|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| React JS | 91 |

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

1. React js, digunakan untuk membuat sebuah web agar hasil refreshnnya (loadnya) tidak terlalu lama. Berbentuk putus-putusan dari sebuah tampilan yang disatukan (maka dari itu isi dari komponennya bila ada perubahan tidak akan lama dalam mengeluarkan hasil refreshan barunya).

2. Bisa mendownload node.js sesuai dengan versi yang dapat digunakan pada laptop saya (laptop saya tidak dapat menggunakan versi terbaru dari node.js, saya mendownload versi 13, namun isinya sama seperti yang ada di video yang terdapat node modules, app, src, dan yang lainnya di dalamnya).

3. Bisa mengatur node.js agar dapat digunakan (di cmd/command prompt) sesuai seperti dengan yang ada di video (sampai keluar happy hacking !). Mengetikkan npx create-react-app dan nama foldernya diberi nama react-js.

4. Meletakkan node.js di dalam C: xampp pada folder htdocs.

5. Mengatur isi src dan public sesuai dengan yang ada di video dan bila ada yang tidak perlu digunakan bisa dihapus.

6. Membuka terminal > new terminal > kemudian mengetikkan npm start untuk menyalakan localhost dari react js nya (bila sudah bisa menyala akan memunculkan dari logo node.js yang sudah tersedia sebelumnya (sudah dari sananya)).

7. Node js, server dari react js yang digunakan untuk menampilkan hasil dari codingan yang telah dibuat sebelumnya (hampir sama seperti xampp fungsinya). Url nya bertuliskan localhost:3000.

8. Bisa menampilkan tulisan dengan h1 (untuk menebalkan tulisan sesuai dengan angka dibelakangnya, terdiri dari 1-6) dan p (untuk isi dari paragraph bisa dibungkus dan diletakkan di dalam p ini) di dalam file App.js pada functionnya dan di dalam divnya (div className=”App”).

9. Membuat folder pages untuk menaruh perbagian dari yang akan dibuat nantinya (seperti kontak, sejarah, tentang, home, nav, dan lain-lain).

10. Membuat file (terutama pada file komponen) dan menamainya harus diawali dengan huruf kapital di depannya (seperti Kontak.js dan yang lainnya pada video).

11. Menuliskan <… (titik-titik tersebut berisikan nama file yang akan ditampilkan) /> pada React.StrictMode (seperti <Kontak />).

12. Menuliskan import … (titik-titik tersebut berisikan nama file yang akan ditampilkan) from ‘./ …’; (titik-titik berisikan nama folder dan file yang akan ditampilkan (bila filenya ada di dalam folder), harus sama seperti nama file pada titik-titik sebelum ini). (Seperti import Kontak from ‘./pages/Kontak’; dan bila filenya tidak berada dalam folder bisa dilangsung seperti import App from ‘./App’;).

13. Bisa membuat komponen sesuai dengan yang ada di video.

14. Komponen, merupakan suatu bagian yang terdiri dari masing-masing sesuai fungsinya, yang isinya bisa beragam dan bisa dikaitkan antara komponen satu dengan komponen yang lainnya.

15. Mengatur emmet jsx di dalam vs code agar tagnya dapat digunakan seperti biasanya (file>preferences>setting>emmet>edit setting.json). Dengan memberikan emmet include languanges ke dalam setting.jsonnya.

16. Pada file yang terdapat function di dalamnya, pada bagian nama function dan export default bawah sendiri harus diberikan nama sesuai dengan nama filenya pada saat itu (seperti file yang bernama Nav, maka nama functionnya juga harus Nav beserta export defaultnya (export default Nav) agar dapat tersambung dan digunakan).

17. Selalu menyambungkan file yang akan digunakan pada index.js (bagian ReactDom.Render (<… />) dan import (import …. from ‘./….’)).

18. Bisa menambahkan gambar ke dalam react-js menggunakan img src =”” (gambarnya pada video bertipe png). Dan bila ada kesalahan dalam gambarnya yang akan muncul adalah tulisan di dalam alt. Pada bagian sejarah yang terdapat gambar seperti yang ada di video.

