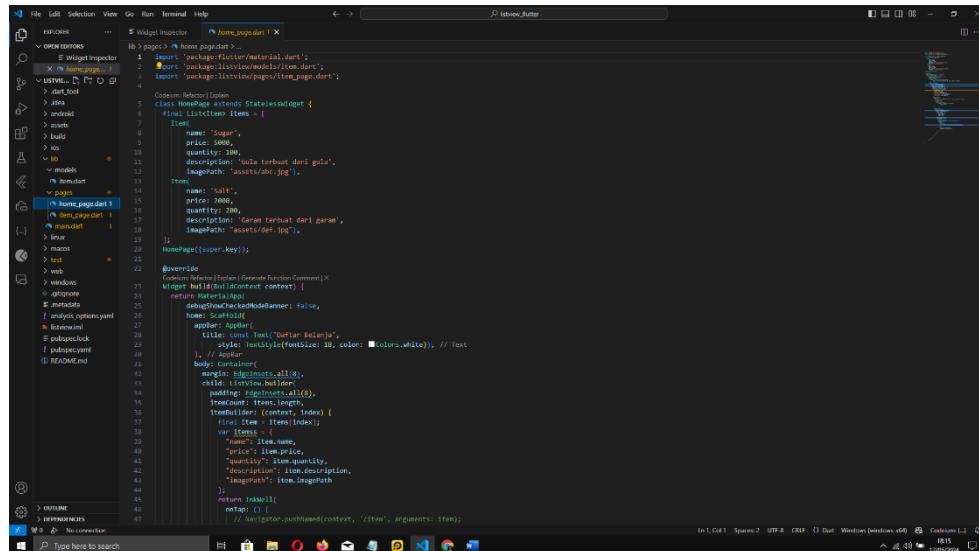


Listview Flutter

Nama : Moch Arif Lestiawan

Kelas : XI RPL 4 (18)

Link Github : <https://github.com/MochArif99/list-view-flutter>

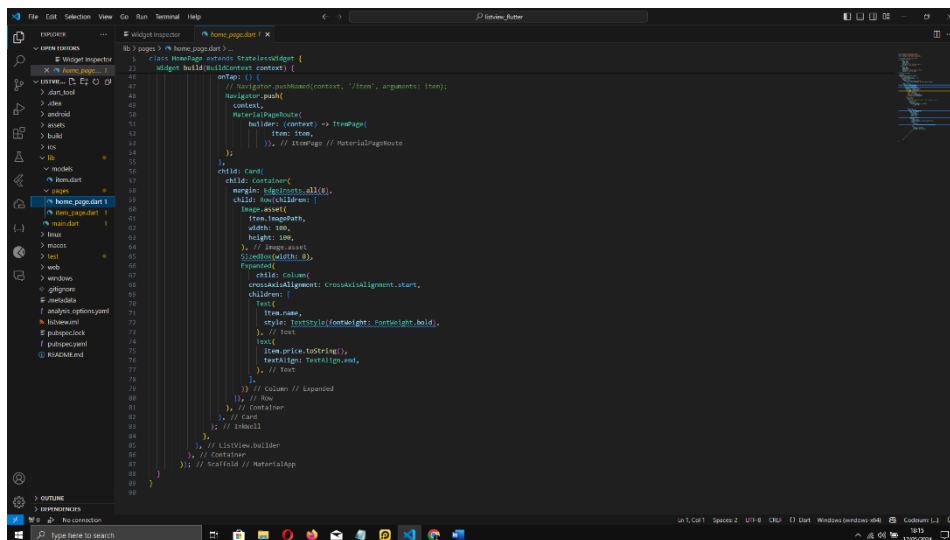


```
import 'package:flutter/material.dart';
import 'package:listview/models/item.dart';
import 'package:listview/pages/item_page.dart';

class HomePage extends StatelessWidget {
  final List<Item> items = [
    Item(
      name: 'Sugar',
      price: 1000,
      quantity: 200,
      description: 'suka terbuat dari gula',
      imagePath: 'assets/suc.jpg',
    ),
    Item(
      name: 'Salt',
      price: 1000,
      quantity: 200,
      description: 'suka terbuat dari garam',
      imagePath: 'assets/sat.jpg',
    ),
  ];

  HomePage({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('Listview Flutter'),
          style: TextStyle(fontSize: 18, color: Colors.white), // text
        ), // AppBar
        body: Container(
          margin: EdgeInsets.all(8),
          child: ListView.builder(
            padding: EdgeInsets.all(8),
            itemCount: items.length,
            itemBuilder: (context, index) {
              final item = items[index];
              var items = {
                'name': item.name,
                'price': item.price,
                'quantity': item.quantity,
                'description': item.description,
                'imagePath': item.imagePath,
              };
              return InkWell(
                onTap: () {
                  // Navigator.pushNamed(context, '/item', arguments: items);
                },
              );
            },
          ),
        ),
      ),
    );
  }
}
```



```
import 'package:flutter/material.dart';
import 'package:listview/models/item.dart';
import 'package:listview/pages/item_page.dart';

class ItemPage extends StatelessWidget {
  final Item item;

  ItemPage({super.key, required this.item});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Item Page'),
        style: TextStyle(fontSize: 18, color: Colors.white), // text
      ), // AppBar
      body: Container(
        margin: EdgeInsets.all(8),
        child: Card(
          margin: EdgeInsets.all(8),
          child: Row(
            children: [
              Image.asset(
                item.imagePath,
                width: 100,
                height: 100,
                fit: BoxFit.cover,
              ), // Image
              Expanded(
                child: Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text(
                      item.name,
                      style: TextStyle(fontWeight: FontWeight.bold),
                    ), // Text
                    Text(
                      item.price.toString(),
                      style: TextStyle(color: Colors.red),
                    ), // Text
                    Text(
                      item.quantity.toString(),
                      style: TextStyle(color: Colors.red),
                    ), // Text
                  ],
                ), // Column
              ), // Image
            ],
          ),
        ),
      ),
    );
  }
}
```

- Kelas HomePage adalah stateless widget yang bertanggung jawab untuk menampilkan daftar belanja.
- Variabel items: Berisi daftar item belanja yang ingin ditampilkan.
- Constructor: Menerima parameter opsional key.
- Deskripsi: Metode build() meng-override metode build() dari kelas StatelessWidget.

