**FORM PENILAIAN MANDIRI**

**PEMRROGRAMAN WEB DAN MOBILE**

|  |  |
| --- | --- |
| **Absen** | **Nama** |
| 20 | Naqsyadenta Afkarin |

|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Project React Resto - 6 | 85 |

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

1. Bisa menampilkan datanya saja (data dari user yang akan di loginkan) pada saat login di dalam web atau browsernya di dalam console bagian inspectnya. Dengan cara menuliskan datanya dua kali di dalam function login bagian console lognya (console.log(res.data.data);).

2. Bisa menampilkan tokennya saja (yang berasal dari macam-macam data di dalam user yang akan di loginkan) pada saat login di dalam web atau browsernya di dalam console bagian inspectnya. Dengan cara menuliskan data dan setelahnya dituliskan token di dalam function login bagian console lognya (console.log(res.data.token);).

3. Await, menunggu perintah dari sesuatu yang sudah dibuat sebelumnya (jadi akan jalan setelah dijalankan terlebih dahulu).

4. SessionStorage, untuk menambahkan key, namun key tersebut akan hilang setelah browsernya ditutup (direstart). Bisa juga dihapus menggunakan remove dan clear.

5. GetItem, untuk melihat value dari key yang telah dibuat sebelumnya di console (inspect).

6. SetItem, untuk menambahkan local storage dan session storage.

7. If, digunakan untuk membuat dua kondisi, bila kondisi yang pertama sudah benar, maka akan langsung dijalankan isi dari kondisi pertamanya. Namun bila salah, maka akan menjalankan isi dari kondisi keduanya.

8. Else, merupakan kondisi kedua dari if.

9. Sama dengan tiga kali (===), untuk menjadikan suatu data menjadi sama persis dengan data yang disamakan (lebih akurat daripada sama dengan dua kali).

10. Console.log (clg), untuk memunculkan hasil dari klik saat mencoba dan akan muncul di console bagian inspect.

11. Tidak sama dengan (!=), untuk tidak menyamakan satu hal dengan hal yang lainnya.

12. Bisa menjadikan login bila salah maka akan tetap di halaman yang sama (halaman login, yang berarti tetap dan tidak berubah) dan bila benar, maka akan berpindah halaman dan masuk ke dalam halaman adminnya (yang berisikan kategori, menu, pelanggan, order, order detail, dan user admin).

13. Push, untuk mendorong suatu perintah ke dalam halaman yang sudah ditujukan sebelumnya. Seperti history.push(‘/admin’); untuk menuju ke dalam halaman admin bila pada saat loginnya sudah berhasil.

14. Redirect, untuk mengembalikan suatu hal dari sesuatu yang telah ditentukan sebelumnya. Sperti bila token pada login tidak ada isinya (kosong/undefined) maka hasilnya diarahkan ke dalam halaman login (tetap).

15. Bisa membuat logout untuk user yang telah login sebelumnya.

16. Menggunakan bootstrap untuk tampilan nav (dashboard) yang bila sudah login akan tertera email user yang login tersebut beserta levelnya (misalkan emailnya admin@gmail.com yang berlevelkan admin).

17. ClassName, untuk membuat css. Menuliskannya ini sebelum memberikan css dari suatu tag div polos (div yang sendiri).

18. OnClick, akan menjalankan/memunculkan sesuatu dari yang sudah dibuat sebelumnya (dengan code) dan akan berjalan pada saat di klik (bila tidak di klik terlebih dahulu, maka tidak akan terjalankan maupun muncul).

19. Menggunakan button dengan bootstrap btn btn-outline-success untuk menjadikan buttonnya lebih oval dari pada button pada awalnya, serta berwarna hijau dan hanya berwarna hijau pada bagian pimggir dan tulisannya bila belum di klik (karena outlinenya).

20. Alert, digunakan untuk memunculkan kotakan form kecil (localhost:3000 says) untuk menunjukkan sesuatu hal, misalkan seperti memunculkan tambah pada saat menekan button tambah user di bagian admin.

21. Menggunakan sessionStorage untuk membuat setiap user yang login akan memasukkan tiap tokennya masing-masing.

22. Async, untuk menjalankan program dengan bertahap dan tidak perlu menunggu proses yang lain terlebih dahulu (tidak secara langsung).

