

# Codelab 1

Pada kelas ini kita akan mengembangkan sebuah aplikasi yang menampilkan tempat-tempat wisata di Bandung. Hasil akhir dari keseluruhan codelab akan seperti berikut:

3:57



DEB. 3

DEB. 3

## Wisata Bandung



Farm House Lembang  
Lembang



Observatorium Bosscha  
Lembang



Jalan Asia Afrika  
Kota Bandung



Stone Garden  
Padalarang



Taman Film Pasopati  
Kota Bandung



Museum Geologi  
Kota Bandung



Floating Market  
Lembang

Dalam codelab pertama ini kita akan membuat sebuah tampilan yang menggabungkan semua widget-widget yang sebelumnya kita pelajari. Tampilannya adalah seperti berikut:

11:26



DEVELOPER



DEVELOPER

## Farm House Lembang



Open Everyday



09:00 - 20:00

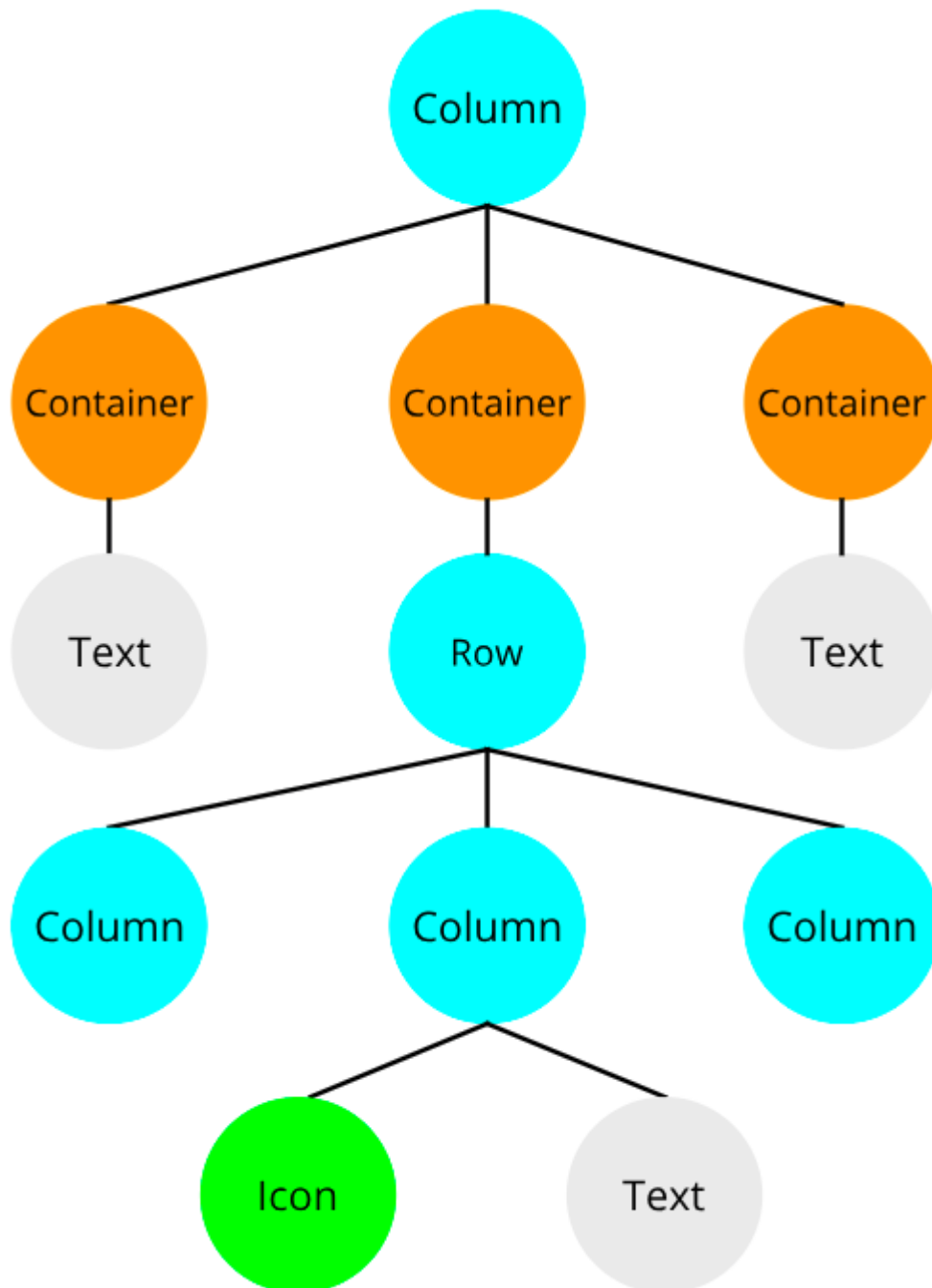


Rp 25.000

Berada di jalur utama Bandung-Lembang, Farm House menjadi objek wisata yang tidak pernah sepi pengunjung. Selain karena letaknya strategis, kawasan ini juga menghadirkan nuansa wisata khas Eropa. Semua itu diterapkan dalam bentuk spot swafoto Instagramable.

Sebelum kita membuat tampilan di atas, kita akan bedah terlebih dahulu *layout*-nya.

Pada layout di atas kita dapat memetakan widget-widget dalam bentuk diagram seperti di bawah ini:



1. Buat project Flutter baru dan berikan nama yang sesuai, misalnya wisata\_bandung. Hapus kode aplikasi counter yang diberikan ketika project dibuat.

