

Praktikum 1: Mendefinisikan Komponen

Langkah 1: Buat Folder Baru

Buatlah folder baru bernama **belajar-komponen** lalu di dalam folder tersebut, jalankan terminal dengan mengetikkan perintah berikut:

```
npx create-next-app
```

```
PS E:\pembelajaran_polienama> npx create-next-app
✓ What is your project named? ... my-app
✓ Would you like to use TypeScript? ... No / Yes
✓ Would you like to use ESLint? ... No / Yes
✓ Would you like to use Tailwind CSS? ... No / Yes
✓ Would you like your code inside a `src` directory? ... No / Yes
✓ Would you like to use App Router? (recommended) ... No / Yes
✓ Would you like to use Turbopack for `next dev`? ... No / Yes
✓ Would you like to customize the import alias (`@/*` by default)? ... No / Yes
Creating a new Next.js app in E:\pembelajaran_polienama\my-app.

Using npm.

Initializing project with template: app-tw

Installing dependencies:
- react
- react-dom
- next

Installing devDependencies:
- @tailwindcss/postcss
- tailwindcss

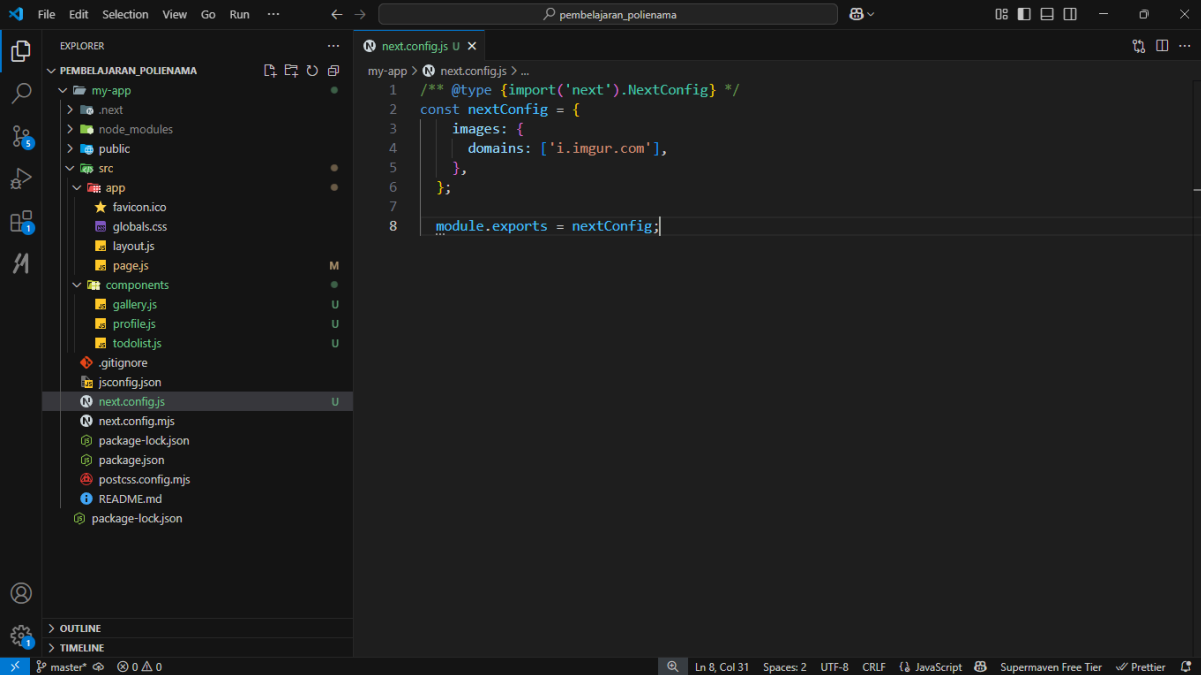
added 42 packages, and audited 43 packages in 3m

9 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New major version of npm available! 10.7.0 -> 11.3.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.3.0
npm notice To update run: npm install -g npm@11.3.0
npm notice
Initialized a git repository.

Success! Created my-app at E:\pembelajaran_polienama\my-app
```

