Pemrograman Mobile

VS Code

GIT

Flutter https://docs.flutter.dev/

Andorid Studio

Postman

Chrome

Flutter

- framework untuk membuat aplikasi multi platform seperti Android, iOS, Web, Windows, Linux dan MacOS

Instalasi FLutter

https://flutter.dev/docs/get-started/install/windows

Ektract file ke direktori C:\flutter\ (Gunakan extract to)

Periksa apakah flutter telah benar-benar terinstall

Buka CMD ketikkan

```
flutter doctor
```

Instalasi FLutter

```
Doctor summary (to see all details, run flutter doctor -v):
[√] Flutter (Channel stable, 3.0.5, on Microsoft Windows
    [Version 10.0.19044.2130], locale en-ID)
[√] Android toolchain - develop for Android devices (Android SDK
    version 33.0.0)
[√] Chrome - develop for the web
[√] Visual Studio - develop for Windows (Visual Studio Community
    2022 17.2.5)
[√] Android Studio (version 2021.1)
[√] VS Code (version 1.73.1)
[√] Connected device (3 available)
[√] HTTP Host Availability
No issues found!
```

Instalasi FLutter

- **Instal SDK Android terbaru**
- **Instal Android Studio terbaru**

Installasi Flutter dan Dart di VSCode

Instal plugin Flutter dan Dart untuk VS Code

1. Buka VS Code.
2. Pilih extention
3. Cari flutter dan Dart
4. Instal
5. Mulai ulang VS Code

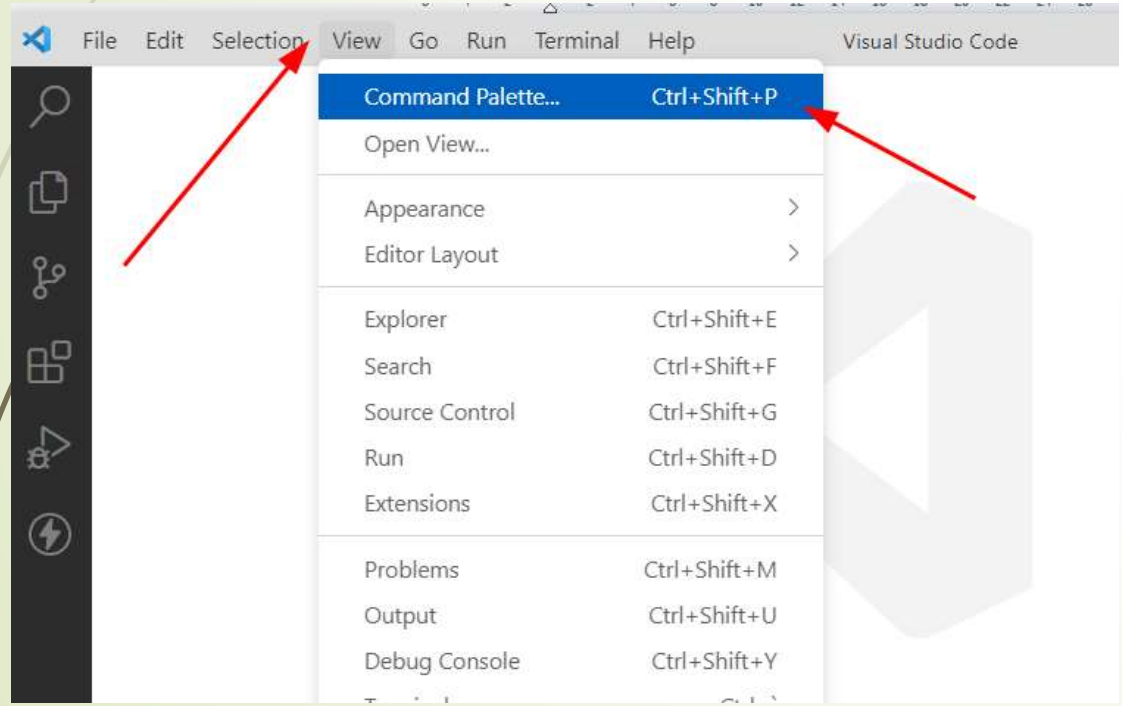
Installasi Flutter dan Dart di Android Studio

Instal plugin Flutter dan Dart untuk Android Studio

1. **Buka Android Studio.**
2. **Klik File → Pengaturan → Plugin.**
3. **Pilih plugin Flutter dan klik Instal.**
4. **Klik Ya saat diminta untuk menginstal plugin Dart.**
5. **Mulai ulang studio Android.**