2. Tuliskan kode dasar yang menampilkan widget **MaterialApp** seperti berikut:

```
1. import 'package:flutter/material.dart';
2.
3. void main() => runApp(MyApp());
4.
5. class MyApp extends StatelessWidget {
6.   @override
7.   Widget build(BuildContext context) {
8.     return MaterialApp(
9.       title: 'Wisata Bandung',
10.      theme: ThemeData(),
11.    );
12.  }
13. }
```

3. Lalu kita akan membuat kode untuk susunan widget sesuai diagram yang telah kita buat. Untuk membuat kode kita lebih rapi kita akan membuat kelas Stateless Widget baru untuk menampung kode tampilan kita. Mari namakan kelas ini **DetailScreen**.

```
1. class DetailScreen extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold();
5.   }
6. }
```

Jangan lupa untuk menambahkan widget **DetailScreen** sebagai home dari **MaterialApp**.

```
7. class MyApp extends StatelessWidget {
8.   @override
9.   Widget build(BuildContext context) {
10.    return MaterialApp(
11.      title: 'Wisata Bandung',
12.      theme: ThemeData(),
13.      home: DetailScreen(),
14.    );
15.  }
16. }
```

4. Sesuai diagram di atas, kita akan menyusun beberapa widget secara vertikal sehingga kita perlu menggunakan widget **Column**.

```
1. class DetailScreen extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold(
5.       body: Column(),
6.     );
7.   }
```