23. Bisa menampilkan emailnya saja (yang berasal dari macam-macam data di dalam user yang akan di loginkan) pada saat login di dalam web atau browsernya di dalam console bagian inspectnya. Dengan cara menuliskan data dan setelahnya dituliskan email di dalam function login bagian console lognya (console.log(res.data.email);).

24. Bisa menampilkan levelnya saja (yang berasal dari macam-macam data di dalam user yang akan di loginkan) pada saat login di dalam web atau browsernya di dalam console bagian inspectnya. Dengan cara menuliskan data dan setelahnya dituliskan level di dalam function login bagian console lognya (console.log(res.data.level);).

25. Menampilkan data dari yang sudah login sebelumnya dengan <li className=”nav-item list-unstyled”(nama bagian yang akan ditampilkan seperti email salah satunya) Email : {sessionStorage.getItem(‘email’)}></li>.

26. List-unstyled, untuk menghilangkan titik dari li yang sudah ada sebelumnya (bawaannya).

27. Bila mengubah urlnya ke /admin maka tidak akan bisa langsung masuk ke dalam tampilan adminnya dan akan tetap dikembalikan ke dalam tampilan login formnya (/login).

28. Hash, untuk menyembunyikan password dari suatu user dengan menggantinya dengan campuran huruf (kecil dan kapital) dan angka untuk menyembunyikan password yang sebenarnya (jadi password yang sebenarnya tidak akan terlihat di databasenya (apirestoran) dengan menggunakan hash tersebut, namun passwordnya tetap sama dan akan bisa diloginkan sesuai dengan yang dibuat pada awalnya).

29. Bisa menghash password sesuai dengan yang ada di video. Dengan cara meletakkan hashnya pada function register bagian $data di passwordnya (‘password’ => Hash::make($request->input(‘password’))).

30. Untuk langsung merefresh suatu tampilan web atau browser menggunakan window.location.reload();. Yang berfungsi agar setelah login suatu user, maka akan langsung tampil juga isi di dalamnya (tidak perlu merefresh sendiri (otomatis)).

31. Bisa bial mengeklik bagian tulisan dashboard maka akan langsung tertuju pada bagian awal dari web atau browsernya (seperti saat pertama kali login).

32. Bisa membuat masing-masing dari status levelnya akan ada bagian apa saja (jadi tidak semua level bisa memegang kendali dari segalanya, hanya admin saja yang dapat memegang kendali dari segalanya (sesuai dengan yang di video, pada bagian Side.js di react-js)). Seperti admin yang akan dapat memegang bagian kendali dari semuanya (kategori, menu, pelanggan, order, order detail, dan user admin), kasir hanya akan memegang kendali pada bagian order dan order detail, serta koki yang hanya dapat memegang kendali pada bagian order detail saja. Meskipun nama emailnya beragam macam, namun yang dapat membedakan dari setiap kendalinya ialah hanya levelnya (admin, kasir, dan koki). Bila ada yang lebih dari satu bagian kendali, maka akan diberikan or (||).

33. Format Document, untuk merapikan coding yang telah dibuat sebelumnya. Dengan cara klik kanan, kemudian pilih Format Document (pastikan prettiernya sudah berjalan terlebih dahulu).

34. Pada saat memberikan await di dalam suatu function, maka function tersebut harus diberikan async agar dapat berjalan.

35. Menyalakan xampp (bagian Apache dan MySQL) untuk menyalakan api lumennya (localhost:8000).

36. Menyalakan yarn di dalam terminal visual studio code dengan mengetikkan yarn start. Bila tidak bisa berjalan, bisa juga menyalakannya lewat command prompt (cmd) dengan mengetikkan hal yang sama (yarn start). Namun sebelumnya harus diletakkan di dalam folder project react resto dibagian react-resto nya pada saat di dalam cmd nya (menyambungkan foldernya dengan menuliskan cd di depan url folder yang akan disambungkan nantinya). Setelah menyala akan menuju localhost:3000.

**Saya Belum Mengerti**

1.

2.

3.