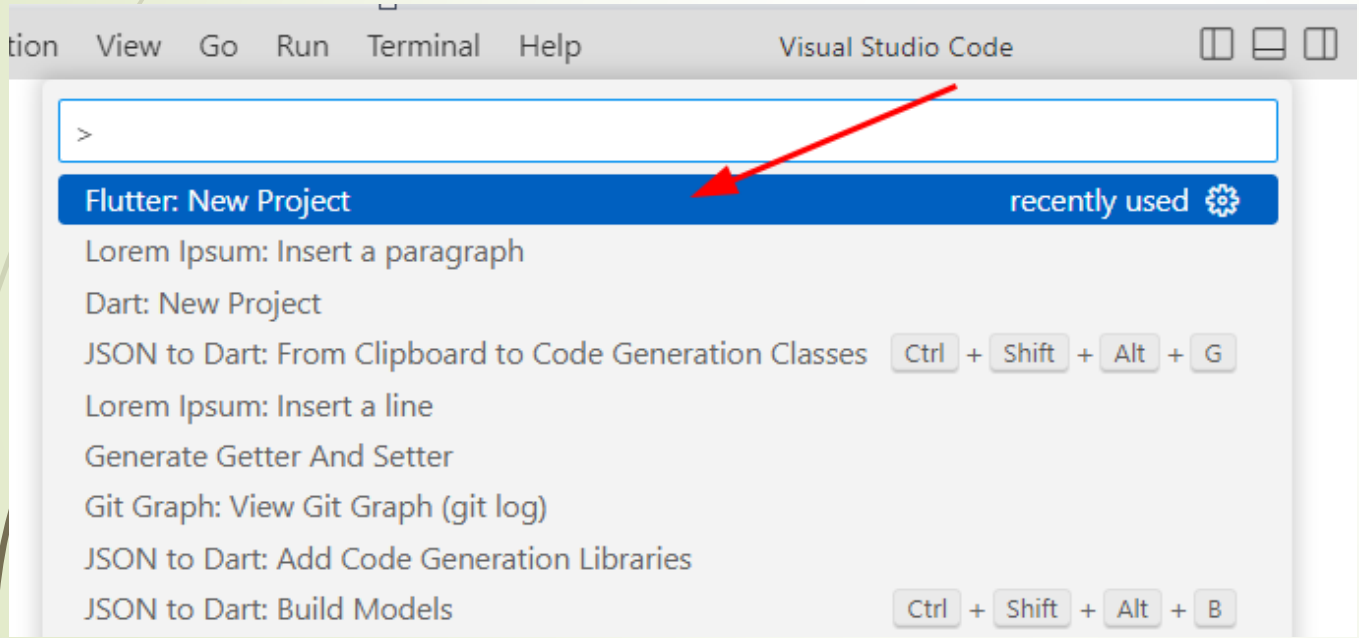
Arsitektur aplikasi flutter, hello world

- Buka VSCode
- Klik View->Command Pallete , atau shorcut di windows ctrl+shift+P



