

NRP : 5026221144
Nama : Alfa Renaldo Aluska
Matkul: Teknologi Berkembang (B)

Tugas:

Buatlah sebuah halaman baru di Flutter dengan nama AboutPage. Halaman ini harus menampilkan:

- Foto diri kalian bebas asal sopan
- Nama lengkap
- NRP
- Funfact kalian

Jawaban:

Aplikasi ini adalah sebuah profil pribadi yang menampilkan informasi tentang saya sendiri, yaitu *Alfa Renaldo Aluska*. Aplikasi ini memiliki beberapa bagian, seperti gambar profil, informasi dasar (nama, lokasi, dan NRP), deskripsi tentang saya (*fun fact*), serta dua tombol yang mengarah ke email dan website.

main.dart

```
import 'package:flutter/material.dart';
import 'package:url_launcher/url_launcher.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    const String appTitle = 'AboutPage';
    return MaterialApp(
      title: appTitle,
      home: Scaffold(
        appBar: AppBar(
          title: const Text(appTitle),
        ),
        body: const SingleChildScrollView(
          child: Column(
            children: [
```

```

ImageSection(image: 'assets/images/alfa.jpg'),
TitleSection(
    name: 'Alfa Renaldo Aluska',
    location: 'Bogor, Jawa Barat, Indonesia',
    nrp: "5026221144",
),
ButtonSection(),
TextSection(
    description: 'Halo, saya Alfa Renaldo Aluska!\n\n'
        'Perjalanan pendidikan saya dimulai di SMA Plus PGRI Cibinong, '
        'dengan fokus pada jurusan IPA. Setelah lulus, '
        'saya melanjutkan studi di Institut Teknologi Sepuluh Nopember, '
        'mengambil jurusan Sistem Informasi untuk memperdalam pengetahuan saya '
        'di bidang teknologi dan manajemen informasi.\n\n'
        'Lahir pada 6 Mei 2002, saya percaya bahwa setiap pengalaman adalah
peluang '
        'untuk belajar dan berkembang. Dengan semangat belajar dan tekad yang kuat,
        'saya terus berusaha memberikan yang terbaik dalam setiap langkah hidup
saya.\n\n'
        'Mari terhubung dan ciptakan hal-hal luar biasa bersama!',
),
],
),
),
),
);
}
}

// Kelas untuk menampilkan gambar
class ImageSection extends StatelessWidget {
    const ImageSection({super.key, required this.image});

    final String image;

    @override
    Widget build(BuildContext context) {
        return Image.asset(
            image,

```

```

        width: 600,
        height: 240,
        fit: BoxFit.cover,
    );
}
}

// Kelas untuk menampilkan informasi nama, lokasi, dan NRP
class TitleSection extends StatelessWidget {
    const TitleSection(
        {super.key,
        required this.name,
        required this.location,
        required this.nrp});

    final String name;
    final String location;
    final String nrp;

    @override
    Widget build(BuildContext context) {
        return Padding(
            padding: const EdgeInsets.all(32),
            child: Row(
                children: [
                    Expanded(
                        child: Column(
                            crossAxisAlignment: CrossAxisAlignment.start,
                            children: [
                                Padding(
                                    padding: const EdgeInsets.only(bottom: 8),
                                    child: Text(
                                        name,
                                        style: const TextStyle(fontWeight: FontWeight.bold),
                                    ),
                                ),
                                Text(
                                    location,
                                    style: TextStyle(color: Colors.grey[500]),
                                ),
                            ],
                        ),
                ],
            ),
        );
    }
}