19. Memindahkan komponen lainnya seperti kontak, tentang, dan lainnya pada file lain yang berbeda dengan index.js, karena disana dipisahkan agar tidak tercampur dengan main komponen / komponen utamanya (hasil tampilannya nanti juga akan sama seperti sebelumnya).

20. Bisa menginstal react-router-dom dan memakainya sesuai dengan yang ada di video ( menginstallnya dengan mengetikkan npm install react-router-dom pada terminal, tempat menuliskannya sama seperti pada saat pertama kali memulai localhost dari react-jsnya (npm start)).

21. Membuat import untuk BrowserRouter dan komponen meletakkan komponen Nav, Home, Kontak, Sejarah, Tentang di dalam <BrowserRouter>.

22. Router, untuk menautkan suatu komponen dan setelah ditautkan maka akan memunculkan hasil dari komponen yang sudah ditautkan sebelumnya (bila dipanggil, maka akan memunculkan apa yang sudah diatur sebelumnya). Menuliskannya, <Route path=’/’ component = {Home} />.

23. Exact, memunculkan komponen yang diberikan router berurutan dari atas lalu ke bawahnya (bila tidak dipanggil komponen lainnya, maka akan selalu memunculkan home page, namun bila dipanggil pada halaman komponen lainnya, maka komponen tersebut yang akan dimunculkan/ditampilkan).

24. Link (Link to=”/….” (titik-titik tersebut berisi nama kompononen yang akan disambungkan nantinya)), untuk menyambungkan masing-masing komponen agar saat dipanggil/diklik akan memunculkan/menampilkan halaman dari tiap masing-masing komponennya (bila yang di klik kontak maka yang muncul juga akan kontak (di halaman browsernya)).

25. Bisa mengganti title pada index.html nya (sesuai dengan yang ada di video).

26. Bisa menambahkan css sesuai dengan yang ada pada video (memberikan background color, memberi warna tulisannya (seperti h1) dan yang lainnya). Ditambahkan di dalam file index.html dari react-jsnya (menambahkan link rel untuk css).

27. Bisa menambahkan css dengan file yang berbeda maupun dalam satu file yang sama (hasilnya akan tetap sama).

28. Memberikan css pada satu file yang sama (inline) dengan cara memilih pada bagian mana yang akan diberikan css (misalkan p seperti yang ada di video), kemudian diberikan syle ={} (<p style = {{color:”red”, background:”blue”, fontSize:50}}Sidoarjo ></p>). Hasilnya akan menjadikan tulisan Sidoarjo memiliki background berwarna biru, tulisannya berwarna merah, dan ukuran tulisannya menjadi sebesar 50. Bisa dipisahkan dengan const (konstanta) yang telah dibuat dalam satu file yang masih sama, jadi nanti yang dipanggil dalam stylenya ialah nama dari const nya (hasilnya tetap akan sama seperti sebelumnya).

29. Pada saat memberikan css dalam satu file yang sama, misalkan seperti penulisan font-size, di dalam react-js tidak akan bisa digunakan bila ada tanda strip/minusnya (akan error nantinya), menggunakannnya dengan cara menghilangkan tanda strip/minus (-) tersebut dan digantikan dengan menuliskan huruf s pada size dengan huruf kapital (menjadi fontSize).

30. Bisa menggunakan nama selain perbagiannya (seperti h1) dengan cara menambahkan className=”….” (titik-titik tersebut berisikan nama yang nantinya akan digunakan dalam menggunakan cssnya, jadi yang dipanggil nanti misalkan bernama judul, .judul. Dan yang terganti adalah bagian yang sudah diberikan className tersebut, selain yang tidak diberikan maka tidak akan ikut berubah cssnya (pada video diberikan pada h1, maka pada bagian h1 background colornya menjadi berubah warna sesuai warna yang sudah dipilih sebelumnya dalam css)).

31. List-style none, untuk menghilangkan bulatan atau style yang ada dalam ul li (menjadi tidak ada bulatannya namun fungsi dari ul li nya masih tetap bisa berjalan (masih bisa digunakan seperti sebelumnya)).

