

Modul 11

Komponen dalam mendesain Aplikasi

Menus, Lists, dan Dividers

Module Overview

Mengenal dan implementasi widget Menu, Lists dan Divider pada Flutter. Widget ini digunakan menampilkan list item pada menggunakan ListView, PopupMenuButton dan Divider sebagai garis pemisah antara item.

Module Objectives

Setelah mempelajari dan mempraktikkan modul ini, mahasiswa diharapkan dapat:

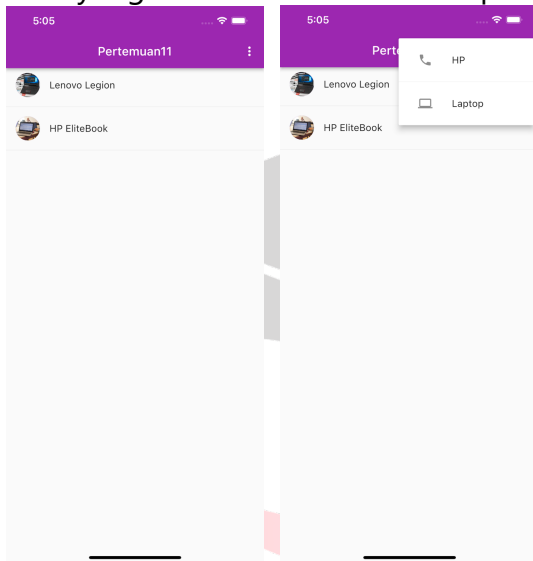
- Mengetahui kegunaan dari widget PopupMenuButton, ListView dan Divider.
- Mampu menerapkan ketiga widget ini pada aplikasi flutter.

Baners, Dialog dan Snackbars

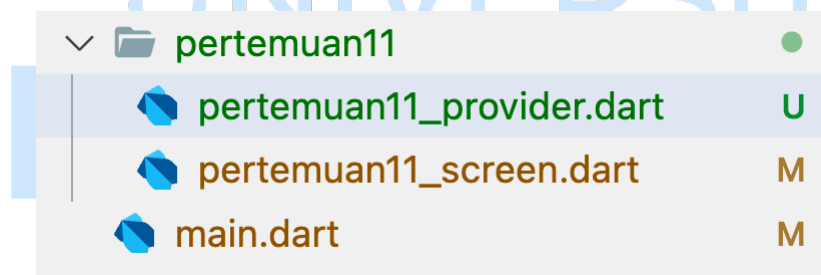
Menerapkan Menu, List dan Divider

Keterangan:

ListTile akan menampilkan item Laptop atau Smartphone dari row data (json) pada sebuah ListView (scrollable). Menu action AppBar menerapkan PopupMenu dengan 2 item yang clickable untuk menampilkan pilihan data yang dipilih pengguna.



1. Berikut adalah struktur folder untuk pertemuan 11.



Keterangan:

- pertemuan11_screen.dart digunakan untuk screen UI flutter (root home)
- pertemuan11_provider.dart digunakan provide logic pada aplikasi.

2. Buatlah screen baru dengan nama file **pertemuan11_screen.dart** pada folder pertemuan11. Import **material**, Gunakan **StatefulWidget**, namai class

statefulWidget dengan **Pertemuan10Screen** dan gunakan **Scaffold** untuk memulai mendesain komponen.

```

1  import 'package:flutter/material.dart';
2
3  class Pertemuan11Screen extends StatefulWidget {
4    const Pertemuan11Screen({Key? key}) : super(key: key);
5
6    @override
7    State<Pertemuan11Screen> createState() => _Pertemuan11ScreenState();
8  }
9
10 class _Pertemuan11ScreenState extends State<Pertemuan11Screen> {
11   @override
12   Widget build(BuildContext context) {
13     return Scaffold(
14       appBar: AppBar(
15         title: const Text('Pertemuan11'),
16       ),
17       body: Text('Pertemuan 11'),
18     );
19   }

```

Keterangan:

- Update **main.dart** menjadi "home: Pertemuan11Screen()"
- Pada terminal root folder ketik "flutter run" untuk menjalankan aplikasi.