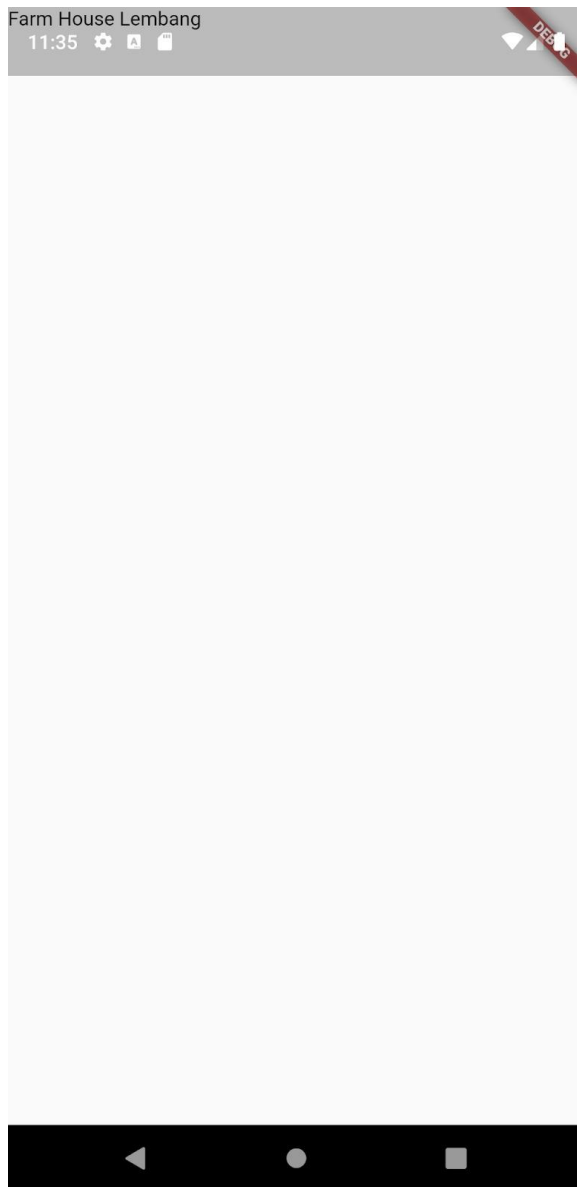
```
8. }
```

Jalankan aplikasi Anda. Saat ini device atau emulator Anda memang masih belum menampilkan apa pun. Namun, kita akan memanfaatkan fitur hot reload untuk melihat perubahan-perubahan yang akan kita lakukan ke depan.

5. Komponen pertama yang akan kita buat adalah bagian judul dari halaman. Tentunya untuk menampilkan teks kita akan menggunakan widget Text.

```
1. class DetailScreen extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return Scaffold(  
5.       body: Column(  
6.         children: <Widget>[  
7.           Text('Farm House Lembang'),  
8.         ],  
9.       ),  
10.    );  
11.  }  
12. }
```

6. Ketika Anda menyimpan project atau menjalankan hot reload, tampilan aplikasi Anda sekarang mungkin tidak sesuai dengan keinginan, seperti teks terlalu ke atas dan juga terlalu kecil.



Untuk itulah kita perlu membungkus widget Text ke dalam Container supaya kita dapat memberikan property seperti margin atau padding. Jika Anda menggunakan IDE Android Studio, Anda dapat memanfaatkan



shortcut **Alt+Enter** untuk membungkus widget ke widget lain.

```
class DetailScreen extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      body: Column(  
        children: <Widget>[  
          Text('Farm House Lembang'),  
        ], // <Widget>[]  
      ), // Column  
    ); // Scaffold  
  }  
}
```

7. Tambahkan margin atas supaya teks memiliki jarak terhadap bagian atas layar.

```
1. class DetailScreen extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return Scaffold(  
5.       body: Column(  
6.         children: <Widget>[  
7.           Container(  
8.             margin: EdgeInsets.only(top: 16.0),  
9.             child: Text('Farm House Lembang'),  
10.            ),  
11.          ],  
12.        ),  
13.      );  
14.    }  
15.  }
```

Pada kode di atas kita hanya memberikan margin atas sebesar 16.0. Anda dapat memanfaatkan metode `EdgeInsets` lain seperti `all()` untuk memberikan margin ke semua sisi atau `symmetric()` apabila Anda ingin memberikan margin ke sisi vertikal atau horizontal.