32. Bisa menambahkan bootstrap sesuai dengan yang ada pada video (menggunakan bootstrap online). Menambahkannya sama seperti css di dalam file index.html (menggunakan link href untuk bootstrap).

33. Membungkus div id=”root” pada index.html dengan div container (agar rapid an berjalan dengan baik bootstrapnya).

34. Bisa menambahkan button (tombol berbentuk kotak) dengan warna biru dan hijau (biru pada primary dan hijau pada succes). Menambahkannya sesuai dengan yang ada di video (pada komponen tentang).

35. Bisa membenarkan error dari console (inspect) sesuai dengan yang ada di video.

36. Penulisan class dalam react-js menggunakan className, bila tidak maka bisa terjadi error pada consolenya.

37. Props (properti), hampir mirip seperti function, ditambahkan dalam params (parameter dari function) untuk memanggil isi dari teks yang ada di file lain, {props. ….} (titik-titik tersebut berisikan nama dari isi variabel yang telah dibuat sebelumnya). Pada file lain ini diberikan nama, misalkan judul = “Daftar Siswa”, maka nanti pada saat memanggilnya di file lain dengan cara {props.judul}. Bisa menggunakan const juga, jadi membuat const terlebih dahulu dengan isinya, dan pada saat pemanggilan (yang judul=””) nanti yang dipanggil ialah nama dari const yang telah dibuat sebelumnya tadi.

38. Map, untuk mengambil hasil dari array yang telah dibuat sebelumnya dan hasil saat ditampilkannya membuat baris baru yang isinya juga berupa array (hasil yang diambil tadi). Namun pada react-js, penggunaan dari mapnya harus diberikan key (bagian indeksnya), agar tidak error pada consolenya (inspect).

39. OnClick, bila mana pada saat diklik akan memunculkan isi yang ada di dalamnya (sudah dibuat sebelumnya).

40. Bisa menambahkan dan mengurangkan angka pada saat di klik dengan button yang telah dibuat sebelumnya. Bila ++ maka angka akan naik satu (1) keatas dan bila --, maka angka akan turun satu (1) kebawah. Angka tersebut seiring bertambah maupun berkurangnya akan muncul di console (bagian inspect).

41. State, untuk digunakan variabel bila mana diubah-ubah ketika hasil dari renderingnya sudah selesai (setelahnya), berbeda dengan props.

42. Bila sudah diberikan setCount, maka nanti pada bagian yang sudah dituliskan (count : .. (titik-titik tersebut berisikan hasil dari angka yang sudah di klik dari button, baik saat ditambah maupun dikurangi)) agar memunculkan hasil dari setiap button yang diklik akan muncul disana (seperti hasil pasa console dari inspect, namun dimunculkan langsung di halaman depan browser atau webnya).

43. If, dua kondisi yang di mana bila kondisi yang pertama benar akan dimunculkan, dan bila selain kondisi yang pertama akan masuk pada kondisi keduanya (else).

44. Menggunakan if agar pada saat mengklik button pengurangan, hasilnya tidak akan lebih dari 0 (tidak akan menjadi – (minus) angkanya).

45. UseState, function yang dapat diisi dengan variabel, untuk memunculkan suatu hasil setelah di rendering. Digunakan bila ada sesuatu hal yang ingin diganti, maka hasil yang diganti itulah yang akan diubah, yang lainnya tidak ikut di loading (render) ulang. Agar hasil setelah merefresh hasil barunya tidak terlalu lama (loadnya).

46. Mengopi isi dari shoppingcart.js (bagian object menu yang akan digunakan kembali, bagian idmenu, idkategori, menu, gambar, dan harga).

47. Reuse, menggunakan ulang hasil dari suatu komponen yang lain dengan cara menggunakan props untuk menyambungkannya.

48. Template, tempat yang digunakan untuk menampilkan (mengisi) data.

49. P (paragraf), untuk diisi dari kalimat yang akan dimunculkan.

50. Table (berisikan tr dan td untuk selanjutnya (di dalamnya)), digunakan untuk membuat tabel.

**Saya Belum Mengerti**

1. Masih sedikit bingung dengan video terakhir (reuse komponen).

2.

3.