Karena disini menggunakan format (.js) maka perlu melakukan setup terlebih dahulu seperti confignya



The screenshot shows the Visual Studio Code editor interface. On the left, the Explorer sidebar displays the file structure of a project named 'PEMBELAJARAN_POLIENAMA'. The project contains a 'my-app' directory with subdirectories like '.next', 'node_modules', 'public', 'src', and 'app'. The 'next.config.js' file is highlighted in the Explorer. The main editor area shows the content of 'next.config.js', which is a JavaScript file used for configuring Next.js. The code includes a JSDoc comment, a constant 'nextConfig' with an 'images' object containing a 'domains' array, and an export statement.

```
next.config.js U
my-app > next.config.js > ...
1  /** @type {import('next').NextConfig} */
2  const nextConfig = {
3    images: {
4      domains: ['i.imgur.com'],
5    },
6  };
7
8  module.exports = nextConfig;
```

The status bar at the bottom indicates the current file is 'next.config.js' at line 8, column 31, using 2 spaces, UTF-8 encoding, CRLF line endings, and JavaScript syntax. It also shows the 'Supremaven Free Tier' and 'Prettier' extensions.

Langkah 2: Buat Komponen Baru

Buatlah folder baru dan file baru di `src/components/profile.js` lalu ketik kode berikut ini.

```
import Image from "next/image";

export default function Profile() {
  return (
    <Image
      src="https://i.imgur.com/MK3eW3Am.jpg"
      alt="Katherine Johnson"
      width={100}
      height={100}
      style={{
        maxWidth: "100%",
        height: "auto",
        margin: "13px"
      }}
    />
  );
}
```

Kode tersebut menggunakan komponen `Image` bawaan dari [Next.js](#).

Komponen React adalah fungsi JavaScript biasa, tetapi nama mereka harus dimulai dengan huruf kapital atau tidak akan berfungsi!

Langkah 3: Import Komponen

Lakukan import komponen `Profile` ke `src/app/page.js`

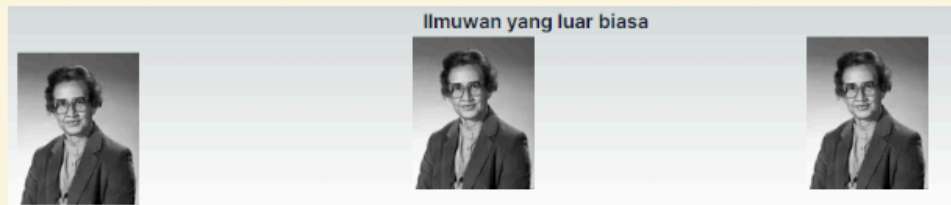
```
import Profile from "../components/profile";

export default function Home() {
  return (
    <div className="grid grid-rows-[20px_1fr_20px]
      items-center justify-items-center min-h-screen p-8 pb-20
      gap-16 sm:p-20 font-[family-name:var(--font-geist-sans)]">
      <main className="flex flex-col gap-[32px]
        row-start-2 items-center sm:items-start">
        <Profile />
      </main>
    </div>
  );
}
```

```
}
```

Soal 1

Ubah isi kode `Home()` sehingga dapat tampil seperti berikut dengan memanfaatkan komponen `Profile()` yang tadi sudah dibuat dari langkah 1 tersebut!



Catatan:

Jika terjadi error, Anda dapat mengakses tautan ini sebagai petunjuk solusinya:

<https://nextjs.org/docs/app/building-your-application/optimizing/images#remote-images>

Menyusun secara bertingkat dan mengorganisir komponen

Komponen adalah fungsi JavaScript biasa, sehingga Anda bisa menjaga beberapa komponen di dalam file yang sama. Ini nyaman ketika komponen-komponen relatif kecil atau saling terkait secara erat. Jika file ini mulai memiliki sangat banyak komponen, Anda bisa memindahkan komponen `Profile` ke suatu file yang berbeda. Anda akan belajar bagaimana cara melakukan ini segera pada bagian codelab ini berikutnya.