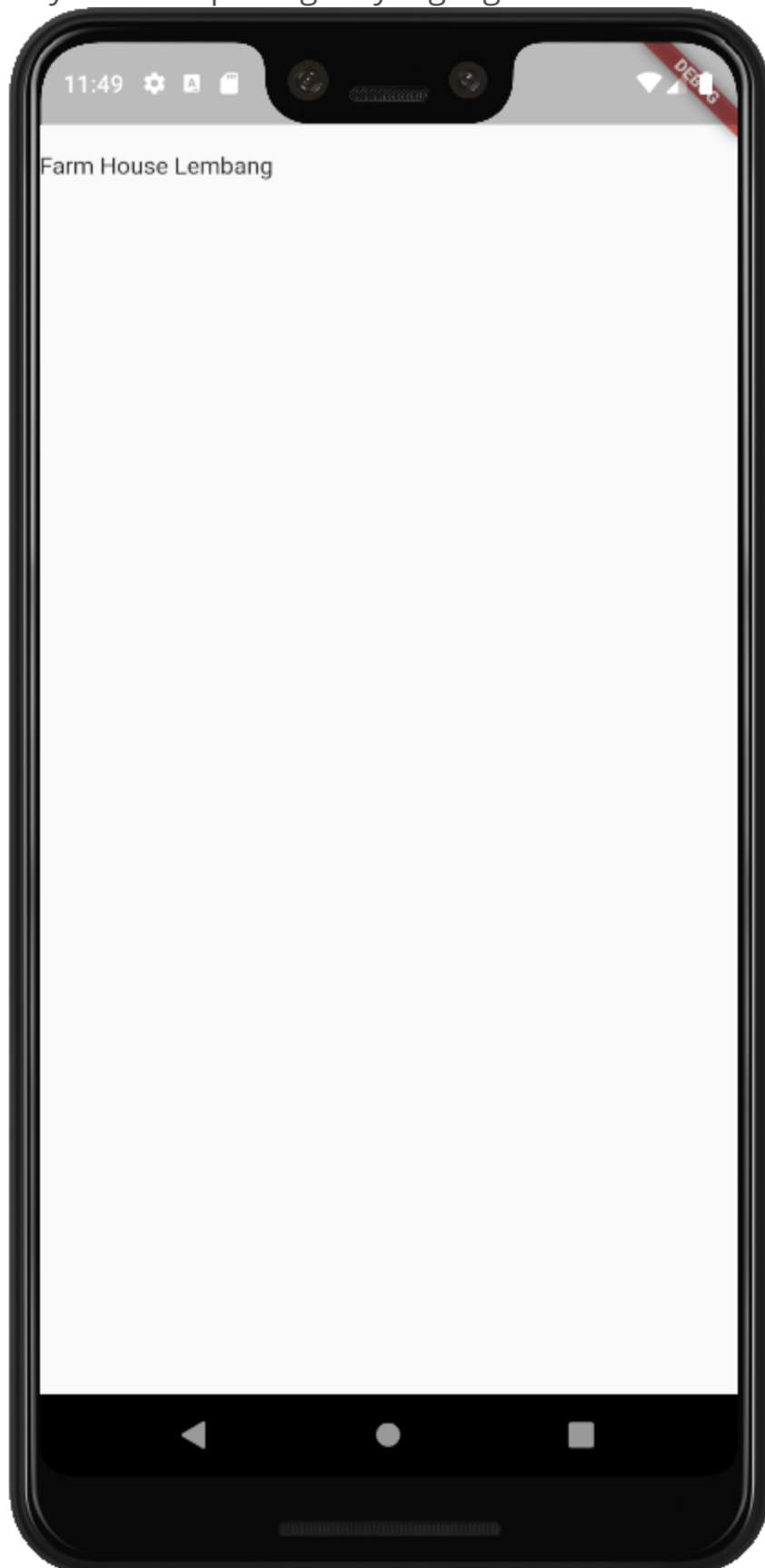
8. Jika Anda kesulitan menentukan margin atas, khususnya pada perangkat yang memiliki notch yang umumnya memiliki status bar yang lebih besar, Anda dapat memanfaatkan widget `SafeArea`.

```
1. class DetailScreen extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return Scaffold(  
5.       body: SafeArea(  
6.         child: Column(  

```

```
7.         children: <Widget>[
8.           Container(
9.             margin: EdgeInsets.only(top: 16.0),
10.            child: Text('Farm House Lembang'),
11.          )
12.        ],
13.      ),
14.    ),
15.  );
16. }
17. }
```

