|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Tugas coding React JS video 21-28 | 90 |

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

1. Penggunaan tag <Route path="/" component={Front} exact /> untuk membuat sebuah statement route atau membentuk sebuah url yang mengarah ke folder yang telah di tentukan
2. Penggunaan tag import {BrowserRouter as Router, Route} from 'react-router-dom'; untuk mengimport penggunaan statement router di dalam React JS
3. Penggunaan tag import Footer from './Footer'; untuk mengimpor isi dari sebuah file di dalam React JS
4. Penggunaan tag const { isi } = useParams(); untuk menggunakan sebuah parameter dalam hal penggunaan atau pemanggilan sebuah route di dalam React JS
5. Penggunaan tag <Route path={`${path}/:isi`}> untuk memanggil sebuah route path dengan mengambil nilai dari sebuah variable yang telah di buat sebelumnya
6. Menampilkan sebuah data dari sbeuah database yang telah di buat sebelumnya dengan menggunakan statement Axios di dalam React JS dengan menambahkan sebuah tag sebuah statement url dan import axios
7. Penggunaan tag import { link } from "../Axios/link"; untuk mengambil sebuah url atau sebuah link yang menuju ke dalam database yang telah di buat sebelumnya
8. Penggunaan tag const [isi, setisi] = useState([]); untuk memanggil sebuah komponen dan sebuah parameter function yang di dalamnya terdapat sebuah statement yang memanggil beberapa data yang ada di dalam database untuk di tampilkan ke dalam bentuk web dengan menggunakan sebuah variable
9. penggunaan tag if (isi === "kategori") { tampil = <Kategori></Kategori>; } untuk menggunakan sebuah kondisi apabila sebuah variable di panggil atau terisi maka akan menjalankan statement di dalamnya
10. penggunaan statement form react hook di dalam React JS dengan menambahkan sebuah tag import { useForm } from "react-hook-form"; untuk mengimport statement hook form yang telah di sediakan React JS
11. melakukan sebuah statement input di dalam React JS dengan menambahkan sebuah tag ref={register} di dalam statement input
12. penggunaan tag <form onSubmit={handleSubmit(simpan)}> untuk apabila di dalam statement tersebut berjalan dengan sebuah if yang di jalankan maka akan menjalankan sebuah parameter
13. penggunaan tag ref={register({ required: true })} untuk apabila dalam sebuah peng inputan data harus bernilai true atau terisi
14. penggunaan tag {errors.kategori && "Kategori harus di isi!"} untuk membuat sebuah notifikasi error saat sebuah nama peng inputan tidak sesuai dengan statement input yang berjalan
15. penggunaan tag reset(); apabila sebuah statement selesai maka akan secara otomatis reset atau kembali ke keadaan kosong atau awal
16. penggunaan tag const { register, handleSubmit, reset, errors } = useForm(); untuk menggunakan beberapa statement di dalamnya dalam bentuk function useForm
17. melakukan sebuah post data di dalam React JS dan menambahkan sebuah statement pesan dengan mengubah pada bagian file lumen dengan menambahkan sebuah kondisi maka akan menjalankan statement di dalamnya
18. penggunaan tag link.post("/kategori", data).then((res) => setPesan(res.data.pesan)) apabila sebuah statement tersebut berhasil maka akan menjalankan statement di dalamnya
19. penggunaan statement hapus data di dalam React JS dengan menggunakan function async function hapus(id) { dengan tag statement di dalamnya const res = await link.delete("/kategori/" + id); untuk menghapus pada bagian id data tertentu di dalam database
20. penggunaan tag if (window.confirm("yakin akan menghapus?")) { untuk apabila ingin menjalankan sebuah statement, sebelumnya akan muncul sebuah pop up notifikasi konfirmasi
21. penggunaan statement showData di dalam React JS
22. penggunaan tag setValue("kategori", res.data[0].kategori); untuk menampilkan sesuai dengan variable dan nilai yang di tentukan
23. penggunaan tag const [idkategori, setIdkategori] = useState(""); untuk menjalankan sebuah statement di dalamnya dengan menggunakan useState yang telah di sediakan oleh React JS
24. penggunaan tag setPilihan(true); untuk mengembalikan kondisi ke keadaan true atau semula