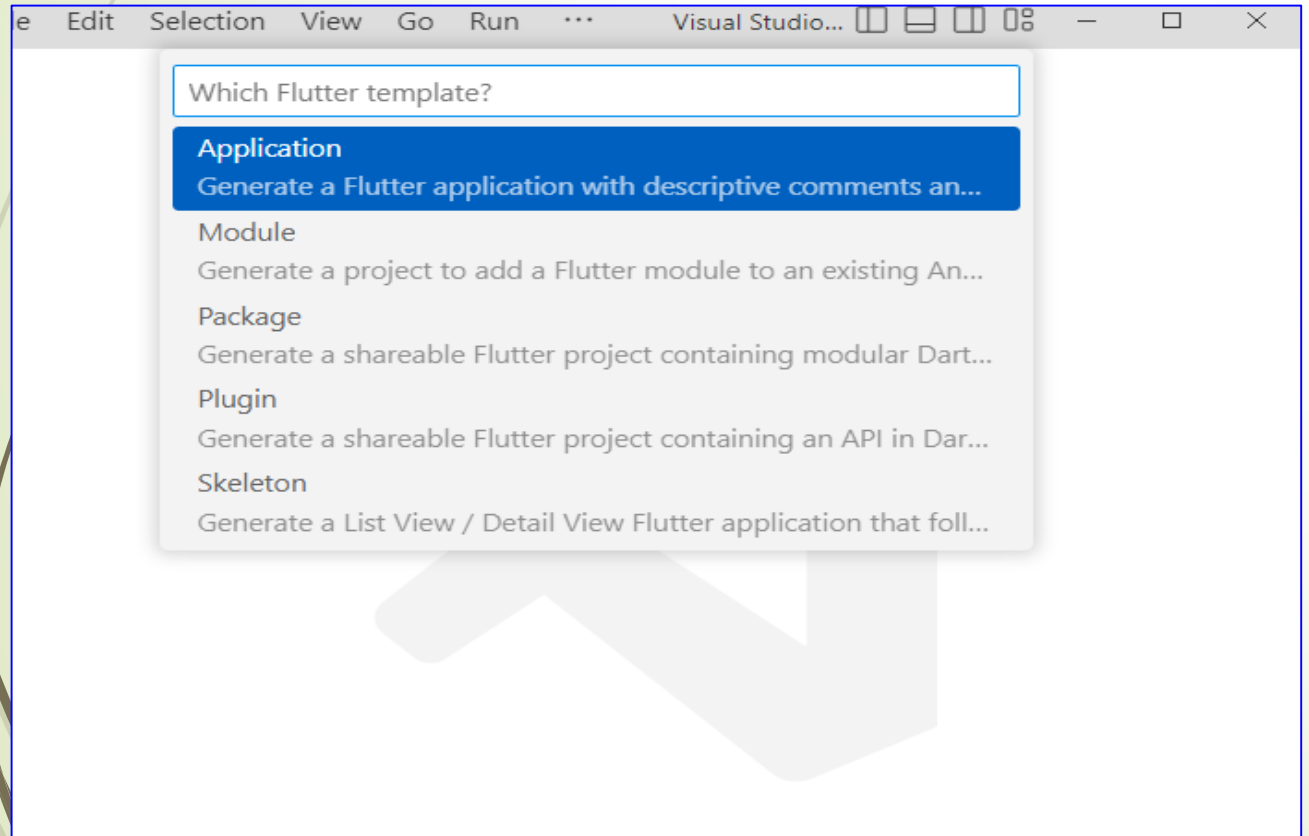
Arsitektur aplikasi flutter, hello world