Karena komponen-komponen `Profile` di-render di dalam `Gallery`—bahkan beberapa kali!—kita dapat mengatakan bahwa `Gallery` adalah sebuah komponen induk, yang me-render setiap `Profile` sebagai sebuah "anak". Ini merupakan bagian ajaib dari React: Anda bisa mendefinisikan suatu komponen sekali, kemudian digunakan di banyak tempat dan sebanyak yang Anda suka.

Perhatian!

Komponen dapat me-render komponen lain, tetapi Anda tidak boleh menyusun definisinya secara bertingkat (bersarang):

```
export default function Gallery() {  
  
  // 🚫 Jangan mendefinisi suatu komponen di dalam komponen lain!  
  
  function Profile() {  
  
    // ...  
  
  }  
  
  // ...  
  
}
```

Potongan kode di atas [sangat lambat dan mengakibatkan bug](#). Sebagai gantinya, definisikan setiap komponen di tingkat atas atau di luar fungsi komponen lainnya:

```
export default function Gallery() {  
  
  // ...  
  
}  
  
// ✅ Deklarasikan komponen di tingkat atas  
  
function Profile() {  
  
  // ...  
  
}
```

Ketika suatu komponen anak membutuhkan data dari suatu induk, [operkan data melalui props](#) daripada menyusun definisinya secara bertingkat.

Praktikum 2: Mengimpor dan Mengekspor Komponen

Selesaikan langkah-langkah praktikum berikut ini menggunakan editor Visual Studio Code (VS Code) atau code editor lain kesukaan Anda. lanjutkan praktikum Anda pada setiap soal yang ada di beberapa langkah praktikum ini.

Perhatian: Melanjutkan project praktikum 1.

Langkah 1: Buat Komponen Baru

Buatlah file baru di `src/components/gallery.js` berisi kode seperti berikut:

```
import Profile from "../profile";

export function Gallery() {

  return (

    <div className="columns-3">

      <Profile />

      <Profile />

      <Profile />

    </div>

  );

}
```

Langkah 2: Impor Komponen

Lakukan impor komponen di `src/app/page.js` seperti berikut ini. Hapus kode komponen lama `Profile`, lalu sesuaikan dengan komponen baru `Gallery` yang telah dibuat.

```
import { Gallery } from "@components/gallery";
```

Praktikum 3: Menggunakan JSX dinamis

Selesaikan langkah-langkah praktikum berikut ini menggunakan editor Visual Studio Code (VS Code) atau code editor lain kesukaan Anda. lanjutkan praktikum Anda pada setiap soal yang ada di beberapa langkah praktikum ini.

Perhatian: Melanjutkan project praktikum 2.

Langkah 1: Buat Komponen Baru

Buatlah file baru di `src/components/todolist.js` berisi kode seperti berikut:

```
const person = {  
  name: 'Gregorio Y. Zara',  
  theme: {  
    backgroundColor: 'black',  
    color: 'pink'  
  }  
};  
  
export default function TodoList() {  
  return (  
    <div style={person.theme}>  
      <h1>{person}'s Todos</h1>  
        
      <ul>
```

```

    <li>Improve the videophone</li>

    <li>Prepare aeronautics lectures</li>

    <li>Work on the alcohol-fuelled engine</li>

  </ul>

</div>

);
}

```

Soal 4

Kode di atas masih terdapat error, silakan diperbaiki.