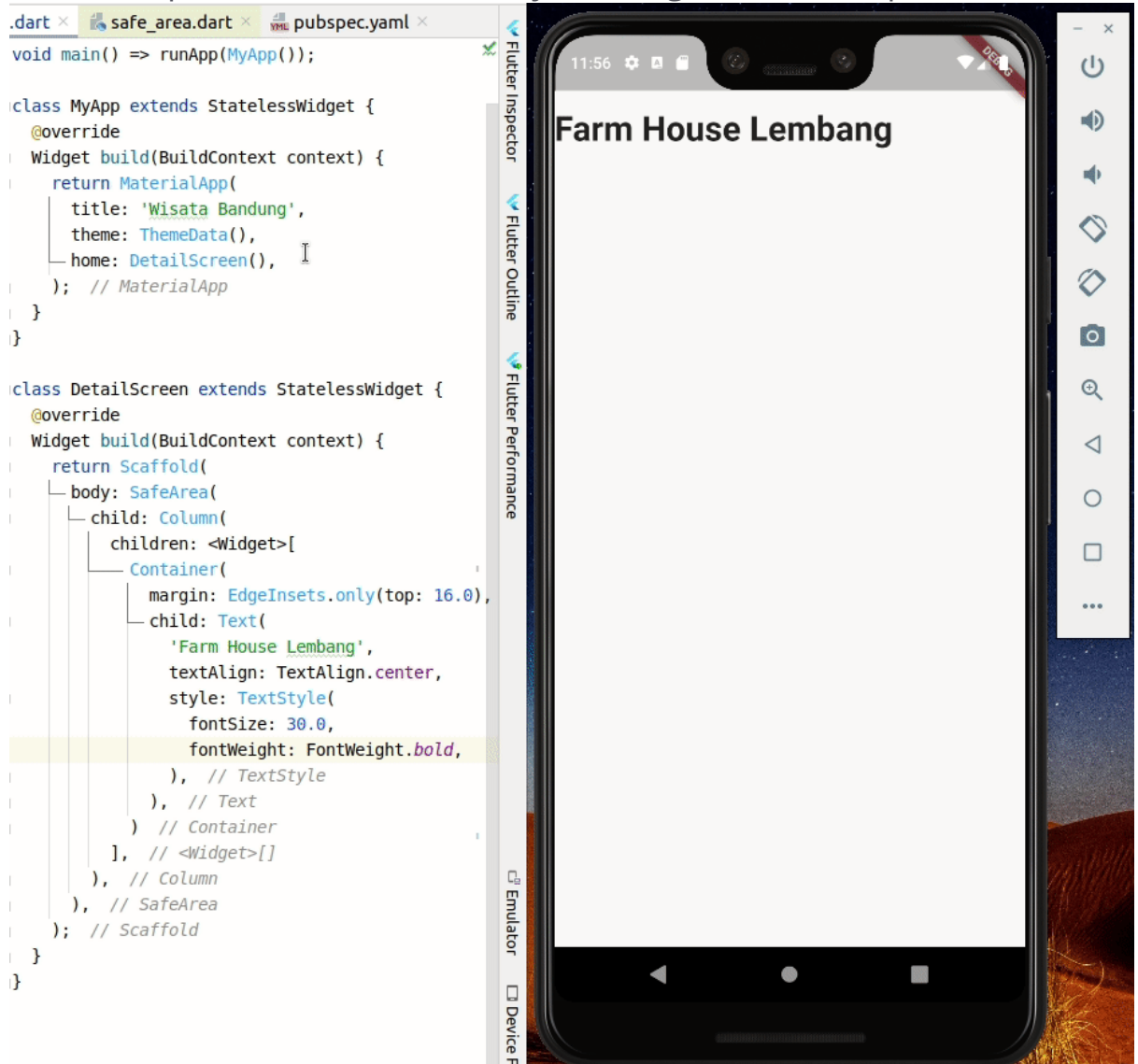
Widget ini akan memberikan padding yang secara otomatis menyesuaikan perangkat yang digunakan.