- Widget MaterialApp: Digunakan untuk membuat aplikasi Flutter. debugShowCheckedModeBanner disetel menjadi false untuk menyembunyikan banner debug di sudut kanan atas aplikasi.
- AppBar: Menampilkan judul halaman.
- Container: Digunakan untuk mengelompokkan elemen-elemen dalam layout.
- ListView.builder: Digunakan untuk menampilkan daftar item belanja dengan kemampuan pemilihan (scrollable).
- InkWell: Digunakan untuk menambahkan efek "tappable" pada item daftar.
- Widget Card: Membungkus setiap item dalam daftar dengan kartu.
- Widget Row: Digunakan untuk menampilkan item dalam satu baris.
- Widget Image.asset: Menampilkan gambar item dari path lokal.
- Widget Expanded: Mengisi sisa ruang dalam baris untuk memastikan teks berada di sebelah kanan gambar.
- Widget Column: Digunakan untuk menampilkan teks dengan format vertikal.
- Penggunaan Navigator.push: Ketika pengguna mengklik salah satu item, mereka akan dialihkan ke halaman detail item menggunakan MaterialPageRoute.

The screenshot shows a code editor with the following Dart code for `ItemPage`:

```
class ItemPage extends StatelessWidget {
  Widget build(BuildContext context) {
    child: ElevatedButton.icon(
      style: ButtonStyle(backgroundColor: MaterialStateProperty.resolveWith(states) => Colors.blue),
      onPressed: () {
        // Navigate back to the homePage
        Navigator.pop(context);
      },
      icon: Icon(Icons.shopping_cart_checkout),
      label: Text('Add to cart'),
    ), // ElevatedButton.icon
    // Expanded
    child: ElevatedButton(
      style: ButtonStyle(backgroundColor: MaterialStateProperty.resolveWith(states) => Colors.red),
      onPressed: () {
        // Navigate back to the homePage
        Navigator.pop(context);
      },
      child: Text('Back to Shopping list'),
    ), // ElevatedButton
  }, // Now
  children: [
    child: ElevatedButton.icon(
      style: ButtonStyle(backgroundColor: MaterialStateProperty.resolveWith(states) => Colors.green),
      onPressed: () {
        // Navigate back to the homePage
        Navigator.pop(context);
      },
      icon: Icon(Icons.notification_on),
      label: Text('Cancel'),
    ), // ElevatedButton.icon
    // Expanded
    // Now
  ], // Column
), // Container
), // Scaffold
} // ItemPage
```

The screenshot shows a code editor with the following Dart code for `Item`:

```
class ItemPage extends StatelessWidget {
  final Item item;
  ItemPage({Key? key, required this.item}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    var abc = item.price;
    var abc = item.quantity;
    // final ItemKey = GlobalKey<Item>? settings.arguments as Map;
    // print('ItemKey: $ItemKey');
    // print('Item: $item');
    // print('Item: $item');

    return Scaffold(
      appBar: AppBar(
        title: Text(item.name),
      ), // AppBar
      body: Container(
        padding: EdgeInsets.all(10),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          children: [
            child: Image.asset(
              item.imageUrl,
              width: 400,
              height: 400,
              fit: BoxFit.cover,
            ), // Image.asset
            // Gap
            Text(item.name, style: TextStyle(fontsize: 20, fontweight: FontWeight.bold)),
            Text('Price: $abc', style: TextStyle(fontsize: 24, fontweight: FontWeight.bold, color: Colors.black)),
            Text('Quantity: $abc', style: TextStyle(fontsize: 24, fontweight: FontWeight.bold)),
            Text('Description: $abc', style: TextStyle(fontsize: 16)),
            Text('Item description: $abc', style: TextStyle(fontsize: 16)),
            // Now
          ], // Column
        ), // Container
      ), // Scaffold
    ); // ItemPage
  } // ItemPage
}
```

- Kelas ItemPage adalah stateless widget yang bertanggung jawab untuk menampilkan detail dari suatu item.
- Properti item: Mewakili item yang akan ditampilkan.
- Constructor: Menerima parameter opsional key dan parameter wajib item.
- Metode build() meng-override metode build() dari kelas StatelessWidget.
- Widget Scaffold: Membungkus seluruh halaman aplikasi.
- Widget AppBar: Menampilkan judul halaman dengan nama item.
- Widget Container: Digunakan untuk mengelompokkan elemen-elemen dalam layout.
- Widget Column: Digunakan untuk menampilkan elemen secara vertikal.
- Widget Card: Membungkus gambar item untuk memberikan efek bayangan.
- Widget Image.asset: Menampilkan gambar item dari path lokal dengan dimensi tertentu dan tipe pemadanan (fit) yang ditentukan. (Tapi gak tampil di hasil nya gatau pak kenapa -_
- Widget Text: Menampilkan nama item, harga, jumlah tersedia, dan deskripsi item.
- Widget ElevatedButton: Digunakan untuk menambahkan tombol dengan latar belakang yang diberikan.
- Penggunaan Navigator.pop: Ketika pengguna menekan tombol, mereka akan dialihkan kembali ke halaman sebelumnya.

Fungsi add card & purchase tidak bisa, Cuma fungsi back pak yang bisa 😊