3. Buatlah sebuah file dengan nama **pertemuan11_provider.dart** untuk menyediakan logic proses nantinya. Nama dengan class **Pertemuan11Provider** seperti gambar dibawah.

```

1  import 'package:flutter/material.dart';
2
3  class Pertemuan11Provider extends ChangeNotifier {
4  }

```

Kemudian, tambahkan pada multiProvider di main.dart seperti gambar berikut:



```

1  void main() {
2    runApp(MultiProvider(providers: [
3      //Provider minggu11
4      ChangeNotifierProvider(create: (_) => Pertemuan11Provider()),
5    ], child: const MyApp()));
6  }

```

****Pastikan import Pertemuan11Provider telah dilakukan.**

4. Membuat raw data dan logic pada **Pertemuan11Provider**.

Buatlah data dengan format json seperti dibawah (copy).

```

final hp = {
  "data": [
    {
      "model": "Samsung Galaxy",
      "img":
        "https://upload.wikimedia.org/wikipedia/commons/thumb/7/71/I7500_with_screen_protector.jpg/220px-I7500_with_screen_protector.jpg",
      "developer": "Samsung Electronics",
      "price": 2500000,
      "rating": 4.5
    },
    {
      "model": "Sony Xperia Z",
      "img":
        "https://upload.wikimedia.org/wikipedia/commons/thumb/8/84/Sony_Xperia_Z.JPG/200px-Sony_Xperia_Z.JPG",
      "developer": "Sony Mobile",
      "price": 1500000,
      "rating": 4.1
    }
  ]
};

```

```
final laptop = {  
  "data": [  
    {  
      "model": "Lenovo Legion",  
      "img":  
  
      "https://upload.wikimedia.org/wikipedia/commons/thumb/8/8e/Lenovo_Legion_Y520_%281%29.jpg/220px-Lenovo_Legion_Y520_%281%29.jpg",  
      "developer": "Lenovo",  
      "price": 12500000,  
      "rating": 4  
    },  
    {  
      "model": "HP EliteBook",  
      "img":  
      "https://upload.wikimedia.org/wikipedia/commons/thumb/a/ad/HP_Elitebook_820_G4.png/250px-HP_Elitebook_820_G4.png",  
      "developer": "HP",  
      "price": 2500000,  
      "rating": 4.8  
    },  
  ]  
};
```



UNIVERSITAS MIKROSKIL

Paste di dalam class Pertemuan11Provider, seperti gambar berikut:

```
1 import 'dart:convert';
2 import 'package:flutter/material.dart';
3
4 class Pertemuan11Provider extends ChangeNotifier {
5   initData() async {
6     setData = hp;
7   }
8
9   final hp = {
10     "data": [
11       {
12         "model": "Samsung Galaxy",
13         "img":
14           "https://upload.wikimedia.org/wikipedia/commons/thumb/7/71/I7500_with_screen_protector.jpg/220px-I7500_with_screen_protector.jpg",
15         "developer": "Samsung Electronics",
16         "price": 2500000,
17         "rating": 4.5
18       },
19       {
20         "model": "Sony Xperia Z",
21         "img":
22           "https://upload.wikimedia.org/wikipedia/commons/thumb/8/84/Sony_Xperia_Z.JPG/200px-Sony_Xperia_Z.JPG",
23         "developer": "Sony Mobile",
24         "price": 1500000,
25         "rating": 4.1
26       },
27     ]
28   };
29
30   final laptop = {
31     "data": [
32       {
33         "model": "Lenovo Legion",
34         "img":
35           "https://upload.wikimedia.org/wikipedia/commons/thumb/8/8e/Lenovo_Legion_Y520_(1).jpg/220px-Lenovo_Legion_Y520_(1).jpg",
36         "developer": "Lenovo",
37         "price": 12500000,
38         "rating": 4
39       },
40       {
41         "model": "HP EliteBook",
42         "img":
43           "https://upload.wikimedia.org/wikipedia/commons/thumb/a/ad/HP_Elitebook_820_G4.png/250px-HP_Elitebook_820_G4.png",
44         "developer": "HP",
45         "price": 2500000,
46         "rating": 4.8
47       },
48     ]
49   };
50
51   dynamic _data;
52   dynamic get data => _data;
53   set setData(val) {
54     var tmp = json.encode(val);
55     _data = json.decode(tmp);
56
57     notifyListeners();
58   }
59
60   ubahList(val) {
61     if (val == 'hp') {
62       setData = hp;
63     } else {
64       setData = laptop;
65     }
66   }
67 }
68
```