Langkah 2: Impor Komponen

Lakukan impor komponen di `src/app/page.js` seperti berikut ini.

```

import { Gallery } from "../components/gallery";

import TodoList from "../components/todolist";

export default function Home() {

  return (

    <section>

      <h1 className="font-semibold text-slate-900 truncate pr-20 text-center">Ilmuwan yang luar biasa</h1>

      <hr />

      <Gallery />

      <hr />

      <TodoList />

    </section>
  )
}

```



```
);  
  
}
```

Soal 5

Buka file `src/components/todolist.js` lakukan ekstrak URL gambar ke dalam objek `(clue : {person. })`

Langkah 3: Ubah ekspresi URL di src

Tetap di file `src/components/todolist.js` ubahlah objek `person` dan tambah variabel `baseUrl` seperti di bawah ini. URL lengkap gambar dibagi menjadi empat bagian: URL dasar, `imageId`, `imageSize`, dan ekstensi file.

Kita ingin URL gambar menggabungkan atribut-atribut ini bersama-sama: URL dasar (selalu `'https://i.imgur.com/'`), `imageId` (`'7vQD0fP'`), `imageSize` (`'s'`), dan ekstensi file (selalu `'.jpg'`). Namun, ada yang salah dengan atribut `src`.

Bisakah anda memperbaikinya?

```
const baseUrl = 'https://i.imgur.com/';  
  
const person = {  
  name: 'Gregorio Y. Zara',  
  imageId: '7vQD0fP',  
  imageSize: 's',  
  theme: {  
    backgroundColor: 'black',  
    color: 'pink'  
  }  
};  
  
export default function TodoList() {  
  return (  

```

```

<div style={person.theme}>

  <h1>{person.name}'s Todos</h1>

  <ul>

    <li>Improve the videophone</li>

    <li>Prepare aeronautics lectures</li>

    <li>Work on the alcohol-fuelled engine</li>

  </ul>

</div>

);
}

```

Soal 6

Perbaiki kode tersebut pada bagian atribut `src`. Kode lainnya dapat Anda sesuaikan dari jawaban soal sebelumnya.

Catatan:

Untuk memeriksa apakah perbaikan Anda berhasil, coba ubah nilai dari `imageSize` menjadi `'b'`. Gambar harus menyesuaikan ukuran setelah diubah.

Praktikum 4: Menggunakan Props

Selesaikan langkah-langkah praktikum berikut ini menggunakan editor Visual Studio Code (VS Code) atau code editor lain kesukaan Anda. Lanjutkan praktikum Anda pada setiap soal yang ada di beberapa langkah praktikum ini.

Perhatian: Melanjutkan project praktikum 3. Jangan lupa untuk menambahkan komponen ke file `src/app/page.js` untuk melihat hasil akhirnya di web.

Langkah 1: Buat Komponen Baru

Buatlah file baru di `src/components/mygallery.js` berisi kode seperti berikut:

```
import { getImageUrl } from '../utils/utils';

export default function MyGallery() {

  return (

    <div>

      <h1>Notable Scientists</h1>

      <section className="profile">

        <h2>Maria Skłodowska-Curie</h2>

        <img

          className="avatar"

          src={getImageUrl('szV5sdG')}

          alt="Maria Skłodowska-Curie"

          width={70}

          height={70}

        />

        <ul>

          <li>

            <b>Profesi: </b>


```

```

        Fisikawan dan kimiawan

    </li>

    <li>

        <b>Penghargaan: 4 </b>

        (Penghargaan Nobel Fisika, Penghargaan Nobel
        Kimia, Medali Davy, Medali Matteucci)

    </li>

    <li>

        <b>Telah Menemukan: </b>

        polonium (unsur kimia)

    </li>

</ul>

</section>

<section className="profile">

    <h2>Katsuko Saruhashi</h2>

    <img

        className="avatar"

        src={getImageUrl('YfeOqp2')}

        alt="Katsuko Saruhashi"

        width={70}

        height={70}

    />

    <ul>

        <li>

```

```

        <b>Profesi: </b>

        Ahli Geokimia

    </li>

    <li>

        <b>Penghargaan: 2 </b>

        (Penghargaan Miyake Geokimia, Penghargaan
Tanaka)

    </li>

    <li>

        <b>Telah Menemukan: </b>

        sebuah metode untuk mengukur karbon dioksida
pada air laut

    </li>

</ul>

</section>

</div>

);
}

```

Soal 7

Jika kode di atas terdapat error, silakan diperbaiki.

