

Praktikum 2 - Matakuliah Pilihan 1 (Web)

Program Studi: Teknik Informatika

Lakukan praktikum dibawah ini, dan buat screenshot untuk pembuktian mengerjakan setiap poin dengan mengisi tabel dibawah, kemudian tunjukan hasil akhir dari men-share repository github yang telah dibuat.

A. Membuat Project [Next.js](#)

1. Buka command prompt atau lewat visual code
2. Ketik : `npx create-next-app@latest praktikum2`
3. Jika ada pilihan Typescript → Yes, App Router → Yes, Tailwindcss → Yes, include src → Yes.
4. Masuk ke folder proyek anda dengan `cd praktikum2`
5. Jalankan proyek dengan mengetik `npm run dev` (Pastikan di dalam alamat folder proyek)
Cek gunakan browser sesuai port yang dibuka dalam cmd.

B. Membuat dan Menggunakan components

1. Buat folder baru di dalam app dengan nama component. Sehingga strukturnya seperti ini `/src/app/components`
2. Pada folder components, buat file baru dengan nama `Navbar.tsx` dengan berisi kode seperti dibawah ini: (`/src/app/components/Navbar.tsx`)

```
src > app > components > Navbar.tsx > Navbar
1  import Link from "next/link";
2
3  export default function Navbar() {
4    return (
5      <nav style={{
6        padding: "10px",
7        background: "#333",
8        color: "#fff",
9        display: "flex",
10       gap: "20px"
11     }}>
12        <Link href="/" style={{ color: "#fff", textDecoration: "none" }}>Home</Link>
13        <Link href="/about" style={{ color: "#fff", textDecoration: "none" }}>About</Link>
14        <Link href="/blog/1" style={{ color: "#fff", textDecoration: "none" }}>Blog</Link>
15        <Link href="/contact" style={{ color: "#fff", textDecoration: "none" }}>Contact</Link>
16      </nav>
17    );
18  }
```

3. Pada file `page.tsx` di app, (`/src/app/page.tsx`) ubah kode program seperti gambar dibawah ini (kode default dihapus semua)

```
src > app > page.tsx > Home
1  import Navbar from "../components/Navbar";
2
3  export default function Home() {
4    return (
5      <Navbar></Navbar>
6    );
7  }
8
9
10
```

4. Cek lewat browser, apa yang ditampilkan? Screenshot hasilnya.
5. Kunjungi website tailwind flowbite, ambil salah satu component Navbar dan terapkan di dalam kode program Navbar.tsx (<https://flowbite.com/docs/components/navbar/>)
6. Kode HTML dari flowbite dimasukkan ke dalam **tag return{ ... }**, dan "class" di replace menjadi "className" gunakan ctrl+H (Replace) agar mempermudah penulisan kode program.
7. Ganti menu menjadi Home, Blog, About dan contact dengan link menuju /, /blog/1, /about, /contact
8. Screenshot hasil dari Navbar dengan component flowbite.

C. Membuat dan Menggunakan Dynamic Components

1. Buat sebuah folder di dalam app dengan nama **about**. Sehingga strukturnya menjadi /src/app/about
2. Buat sebuah folder lagi di dalam folder about dengan nama components, sehingga strukturnya /src/app/about/components
3. Buat file Card.tsx di dalam folder components tersebut. Sehingga lokasinya ada di /src/app/about/components/Card.tsx
4. **(Warning!!) Card.tsx berbeda folder dengan Navbar.tsx**
5. Isi Card.tsx dengan kode program seperti screenshot dibawah ini

```
src > app > about > components > Card.tsx > Card
1
2  export default function Card({ title, description }) {
3    return (
4      <div style={{ border: "1px solid #ddd", margin: "10px", padding: "10px" }}>
5        <h2>{title}</h2>
6        <p>{description}</p>
7      </div>
8    );
9  }
10
```

