Lalu, pada bagian buttonnya tambahkan dengan function onClick lalu akan menjalankan function hapus(onClick={hapus}). Lalu, pada function hapus menggunakan function clear dari sessionStorage untuk menghapus data session(sessionStorage.clear()) lalu mengarahkannya kedalam halaman login dengan menggunakan method push dari useHistory(history.push("/login")). Kemudian agar halamannya tetap ke redirect ke login ketika sudah di logout meskipun sudah dimasukkan url untuk menuju halaman admin adalah dengan menggunakan if untuk mendaftarkan tokennya ketika undifined atau null akan tetap diarahkan ke halaman login(if ( sessionStorage.getItem("token") === "undefined" || sessionStorage.getItem("token") === null ) { return <Redirect to="/login" /> }).

7. Untuk mengambil data dari sessionStorage dengan cara menggunakan setItem dari function sessionStorage lalu menamai data yang diambil kemudian ambil data asalnya (sessionStorage.setItem("email", res.data.data.email)). Untuk menampilkan data yang diambil dari sessionStorage adalah dengan menggunakan method getItem kemudian memanggil nama yang sudah diambil sebelumnya ({sessionStorage.getItem("email")}).

8. Hash password adalah menampilkan password yang didaftarkan dalam database dengan menggunakan simbol-simbol acak sehingga orang lain tidak mengetahui isi password aslinya. Cara menggunakannya didalam lumen menambahkan use Illuminate\Support\Facades\Hash di dalam controller loginnya. Kemudian, pada bagian yang digunakan untuk mengimput password ditambahkan dengan Hash::make agar password yang nantinya akan diinputkan oleh user akan terhash secara otomatis.

9. Untuk mengecek password yang dihash adalah dengan menggunakan perintah Hash::Check.

10. window.location.reload() adalah untuk merefresh halaman secara otomatis sesudah memasukkan data/menambahkan data.

11. Untuk memberi hak kses user bisa dengan memberikan tambahakan pada bagian sidebarnya yang sudah disediakan untuk menuju halaman-halaman Kategori, Menu, Pelanggan, Order, Order Detail, dan User Admin dengan misal, {sessionStorage.getItem("level") === "admin(user yang ditunjuk untuk mengakses halaman tersebut)" ? ( <li className="list-group-item">User Admin</li>(halaman yang bisa diakses user admin) ) : ( "" )}. Jadi, sessionStorage akan mengambil data tabel level dalam database jika level user tersebut admin akan ditampilkan untuk bisa mengakses halaman User Admin.

**Saya Belum Mengerti**

1.

2.

3.