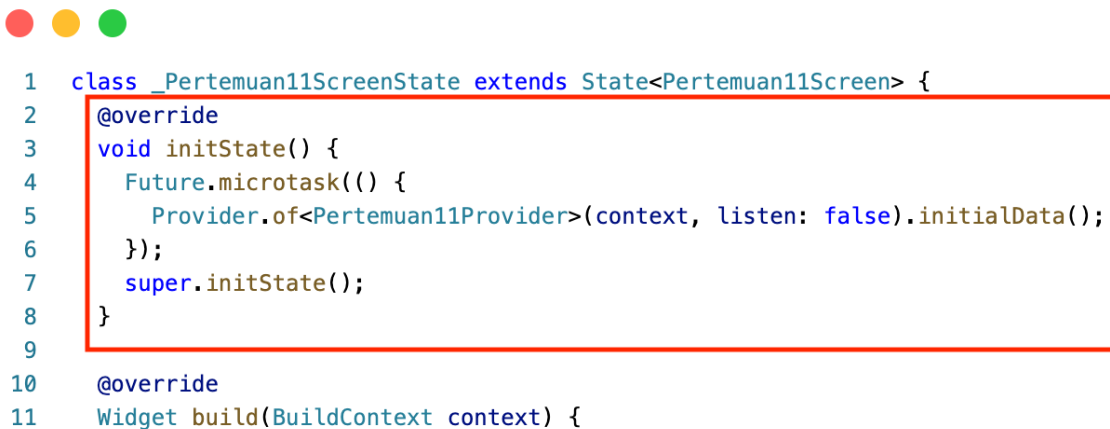
Keterangan:

- `initData()` pada baris 5 - 7 adalah sebuah fungsi yang nantinya akan dipanggil pertama kali saat aplikasi di jalan (`initialState`) untuk `setData` awal yaitu **hp**.

- Baris 9 - 49 adalah data list objek untuk data HP dan Laptop untuk nanti digunakan sebagai data item pada list.
- Baris 51: **dynamic _data;** sebuah local variable untuk menampung data hp / laptop yang dipilih user.
- Baris 52: **dynamic get data => _data;** sebuah method get yang mengembalikan nilai _data berisi list objek hp atau laptop.
- Baris 53 - 58, adalah sebuah method setter untuk set data ke dalam **_data**. **json.encode(val)** digunakan untuk convert list objek ke JSON String. kemudian **json.decode(tmp)** untuk parse String dan menghasilkan sebuah JSON Objek. Kemudian, **notifyListeners()** digunakan untuk notif pada state bawah ada perubahan.
- Baris 60 - 66: **ubahList(val)** adalah sebuah method yang digunakan untuk ubah data saat user pilih menu HP atau laptop pada PopupMenu.

5. Menampilkan item dalam **ListView** dan **ListTile**.

Tambah initState() sebelum Widget build, seperti gambar di bawah:



```

1 class _Pertemuan11ScreenState extends State<Pertemuan11Screen> {
2   @override
3   void initState() {
4     Future.microtask(() {
5       Provider.of<Pertemuan11Provider>(context, listen: false).initialData();
6     });
7     super.initState();
8   }
9
10  @override
11  Widget build(BuildContext context) {

```

Keterangan:

- **initState** akan jalan pertama sekali saat halaman ini diakses

- Karena inialisasi data ke `_data` butuh proses, maka gunakan fungsi `Future.microtask()` untuk `initialData`.