6. Buat **File Baru** pages.tsx di dalam folder about tersebut, /src/app/about/page.tsx
7. **(Warning!!)** pages.tsx **tidak sama** dengan page.tsx pada poin C. page.tsx ini ada di dalam folder /about/
8. Isi pages.tsx dengan kode program seperti dibawah ini:

```
src > app > about > page.tsx > ...
1  import Card from "../components/Card";
2
3  export default function Home() {
4    return (
5      <div>
6        <Card title="Komponen 1" description="Ini adalah komponen pertama" />
7        <Card title="Komponen 2" description="Ini adalah komponen kedua" />
8      </div>
9    );
10
11 }
```

9. Gunakan browser dan menuju ke link localhost:3000/about . Screenshot hasilnya
10. Tambahkan di pages.tsx tersebut, dengan Navbar pada components navbar di parent, caranya dengan menulis Navbar di return, nanti visual code akan otomatis mengimport Navbar di atas kode program.

```
src > app > about > page.tsx > Home
1  import Navbar from "../components/Navbar";
2  import Card from "../components/Card";
3
4  export default function Home() {
5    return (
6      <>
7      <Navbar/>
8      <div>
9        <Card title="Komponen 1" description="Ini adalah komponen pertama" />
10       <Card title="Komponen 2" description="Ini adalah komponen kedua" />
11     </div>
12
13   </>
14 );
15 }
```

11. Penulisan <> dan </> artinya untuk menambahkan komponen-komponen yang berlainan, dan agar posisi pada html berurutan dari atas kebawah dengan membuat line baru.
12. Screenshot hasilnya.
13. Ubah Card.tsx dengan menggunakan komponen card dari flowbite <https://flowbite.com/docs/components/card/> (Pilih salah satu)
14. Tidak lupa untuk mengganti class ke className dan menambahkan {title} dan {description}
15. Screenshot hasilnya.

D. Dynamic Routes

1. Dalam poin sebelumnya kita sudah menerapkan Static Routes, seperti /about. Pada poin kali ini kita akan mempraktekan dynamic routes
2. Buat folder blog di dalam /src/app. Sehingga strukturnya seperti ini /src/app/blog
3. Buat folder [id] di dalam folder blog, sehingga seperti ini /src/app/blog/[id]
4. Buat file dengan nama page.tsx di dalam folder [id] Sehingga file tersebut memiliki lokasi seperti ini /src/app/blog/[id]/page.tsx
5. Tulis kode program pada file [id]/page.tsx seperti pada kode program dibawah ini

```
src > app > blog > [id] > ✎ page.tsx > ...  
1  
2  
3 interface BlogProp {  
4   |   params: { id: string };  
5   |  
6   |  
7   export default function BlogDetail({ params }: BlogProp) {  
8     |   return <h1>Detail Blog dengan ID: {params.id}</h1>;  
9     |  
10    |
```

6. Coba menggunakan browser untuk menampilkan hasilnya dengan mengunjungi localhost:3000/blog/1 localhost:3000/blog/23
localhost:3000/blog/Indonesia-Raya
7. Screenshot hasilnya
8. Bagaimana menerapkan 2 params misalkan id dan title?
9. Buat folder baru [title] dan letakan page.tsx di dalamnya sehingga struktur file dan foldernya menjadi seperti ini [id]/[title]/page.tsx
10. Pada kode program page.tsx tambahkan params: {id: string; title: string};
Dan pada return <h1> tambahkan params.title terserah ditulis dimana.
11. Coba menggunakan browser untuk menampilkan hasilnya dengan mengunjungi localhost:3000/blog/1945/Indonesia-Raya
12. Tambahkan folder components, tambahkan Card dari flowbite, seperti poin C-13.
ubah id dan title untuk mengisi component card tersebut.
13. Nama file dan fungsi tidak boleh sama. Karena ada di folder blog, maka gunakan nama BlogCard seperti pada kode di bawah ini

```
src > app > blog > [id] > [title] > components > BlogCard.tsx > ...
1
2 export default function BlogCard({ title, description }) {
3   return (
4     <div style={{ border: "1px solid #ddd", margin: "10px", padding: "10px" }}>
5       <h2>{title}</h2>
6       <p>{description}</p>
7     </div>
8   );
9 }
10
```