```

```

        ],
      ),
    ),
    // Ikon untuk menandakan ini profil
    Icon(
      Icons.numbers,
      color: Colors.red[500],
    ),
    Text(nrp),
  ],
),
);
}

class ButtonSection extends StatelessWidget {
  const ButtonSection({super.key});

  // Fungsi untuk membuka email
  Future<void> _launchEmail() async {
    final Uri emailUri = Uri(
      scheme: 'mailto',
      path: 'renaldoaluska@gmail.com', // Ganti dengan alamat email kamu
      query: 'subject=Hello&body=How are you?', // Parameter tambahan
    );
    if (!await launchUrl(emailUri)) {
      throw Gagala launch $emailUri';
    }
  }

  // Fungsi untuk membuka website
  Future<void> _launchWeb() async {
    final Uri webUri = Uri.parse(
      'https://renaldo.my.id'); // Ganti dengan link website kamu
    if (!await launchUrl(webUri, mode: LaunchMode.externalApplication)) {
      throw Gagala launch $webUri;
    }
  }
}

@override

```

```

Widget build(BuildContext context) {
  final Color color = Theme.of(context).primaryColor;

  return SizedBox(
    child: Row(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: [
        // Button untuk email, jadi ketika klik bakal buka email
        TextButton.icon(
          onPressed: _launchEmail,
          icon: Icon(Icons.email, color: color),
          label: Text(
            'EMAIL',
            style: TextStyle(
              fontSize: 12, fontWeight: FontWeight.w400, color: color),
          ),
        ),
        // Button untuk membuka website
        TextButton.icon(
          onPressed: _launchWeb,
          icon: Icon(Icons.link, color: color),
          label: Text(
            'WEBSITE',
            style: TextStyle(
              fontSize: 12, fontWeight: FontWeight.w400, color: color),
          ),
        ),
      ],
    ),
  );
}

```

// Kelas untuk tombol dengan ikon dan teks di bawahnya

```

class ButtonWithText extends StatelessWidget {
  const ButtonWithText({
    super.key,
    required this.color,
    required this.icon,
    required this.label,

```

```

        this.onPressed,
    });

    final VoidCallback? onPressed;
    final Color color;
    final IconData icon;
    final String label;

    @override
    Widget build(BuildContext context) {
        return Column(
            mainAxisAlignment: MainAxisAlignment.min,
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
                Icon(icon, color: color),
                Padding(
                    padding: const EdgeInsets.only(top: 8),
                    child: Text(
                        label,
                        style: TextStyle(
                            fontSize: 12,
                            fontWeight: FontWeight.w400,
                            color: color,
                        ),
                    ),
                ),
            ],
        );
    }
}

// Kelas untuk menampilkan teks deskripsi
class TextSection extends StatelessWidget {
    const TextSection({
        super.key,
        required this.description,
    });

    final String description;

```

```
@override
Widget build(BuildContext context) {
  return Padding(
    padding: const EdgeInsets.all(32),
    child: Text(
      description,
      softWrap: true,
    ),
  );
}
```

Struktur Kode

1. Main Application (**MyApp**)

- **MyApp** adalah aplikasi utama yang menjalankan tampilan antarmuka dengan judul "AboutPage".
- **MaterialApp** digunakan untuk pengaturan tema dan struktur dasar.
- **Scaffold** digunakan untuk menampilkan struktur halaman dengan **AppBar** di bagian atas dan konten utama berupa **SingleChildScrollView**.

2. Image Section

- **ImageSection** adalah widget yang digunakan untuk menampilkan gambar profil (gambar **alfa.jpg**).
- Gambar ini ditampilkan dengan ukuran 600x240 px dan menggunakan metode **BoxFit.cover** untuk memastikan gambar menyesuaikan dengan ukuran yang ditentukan.

3. Title Section

- **TitleSection** adalah widget yang menampilkan informasi seperti nama, lokasi, dan NRP.
- Menggunakan **Row** untuk menampilkan data dalam bentuk kolom untuk nama dan lokasi, serta menambahkan ikon di sebelah NRP.

4. Button Section

- **ButtonSection** adalah widget yang berisi dua tombol:
 - Tombol pertama untuk membuka email menggunakan **mailto** scheme, mengarahkan ke email yang telah disesuaikan dengan subjek dan isi pesan.
 - Tombol kedua membuka website pribadi menggunakan URL yang ditentukan. Kedua tombol ini menggunakan **TextButton.icon** untuk menampilkan ikon dan teks di sebelahnya.