9. Selanjutnya, sesuai contoh kita akan membuat teks judul berada di tengah. Tambahkan parameter atau properti `textAlign` pada widget `Text`.

```
1. Container(  
2.   margin: EdgeInsets.only(top: 16.0),  
3.   child: Text(  
4.     'Farm House Lembang',  
5.     textAlign: TextAlign.center,  
6.     style: TextStyle(  
7.       fontSize: 30.0,  
8.       fontWeight: FontWeight.bold,  
9.     ),  
10.  ),  
11. ),
```

Lakukan hot reload. Tidak ada perubahan, apa sebabnya? Jika menggunakan Android Studio Anda dapat memanfaatkan fitur Flutter Inspector untuk melihat layout widget di dalam aplikasi.



Dari gambar di atas bisa kita lihat ternyata layout aplikasi kita tidak penuh hingga seluruh halaman. Ini disebabkan sisi horizontal dari Column hanya menyesuaikan dengan konten yang ada di dalamnya. Untuk memaksimalkan ukuran lebar dari Column, tambahkan kode berikut:

```

12.   body: SafeArea(
13.     child: Column(
14.       crossAxisAlignment: CrossAxisAlignment.stretch,
15.       children: <Widget>[
16.         Container(
17.           margin: EdgeInsets.only(top: 16.0),
18.           child: Text(
19.             'Farm House Lembang',
20.             textAlign: TextAlign.center,
21.             style: TextStyle(
22.               fontSize: 30.0,
23.               fontWeight: FontWeight.bold,
24.             ),
25.           ),
26.         ),
27.       ],
28.     ),
29.   ),

```

10. Setelah menyelesaikan judul, selanjutnya kita akan membuat bagian kedua yaitu informasi dari tempat wisata.



Seperti yang terlihat

kita perlu menyusun widget secara horizontal dan vertikal. Mari tambahkan child kedua dari Column dengan sebuah Container berisi Row. Tambahkan juga margin pada sisi atas dan bawah untuk memberikan jarak antar widget.

```

1. class DetailScreen extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold(
5.       body: SafeArea(
6.         child: Column(
7.           crossAxisAlignment: CrossAxisAlignment.stretch,
8.           children: <Widget>[
9.             Container(...),
10.            Container(
11.              margin: EdgeInsets.symmetric(vertical: 16.0),
12.              child: Row(
13.                children: <Widget>[],
14.              ),
15.            ),

```

```

16.         ],
17.     ),
18. ),
19. );
20. }
21. }

```

11. Buat widget Column untuk menyusun Icon dan Text.

```

1. class DetailScreen extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold(
5.       body: SafeArea(
6.         child: Column(
7.           crossAxisAlignment: CrossAxisAlignment.stretch,
8.           children: <Widget>[
9.             Container(...),
10.            Container(
11.              margin: EdgeInsets.symmetric(vertical: 16.0),
12.              child: Row(
13.                children: <Widget>[
14.                  Column(
15.                    children: <Widget>[
16.                      Icon(Icons.calendar_today),
17.                      Text('Open Everyday'),
18.                    ],
19.                  ),
20.                ],
21.              ),
22.            ),
23.          ],
24.        ),
25.      ),
26.    );
27.  }
28. }

```

Jika Anda merasa jarak antara Icon dan Text terlalu rapat, Anda dapat menambahkan widget `SizedBox` untuk membuat “kotak” yang berguna untuk memberikan jarak.

```

29.   Column(
30.     children: <Widget>[
31.       Icon(Icons.calendar_today),
32.       SizedBox(height: 8.0),
33.       Text('Open Everyday'),
34.     ],
35.   ),

```

12. Selanjutnya sebagai tantangan, lengkapi informasi tempat wisata dengan pasangan ikon dan teks sesuai contoh

yang diberikan.