14. Screenshot hasil dari pengerjaan ini.

E. Github + Visual Code

1. Buat proyek di Github dengan nama **Latihan2**

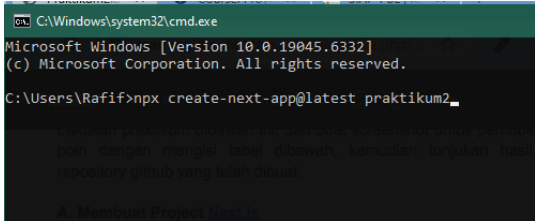
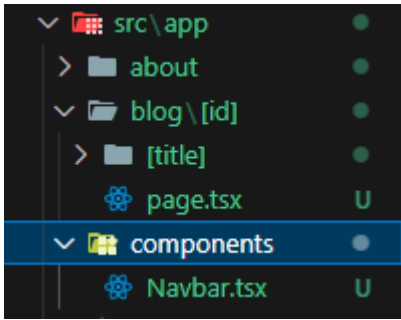
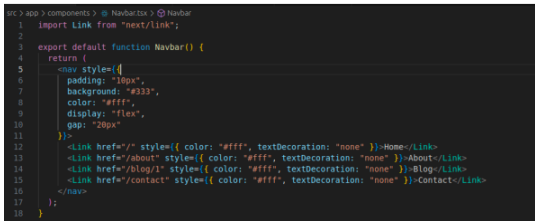
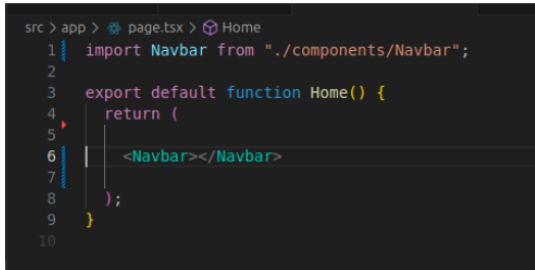
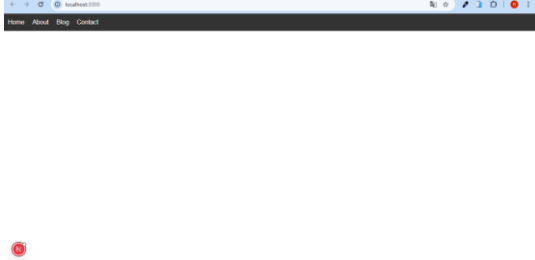
git init


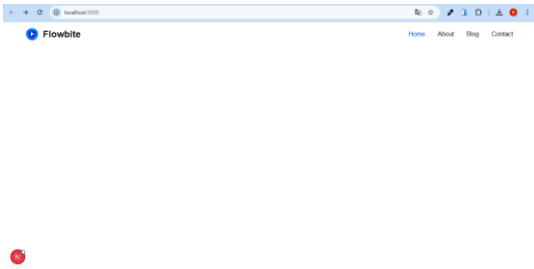
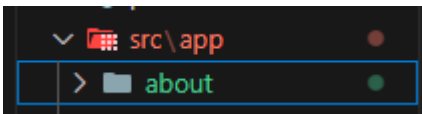
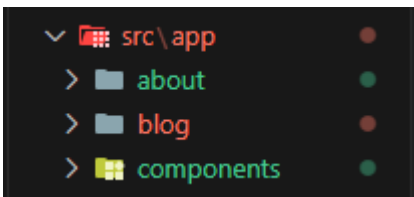
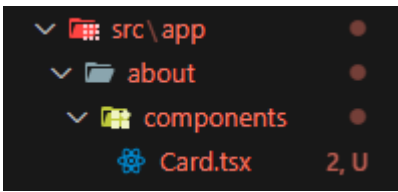