Tampilkan `ListView` pada `body`. Buatlah method `body()` di **bawah method `Widget build()`** seperti gambar di bawah:



```

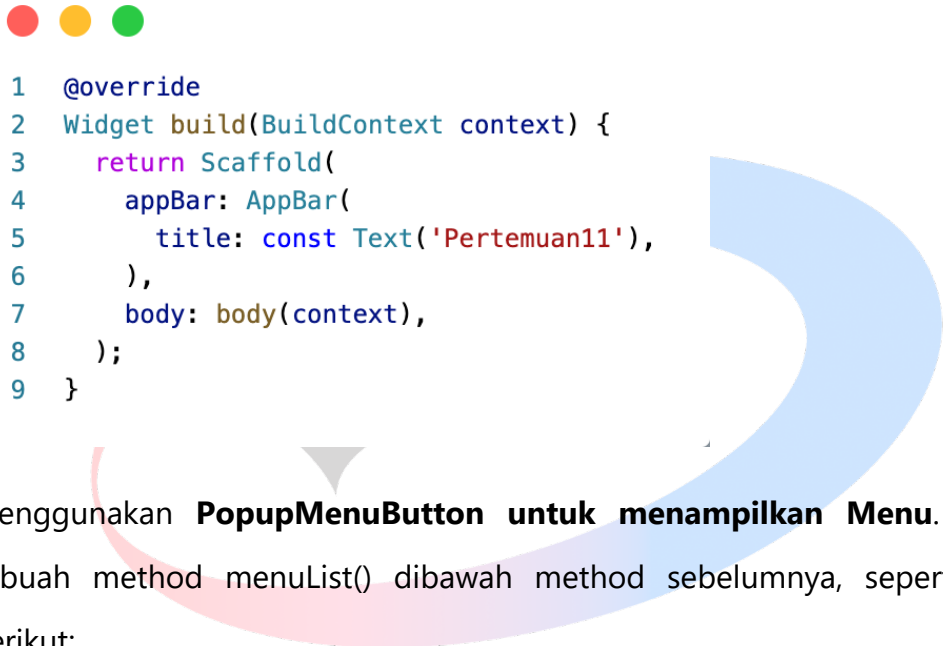
1  body(BuildContext context) {
2    final prov = Provider.of<Pertemuan11Provider>(context);
3    if (prov.data == null) {
4      return const CircularProgressIndicator();
5    } else {
6      return ListView(
7        children: List.generate(
8          prov.data['data']!.length,
9          (index) {
10             var item = prov.data['data']![index];
11             return Column(
12               children: [
13                 ListTile(
14                   leading:
15                     CircleAvatar(backgroundImage: NetworkImage(item['img'])),
16                   title: Text(item['model']),
17                 ),
18                 const Divider()
19               ],
20             );
21           },
22         ));
23   }
24 }
```

Keterangan:

- Baris 3 berfungsi untuk menghindari data null pada `listView`, jadi perlu dicek apakah `data == null`, karena data baru diinisialisasi saat `initialData` dimulai.
- Baris 10 digunakan untuk meringkas kode pemanggilan data di dalam array. Seharusnya setiap item dipanggil dengan: **`prov.data['data']![index]['model']`**. ***nama didalam kurung siku disesuaikan dengan key dari objek.*

- Baris 18 menampilkan penggunaan **Divider()** sebagai pemisah antara item.

Tampilkan method **body()** ini pada property body scaffold, seperti gambar berikut:



```
1  @override
2  Widget build(BuildContext context) {
3    return Scaffold(
4      appBar: AppBar(
5        title: const Text('Pertemuan11'),
6      ),
7      body: body(context),
8    );
9  }
```

6. Menggunakan **PopupMenuButton** untuk menampilkan **Menu**. Buatlah sebuah method `menuList()` dibawah method sebelumnya, seperti gambar berikut:

UNIVERSITAS
MIKROSKIL



```

1  menuList(BuildContext context) {
2    final prov = Provider.of<Pertemuan11Provider>(context);
3
4    return PopupMenuButton(
5      icon: const Icon(Icons.more_vert),
6      itemBuilder: (BuildContext context) {
7        return <PopupMenuEntry>[
8          PopupMenuItem(
9            child: ListTile(
10             onTap: () => prov.ubahList('hp'),
11             leading: const Icon(Icons.phone),
12             title: const Text('HP'),
13           ),
14         ),
15         const PopupMenuDivider(),
16         PopupMenuItem(
17           child: ListTile(
18             onTap: () => prov.ubahList('laptop'),
19             leading: const Icon(Icons.laptop),
20             title: const Text('Laptop'),
21           ),
22         ),
23       ];
24     },
25   );
26 }

```

Keterangan:

- Baris 7, akan return sebuah list dengan Type PopupMenuEntry dengan beberapa PopUpMenuItem(). Cara ini mirip jika kita return widget pada column yaitu **children: <Widget>[]**, artinya children pada column harus berupa Widget. atau contoh lain seperti <String>[], artinya ini adalah list dengan isian hanya String saja.
- Baris 10 dan 18 akan trigger fungsi ubahList pada Pertemuan11Provider.

Tampilkan **menuList()** pada actions AppBar, seperti gambar berikut:

```
1  @override
2  Widget build(BuildContext context) {
3    return Scaffold(
4      appBar: AppBar(
5        title: const Text('Pertemuan11'),
6        actions: [
7          menuList(context),
8        ],
9      ),
10     body: body(context),
11   );
12 }
```



Latihan Praktek

1. Lengkapi **ListTile()** untuk menampilkan ***developer, price dan rating*** pada masing-masing item.
2. Tambah item "Clear" pada Menu untuk bersihkan list pada screen. Jadi, saat "**Clear**" di klik, maka pada body muncul pesan "***data tidak ditemukan.***"



UNIVERSITAS
MIKROSKIL