13. Untuk menyusun Row seperti di atas, pastikan menggunakan `mainAxisAlignment` seperti ini:

```
1. class DetailScreen extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold(
5.       body: SafeArea(
6.         child: Column(
7.           crossAxisAlignment: CrossAxisAlignment.stretch,
8.           children: <Widget>[
9.             Container(...),
10.            Container(
11.              margin: EdgeInsets.symmetric(vertical: 16.0),
12.              child: Row(
13.                mainAxisAlignment: MainAxisAlignment.spaceEvenly,
14.                children: <Widget>[
15.                  Column(...),
16.                  Column(...),
17.                  Column(...)
18.                ],
19.              ),
20.            ),
21.          ],
22.        ),
23.      ),
24.    );
25.  }
26. }
```

14. Pada langkah ini harusnya Anda sudah bisa menampilkan teks deskripsi sesuai langkah yang diberikan sebelumnya. Anda cukup menambahkan widget `Container` dan `Text` untuk menampilkan konten deskripsi. Anda juga dapat menambahkan style sesuai selera Anda.

```
1. Container(
2.   padding: EdgeInsets.all(16.0),
3.   child: Text(
4.     'Berada di jalur utama Bandung-Lembang, Farm House menjadi objek wisata yang tidak pernah sepi pengunjung. Selain karena letaknya strategis, kawasan ini juga menghadirkan nuansa wisata khas Eropa. Semua itu diterapkan dalam bentuk spot swafoto Instagramable.',
5.     textAlign: TextAlign.center,
6.     style: TextStyle(fontSize: 16.0),
7.   ),
```

15. Keseluruhan kode Anda akan seperti berikut:

```
8. ),
1. import 'package:flutter/material.dart';
2.
3. void main() => runApp(MyApp());
4.
5. class MyApp extends StatelessWidget {
6.   @override
7.   Widget build(BuildContext context) {
8.     return MaterialApp(
9.       title: 'Wisata Bandung',
10.      theme: ThemeData(),
11.      home: DetailScreen(),
12.    );
13.  }
14. }
15.
16. class DetailScreen extends StatelessWidget {
17.   @override
18.   Widget build(BuildContext context) {
19.     return Scaffold(
20.       body: SafeArea(
21.         child: Column(
22.           crossAxisAlignment: CrossAxisAlignment.stretch,
23.           children: <Widget>[
24.             Container(
25.               margin: EdgeInsets.only(top: 16.0),
26.               child: Text(
27.                 'Farm House Lembang',
28.                 textAlign: TextAlign.center,
29.                 style: TextStyle(
30.                   fontSize: 30.0,
31.                   fontWeight: FontWeight.bold,
32.                 ),
33.             ),
34.             ),
35.             Container(
36.               margin: EdgeInsets.symmetric(vertical: 16.0),
37.               child: Row(
38.                 mainAxisAlignment: MainAxisAlignment.spaceEvenly,
39.                 children: <Widget>[
40.                   Column(
41.                     children: <Widget>[
42.                       Icon(Icons.calendar_today),
43.                       SizedBox(height: 8.0),
44.                       Text('Open Everyday'),
45.                     ],
46.                   ),
47.                   Column(
48.                     children: <Widget>[
49.                       Icon(Icons.access_time),
50.                       SizedBox(height: 8.0),
```



```

51.             Text('09:00 - 20:00'),
52.         ],
53.     ),
54.     Column(
55.         children: <Widget>[
56.             Icon(Icons.monetization_on),
57.             SizedBox(height: 8.0),
58.             Text('RP 25.000'),
59.         ],
60.     ),
61. ],
62. ),
63. ),
64. Container(
65.     padding: EdgeInsets.all(16.0),
66.     child: Text(
67.         'Berada di jalur utama Bandung-Lembang, Far
        m House menjadi objek wisata yang tidak pernah sepi pengunjung
        . Selain karena letaknya strategis, kawasan ini juga menghadirkan
        nuansa wisata khas Eropa. Semua itu diterapkan dalam bentuk spot
        swafoto Instagramable.',
68.         textAlign: TextAlign.center,
69.         style: TextStyle(fontSize: 16.0),
70.     ),
71. )
72. ],
73. ),
74. ),
75. );
76. }
77. }

```

Anda juga dapat mengunduh keseluruhan kodenya pada tautan berikut: <https://github.com/dicodingacademy/a159-flutter-pemula-labs>