Komponen `MyGallery` ini berisi dua markup yang sama persis. Ekstraklah menjadi komponen `MyProfile` untuk mengurangi duplikasi. Anda perlu memilih props apa saja yang akan dikirimkan.

Langkah 2: Buat fungsi getImageUrl

Buatlah folder baru dan file baru di `src/utlis/utlis.js` berisi kode seperti berikut:

```
export function getImageUrl(imageId, size = 's') {  
    return 'https://i.imgur.com/' + imageId + size +  
    '.jpg';  
}
```

Langkah 3: Buat komponen baru

Buatlah file baru di `src/components/myprofile.js` berisi kode seperti berikut:

```
import { getImageUrlV2 } from '../utlis/utlis';  
  
function MyAvatar({ person, size }) {  
    return (  
        <img  
            className="avatar"  
            src={getImageUrl(person, 'b')}  
            alt={person.name}  
            width={size}  
            height={size}  
        />  
    );  
}  
  
export default function MyProfile() {  
    return (  
        <MyAvatar
```

```

        size={40}

        person={{
          name: 'Gregorio Y. Zara',
          imageId: '7vQD0fP'
        }}
      />
    );
  }
}

```

Tambahkan fungsi `getImageUrlV2` seperti berikut di file yang sama yaitu [utils.js](#)

`src/utils/utils.js`

```

export function getImageUrlV2(person, size) {
  return 'https://i.imgur.com/' + person.imageId + size
+ '.jpg';
}

```

Pada contoh ini, komponen `MyAvatar` menerima prop `size` yang berupa angka dan akan digunakan untuk menentukan lebar dan tinggi ``. Prop `size` dibuat `40` pada contoh ini. Bagaimanapun, jika Anda membuka gambar tersebut di web, Anda akan melihat bahwa gambarnya akan lebih besar (`160` piksel). Ukuran gambar yang sebenarnya ditentukan oleh ukuran thumbnail yang Anda minta.

Soal 8

Jika kode di atas terdapat error, silakan diperbaiki.

Ubahlah komponen `MyAvatar` untuk diolah ukuran gambarnya berdasarkan prop `size`. Khususnya, jika `size` kurang dari `90`, kirimkan `'s'` ("small") bukan `'b'` ("big") pada fungsi `getImageUrl`. Pastikan bahwa perubahan yang Anda buat berjalan dengan cara me-render avatars dengan nilai prop `size` yang berbeda dan buka gambar pada tab baru di browser.

Langkah 4: Buat komponen baru `MyProfileV2`

Buatlah file baru di `src/components/myprofilev2.js` berisi kode seperti berikut:

```
import React from 'react';

import Card from '../Card';

export default function MyProfileV2() {

  return React.createElement(

    'div',

    null,

    React.createElement(

      Card,

      null,

      React.createElement('h1', null, 'Foto'),

      React.createElement('img', {

        className: 'avatar',

        src: 'https://i.imgur.com/OKS67lhm.jpg',

        alt: 'Aklilu Lemma',

        width: 70,

        height: 70

      })

    ),

    React.createElement(

      Card,

      null,

      React.createElement('h1', null, 'Tentang'),
```



```

    React.createElement (

      'p',

      null,

      'Aklilu Lemma adalah seorang ilmuwan terkemuka
dari Etiopia yang telah menemukan pengobatan alami untuk
skistosomiasis.'

    )

  )

);
}

```

Soal 9

Jika kode di atas terdapat error, silakan diperbaiki.

Buatlah komponen `Card` dari kode di atas, lalu gunakan prop `children` untuk mengirimkan JS yang berbeda.

Hasil