5. Text Section

- **TextSection** adalah widget untuk menampilkan teks deskripsi yang menjelaskan lebih lanjut tentang individu yang dimaksud (Alfa Renaldo Aluska).

- Deskripsi ini disertakan dengan paragraf yang memberikan latar belakang pendidikan, semangat belajar, dan ajakan untuk terhubung.
6. **Button with Text**
- Kelas `ButtonWithText` seharusnya digunakan untuk membuat tombol dengan ikon dan teks di bawahnya. Namun, dalam kode ini, tampaknya kelas ini tidak digunakan.
-

Row, Column, Padding, Text, Icon, TextButton, dan SizedBox

1. Penggunaan `Row`

- Dalam aplikasi ini, `Row` digunakan untuk menyusun elemen secara horizontal. Sebagai contoh, di dalam widget `TitleSection`, `Row` digunakan untuk menampilkan nama, lokasi, dan NRP. Nama dan lokasi ditempatkan di sebelah kiri, sedangkan NRP dan ikon ditampilkan di sebelah kanan. Semua elemen ini diatur dalam satu baris horizontal.
- Digunakan widget `Expanded` di dalam `Row` untuk memastikan bahwa ruang yang tersedia di bagian nama dan lokasi dapat diisi secara maksimal, sehingga NRP dan ikon di sebelahnya tidak berjarak terlalu dekat.

2. Penggunaan `Column`

- Widget `Column` digunakan untuk menyusun elemen secara vertikal. Misalnya, di dalam `TitleSection`, `Column` digunakan untuk menampilkan nama dan lokasi dalam satu kolom, sehingga keduanya ditampilkan satu di bawah yang lain dengan jarak yang cukup agar tidak terkesan terlalu rapat.
- Selain itu, di dalam `TextSection`, `Column` juga digunakan untuk menampilkan deskripsi secara vertikal, sehingga teks dapat dibaca dengan nyaman dalam satu kolom tanpa terpotong.

3. Penggunaan `Padding`

- Widget `Padding` sering digunakan di aplikasi ini untuk memberi ruang atau jarak di sekitar elemen, seperti teks dan gambar. Sebagai contoh, pada `TitleSection`, padding 32 pixel untuk sekitar nama dan lokasi agar teks tersebut tidak terlalu dekat dengan batas layar atau elemen lainnya. Hal ini membuat tampilan lebih rapi dan nyaman untuk dibaca.

4. Penggunaan `Text`

- Widget `Text` digunakan untuk menampilkan teks di berbagai bagian aplikasi, seperti nama, lokasi, deskripsi, dan NRP. Setiap teks ditempatkan di dalam widget `Column` atau `Row`, sesuai dengan pengaturan tata letak yang diinginkan.

5. Penggunaan `Icon`

- Widget `Icon` digunakan untuk menambahkan elemen visual yang memudahkan untuk mengidentifikasi fungsionalitas tertentu. Misalnya, ikon angka ditampilkan di samping NRP untuk membedakan antara teks dan elemen visual. Ikon juga digunakan dalam tombol untuk email dan website, memberikan petunjuk visual bahwa tombol tersebut dapat digunakan untuk membuka aplikasi email atau situs web.

6. Penggunaan `TextButton`

- Widget **TextButton** digunakan untuk membuat tombol yang dapat diklik. Di dalam kode, **TextButton.icon** digunakan untuk membuat tombol dengan ikon di samping teks. Contohnya, tombol untuk membuka aplikasi email dan website. Ketika tombol ini diklik, fungsi tertentu seperti membuka aplikasi email atau situs web akan dijalankan.

7. Penggunaan **SizedBox**

- Widget **SizedBox** digunakan untuk memberi jarak antar elemen atau mengatur ukuran elemen yang dibungkusnya. Di dalam **ButtonSection**, **SizedBox** digunakan untuk memastikan bahwa tombol-tombol seperti tombol untuk email dan website memiliki jarak yang cukup di antara satu sama lain, sehingga tampilannya tidak terlalu rapat atau berantakan.

