|  |  |
| --- | --- |
| Materi | Nilai |
| REACT-JS-V31-40 | **98** |

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

1. Saya sudah mengerti bahwa react telah membuat semacam fungsi khusus bernama *hook*, yaitu merupakan fungsi yang memungkinkan kita untuk “mengaitkan” state dan fitur-fitur lifecycle React dari function component. Contoh hook itu sendiri ialah **usestate()** yang biasa kita gunakan.
2. Saya sudah mengerti bahwa dalam pembuatan useState itu dibutuhkan dua variable penting yaitu variable untuk menampung nilai state saat ini dan variabel untuk memperbarui nilai tersebut atau untuk meng-set var tersebut, contohnya yaitu

const [isi,setIsi ]= useState([]);

1. Saya sudah mengerti bahwa kita bisa memanfaatkan *hook* useEffect() untuk mengambil atau mengfetch data dari database, caranya kita harus membuat function terlebih dahulu kemudian mengisinya dengan axios, contoh penggunaanya yaitu

useEffect(() => {

        async function fetchData() {

            const request = await link.get('/kategori'); }}

1. Saya sudah mengerti bahwa untuk menggunakan react-hook-form kita harus mengimportnya terlebih dahulu, caranya yaitu kita harus mengetikan kode berikut

import { useForm } from "react-hook-form";

1. Saya sudah mengerti bahwa kita bisa mengcustom hook untuk mempermudah pekerjaan kita, caranya kita harus membuat file javascript yang berisi usestate dan useeffect kemudian kita atur programnya, misal untuk get data atau delete data, dan untuk menggunakannya kita bisa mengimport file tersebut lalu memasukkan url di file utama sehingga file utama tak terlalu penuh. Contohnya

const [isi] = useGet('/kategori');

1. Saya sudah mengerti bahwa untuk menampilkan data dari database ke dalam table di project react, kita harus menggunakan map namun sebelumnya kita harus menggunakan usestate ataupun custom hook untuk get data, contoh penggunaan map yaitu,

{isi.map((val, index) => { <tr>{val.menu}</tr> }

1. Saya bisa membuat penomoran otomatis di dalam table, caranya kita bisa membuat variable di luar return kemudian menetapkan nilainya menjadi bernilai 1, jika sudah kita bisa memanggil variable tersebut beserta operator increment(**++**) contohnya

<td>{no++}</td>

1. Saya sudah bisa membuat cleanup, contohnya

useEffect(() => {

//program effect

fetchData();

        return () => { ambil = false;  };

    }, [isi]);

Jadi, menurut saya cleanup ini berperan ketika program effect itu tidak berjalan, jika sebelumnya variable ambil itu bernilai true, maka saat tidak digunakan ambil akan bernilai false.

1. Saya sudah mengerti bahwa kita bisa memanfaatkan class table dari bootstrap untuk membuat table menjadi lebih rapi, contohnya

<table className="table mt-4">…</table>

1. Saya sudah mengerti bahwa kita bisa memanfaatkan trigger dari useEffect supaya kita tidak perlu memanggil function berkali-kali, jadi hanya cukup sekali saja sehingga proses load akan menjadi lebih cepat, contoh kodenya yaitu

useEffect(() => {

    fetchData();

  }, [isi]);

1. Saya sudah mengerti bahwa jika kita ingin menggunakan variable yang ada di dalam file custom hook, kita harus mereturn variable tersebut terlebih dahulu supaya bisa dipakai, caranya dengan menggunakan kode berikut

return {hapus, pesan, setPesan};

1. Saya sudah bisa membuat custom hook pada onclick, caranya seperti membuat custom hook sebelumnya namun bedanya hanya terdapat pada peletakkannya yang ada di dalam onclick, contohnya

<button onClick={() => hapus(val.idpelanggan}>Delete</button>

1. Saya sudah mengerti bahwa saat kita ingin menampilkan data berupa gambar kita harus menggunakan tag <img>, jika tidak amaka akan keluar direktori dari gambar tersebut, contoh penggunaanya yaitu

<img src={val.gambar} height="100" width="150" />

1. Saya sudah mengerti bahwa jika kita ingin menginputkan data ke dalam form dan data tersebut berupa file maka kita harus menggunakan append untuk menginsertnya. contohnya

formData.append("idkategori", data.idkategori);

1. Saya sudah bisa membuat object baru, caranya kita harus menentukan variable atau konstantanya dulu kemudian kita bisa mengisi object lamanya, contohnya

const formData = new FormData();

1. Saya sudah mengerti bahwa ketika kita ingin mengudate data yang bertipe file, kita harus menggunakan perintah post diikuti dengan idnya, bukan menggunakan put, contoh perintahnya yaitu

link.post("/menu/" + idmenu, formData).then((res) => setPesan(res.data.pesan));

1. Saya bisa menampilkan pesan ke dalam alert dari bootstrap sehingga enjadi lebih enak dipandang, dengan menggunakan kode

<p className="alert alert-dark" role="alert">{pesan} </p>

1. Saat mengupdate data dengan form input, namun ada salah satu data yang ingin kita biarkan kosong kita harus menetapkan nilai register menjadi null terlebih dahulu supaya form tersebut dapat disubmit, contohnya yaitu

{...register("gambar", { required: null })}

1. Saya sudah bisa memanggil komponen saat menggunakan navbar. Jadi ketika kita mengetik pilihan di navbar akan membawa kita menuju komponen file yang sudah diimport. Caranya kita bisa menggabungkannya dengan percabangan if, contoh

if (isi===order) {

        tampil = <Order />

    }

1. Saya bisa mengimplementasikan ifelse untuk menampilkan apakah pelanggan sudah membayar pesanannya atau belum. Yaitu dengan mengetikkan kode perintah berikut

val.status===0 ? <button className='btn btn-danger'>Belum Bayar</button> : <p>Lunas</p>

1. l

**Saya kurang paham**

* + - 1. -