- **Ketikkan Flutter pada kotak tersedia lalu pilih Flutter New Projek**

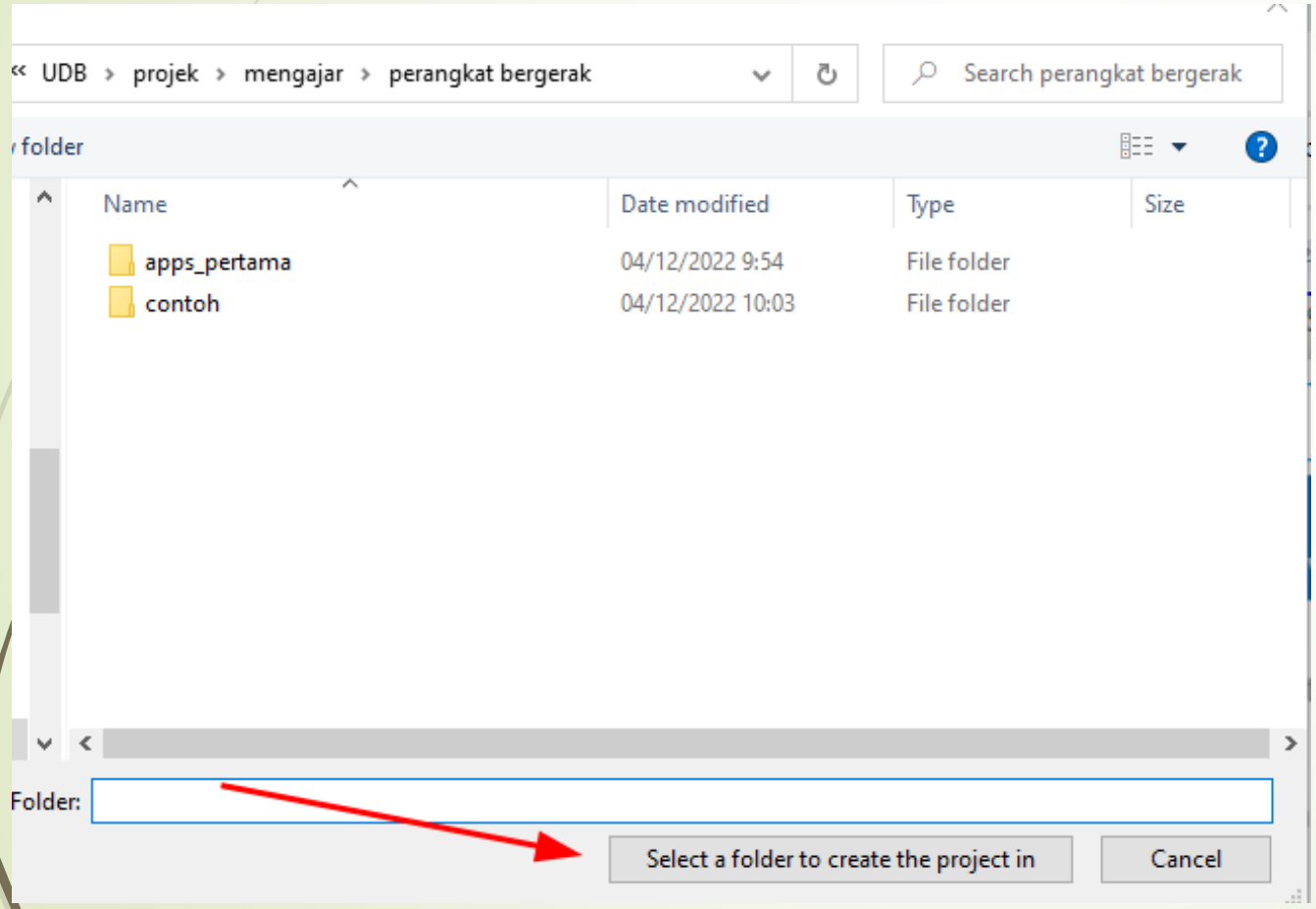


Arsitektur aplikasi flutter, hello world

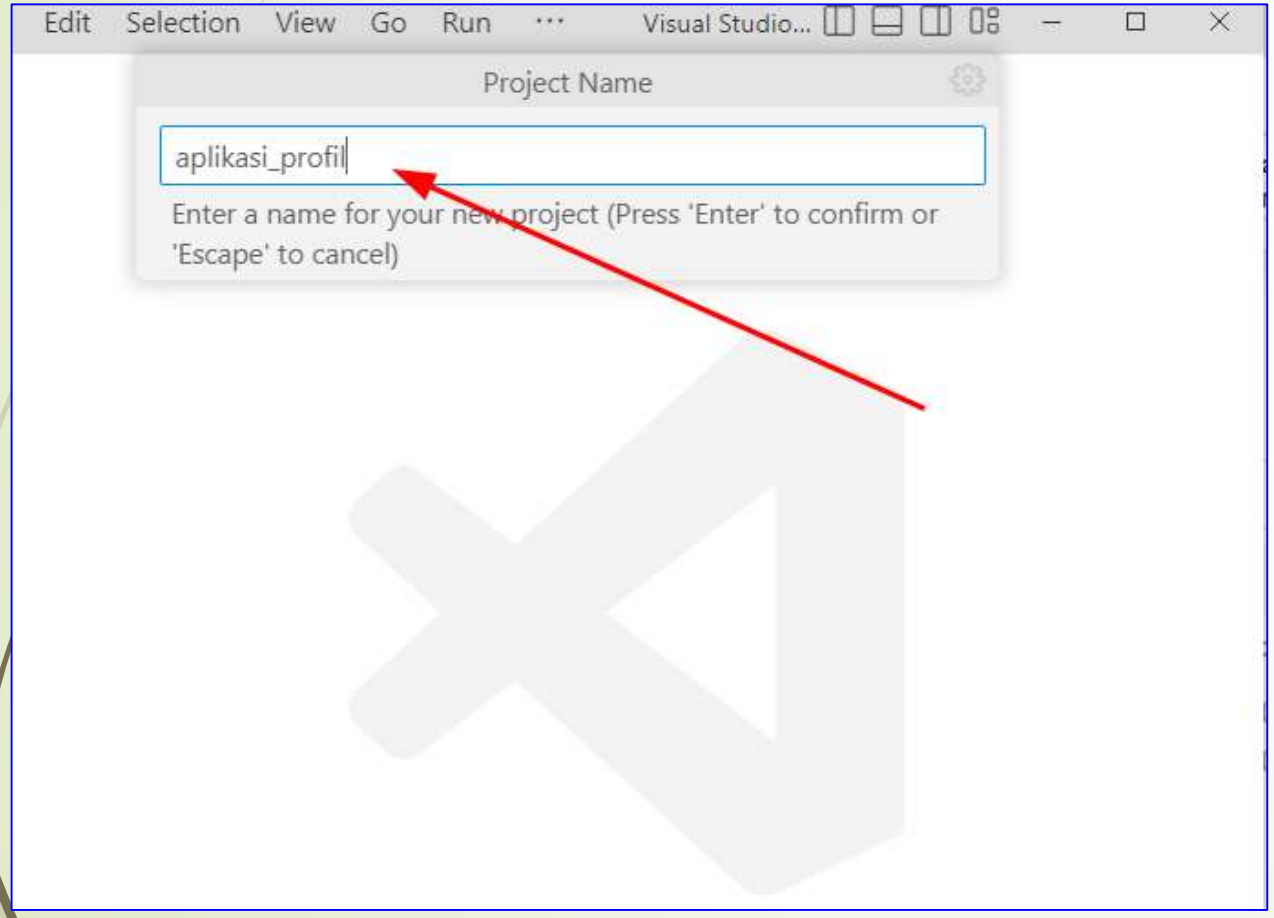
- **Ketikkan nama aplikasi ,tekan Enter , lalu pilih folder tempat projek akan disimpan**