Gambar

- Gambar yang ditampilkan di aplikasi ini diambil dari folder **assets/images** dengan menggunakan widget **Image.asset**. Di kode ini, gambar yang digunakan adalah **'assets/images/alfa.jpg'**. Ini berarti gambar **alfa.jpg** yang berada dalam folder **images** di dalam folder **assets** akan dimuat dan ditampilkan pada tampilan aplikasi.
- Untuk memastikan gambar bisa dimuat dengan benar, dideklarasikan folder **assets** di dalam file **pubspec.yaml**. Di dalam file ini, perlu menambahkan path ke folder **assets/images** agar Flutter tahu bahwa gambar tersebut akan digunakan dalam aplikasi.

pubspec.yaml

```
assets:  
  
- assets/images/alfa.jpg
```

Tombol Email

```
// Fungsi untuk membuka email  
Future<void> _launchEmail() async {  
  final Uri emailUri = Uri(  
    scheme: 'mailto',  
    path: 'renaldoaluska@gmail.com', // Ganti dengan alamat email kamu  
    query: 'subject=Hello&body=How are you?', // Parameter tambahan  
  );  
  if (!await launchUrl(emailUri)) {  
    throw 'Gagal launch $emailUri';  
  }  
}
```

pubspec.yaml

```
dependencies:  
  
  flutter:  
  
    sdk: flutter  
  
  url_launcher: ^6.3.1
```

Bagian yang mengatur klik untuk tombol **email** dan **website** berada di dalam kelas **ButtonSection**:

- Fungsi untuk membuka email berada di dalam fungsi `_launchEmail` yang didefinisikan di dalam **ButtonSection**.
- Pakai dependency `url_launcher` agar aplikasi dapat membuka URL eksternal, seperti membuka aplikasi email atau website, secara langsung dari dalam aplikasi. Ada import statement `import 'package:url_launcher/url_launcher.dart';` di paling atas dalam `main.dart`. Ini digunakan untuk mengimpor paket `url_launcher` yang dependency-nya sudah di-define di `pubspec.yaml`.
- **Penjelasan:**
 - Fungsi `_launchEmail()` membuat objek `Uri` dengan skema `mailto`, yang memungkinkan untuk membuka aplikasi email mereka secara langsung dengan alamat tujuan (`renaldoaluska@gmail.com`) dan subjek dan body email yang telah diisi secara otomatis.
 - Fungsi `launchUrl()` digunakan untuk mencoba membuka `Uri` ini menggunakan aplikasi email yang terpasang di perangkat. Jika gagal, maka aplikasi akan melempar error dengan pesan `'Gagal launch $emailUri'`.
 - Tombol untuk membuka email dibuat dengan `TextButton.icon`. Ikon email digunakan, dan label "EMAIL" ditampilkan di sampingnya. Ketika mengklik tombol ini, fungsi `_launchEmail()` akan dipanggil.

Tombol Website

- Fungsi untuk membuka website berada di dalam fungsi `_launchWeb`, yang juga didefinisikan di dalam **ButtonSection**.
- **Penjelasan:**
 - Fungsi `_launchWeb()` membuat objek `Uri` menggunakan `Uri.parse()` untuk mengonversi string URL (`'https://renaldo.my.id'`) menjadi objek `Uri`.
 - Fungsi `launchUrl()` kemudian digunakan untuk membuka URL tersebut menggunakan aplikasi eksternal, seperti browser. Parameter `LaunchMode.externalApplication` memastikan bahwa URL dibuka di aplikasi luar (misalnya browser, bukan di dalam aplikasi itu sendiri).

- Jika gagal membuka URL tersebut, maka aplikasi akan melempar error dengan pesan 'Gagal launch \$webUri'.
- Tombol untuk membuka website juga dibuat dengan `TextButton.icon`. Ikon link digunakan, dan label "WEBSITE" ditampilkan di sampingnya. Ketika mengklik tombol ini, fungsi `_launchWeb()` akan dipanggil.

Tampilan Akhir