git add

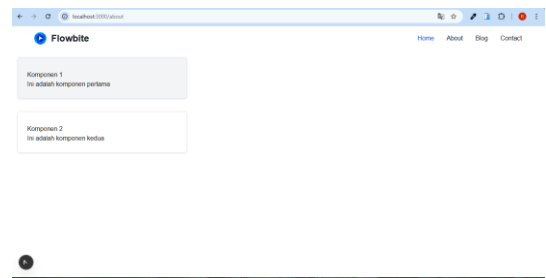
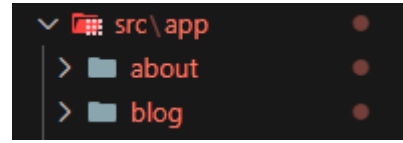
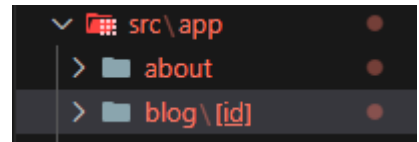
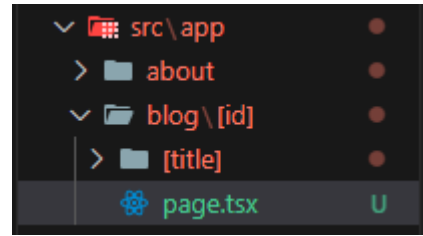
.

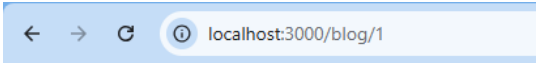
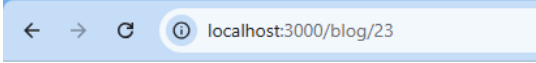
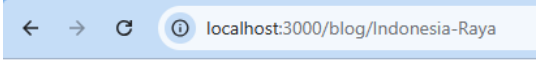
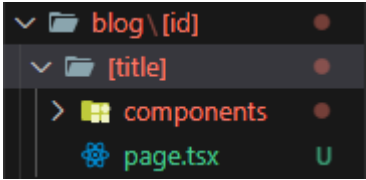
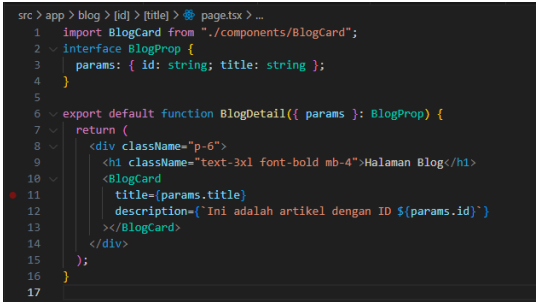
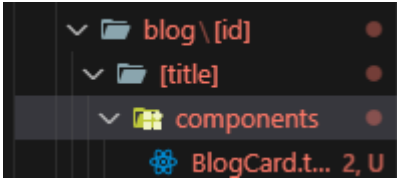


git commit -m "first commit" git branch -M main git remote add
origin https://github.com/agunghakase/Latihan1.git git push -u
origin main

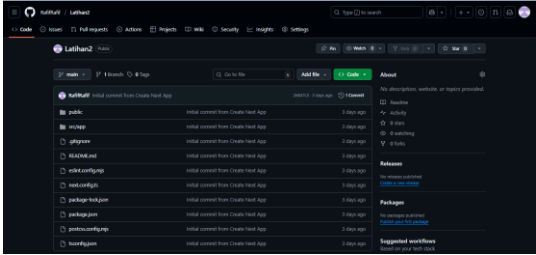
Hasil Pengerjaan

No.	Instruksi	Screenshot	Kendala/Saran
A.	Membuat proyek		
1.	Membuat proyek dengan nama praktikum2	 <pre> C:\Windows\system32\cmd.exe Microsoft Windows [Version 10.0.19045.6332] (c) Microsoft Corporation. All rights reserved. C:\Users\Rafif>npx create-next-app@latest praktikum2_ </pre>	
B.	Membuat dan Menggunakan components		
1.	Buat folder baru di dalam app dengan nama component. Sehingga strukturnya seperti ini /src/app/components		
2.	Pada folder components, buat file baru dengan nama Navbar.tsx dengan berisi kode seperti dibawah ini: (/src/app/components/Navbar.tsx)	 <pre> src > app > components > @ Navbar.tsx 1 import Link from "next/link"; 2 3 export default function Navbar() { 4 return (5 <nav style={{ 6 padding: "10px", 7 background: "#333", 8 color: "white", 9 display: "flex", 10 gap: "20px" 11 }}> 12 <Link href="/" style={{ color: "white", textDecoration: "none" }}>Home</Link> 13 <Link href="/about" style={{ color: "white", textDecoration: "none" }}>About</Link> 14 <Link href="/blog/" style={{ color: "white", textDecoration: "none" }}>Blog</Link> 15 <Link href="/contact" style={{ color: "white", textDecoration: "none" }}>Contact</Link> 16 </nav> 17); 18 } </pre>	
3.	Pada file page.tsx di app, (/src/app/page.tsx) ubah kode program seperti gambar dibawah ini (kode default dihapus semua)	 <pre> src > app > @ page.tsx > @ Home 1 import Navbar from "../components/Navbar"; 2 3 export default function Home() { 4 return (5 <Navbar/> 6); 7 } 8 9 10 </pre>	
4.	Cek lewat browser, apa yang ditampilkan?		Sebelum menggunakan flowbite

5.	Kunjungi website tailwind flowbite, ambil salah satu component Navbar dan terapkan di dalam kode program Navbar.tsx		
6.	Screenshot hasil dari Navbar dengan component flowbite		Sesudah menggunakan flowbite
.C.	Membuat dan Menggunakan Dynamic Components		
1.	Buat sebuah folder di dalam app dengan nama about. Sehingga strukturnya menjadi /src/app/about		
2.	Buat sebuah folder lagi di dalam folder about dengan nama components, sehingga strukturnya /src/app/about/components		
3.	Buat file Card.tsx di dalam folder components tersebut. Sehingga lokasinya ada di /src/app/about/components/Card.tsx		
4.	Isi Card.tsx dengan kode program		

11.	Screenshot hasilnya		
D.	Dynamic Routes		
1.	Buat folder blog di dalam /src/app. Sehingga strukturnya seperti ini /src/app/blog		
2.	Buat folder [id] di dalam folder blog, sehingga seperti ini /src/app/blog/[id]		
3.	Buat file dengan nama page.tsx di dalam folder [id] Sehingga file tersebut memiliki lokasi seperti ini /src/app/blog/[id]/page.tsx		
4.	Tulis kode program pada file [id]/page.tsx seperti pada kode program dibawah ini	<pre>src > app > blog > [id] > page.tsx > ... 1 2 3 interface BlogProp { 4 params: { id: string }; 5 } 6 7 export default function BlogDetail({ params }: BlogProp) { 8 return <h1>Detail Blog dengan ID: {params.id}</h1>; 9 } 10</pre>	

5.	Coba menggunakan browser untuk menampilkan hasilnya dengan mengunjungi localhost:3000/blog/1 localhost:3000/blog/23 localhost:3000/blog/Indonesia-Raya	 Detail Blog dengan ID: 1  Detail Blog dengan ID: 23  Detail Blog dengan ID: Indonesia-Raya	
6.	Buat folder baru [title] dan letakan page.tsx di dalamnya sehingga struktur file dan foldernya menjadi seperti ini [id]/[title]/page.tsx		
7.	Pada kode program page.tsx tambahkan params: {id: string; title: string}; Dan pada return tambahkan params.title terserah ditulis dimana.		
8.	Tambahkan folder components, sehingga menjadi [id]/[title]/components/Blog Card.tsx		
9.	Tulis kode program pada file [id]/[title]/components/Blog Card.tsx seperti pada kode program digambar ini		
10.	Ubah BlogCard.tsx dengan menggunakan komponen card dari flowbite		

B.	Github dan Viscode		
1.			
2.			
...			