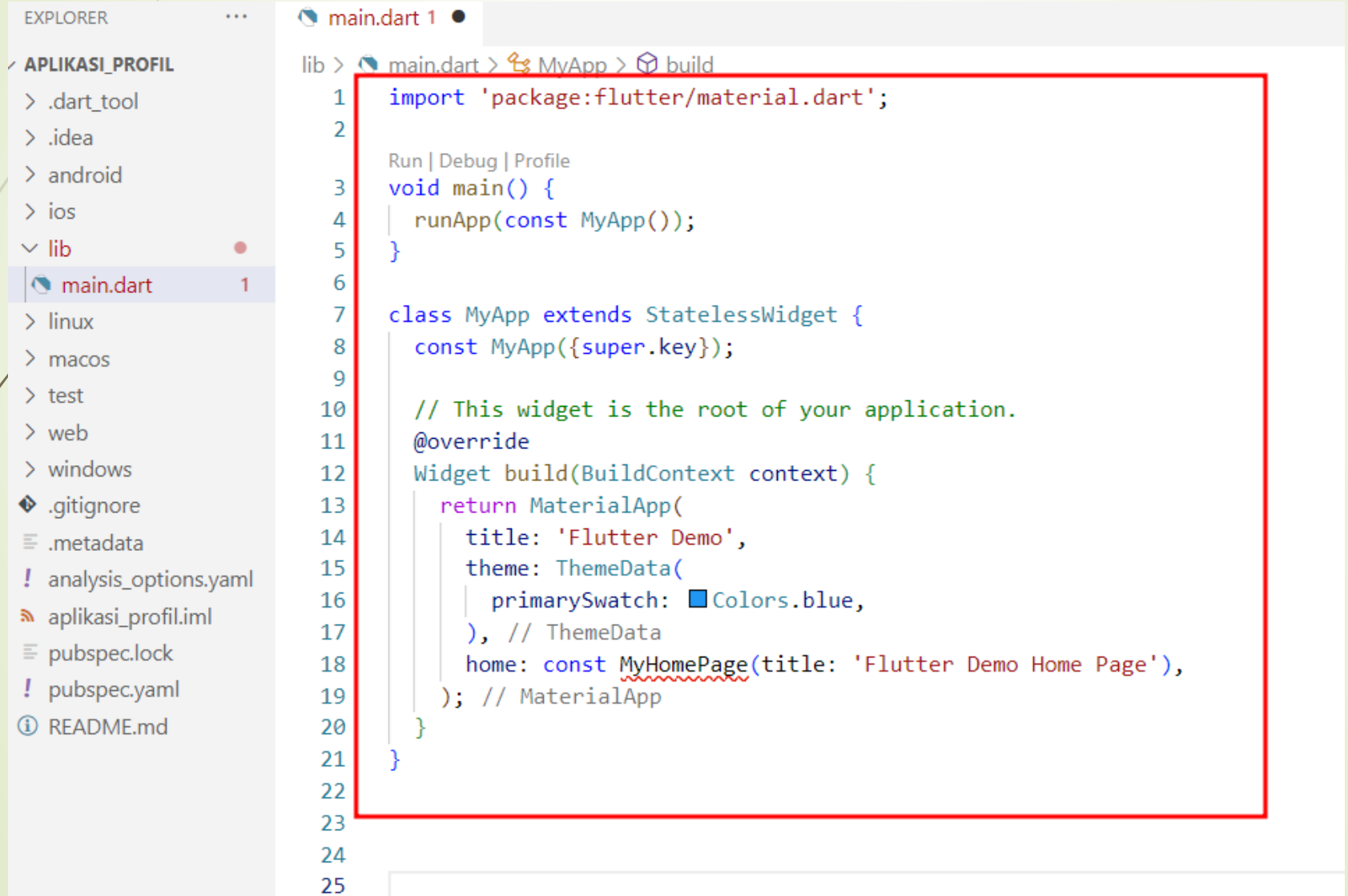
Arsitektur aplikasi flutter, hello world



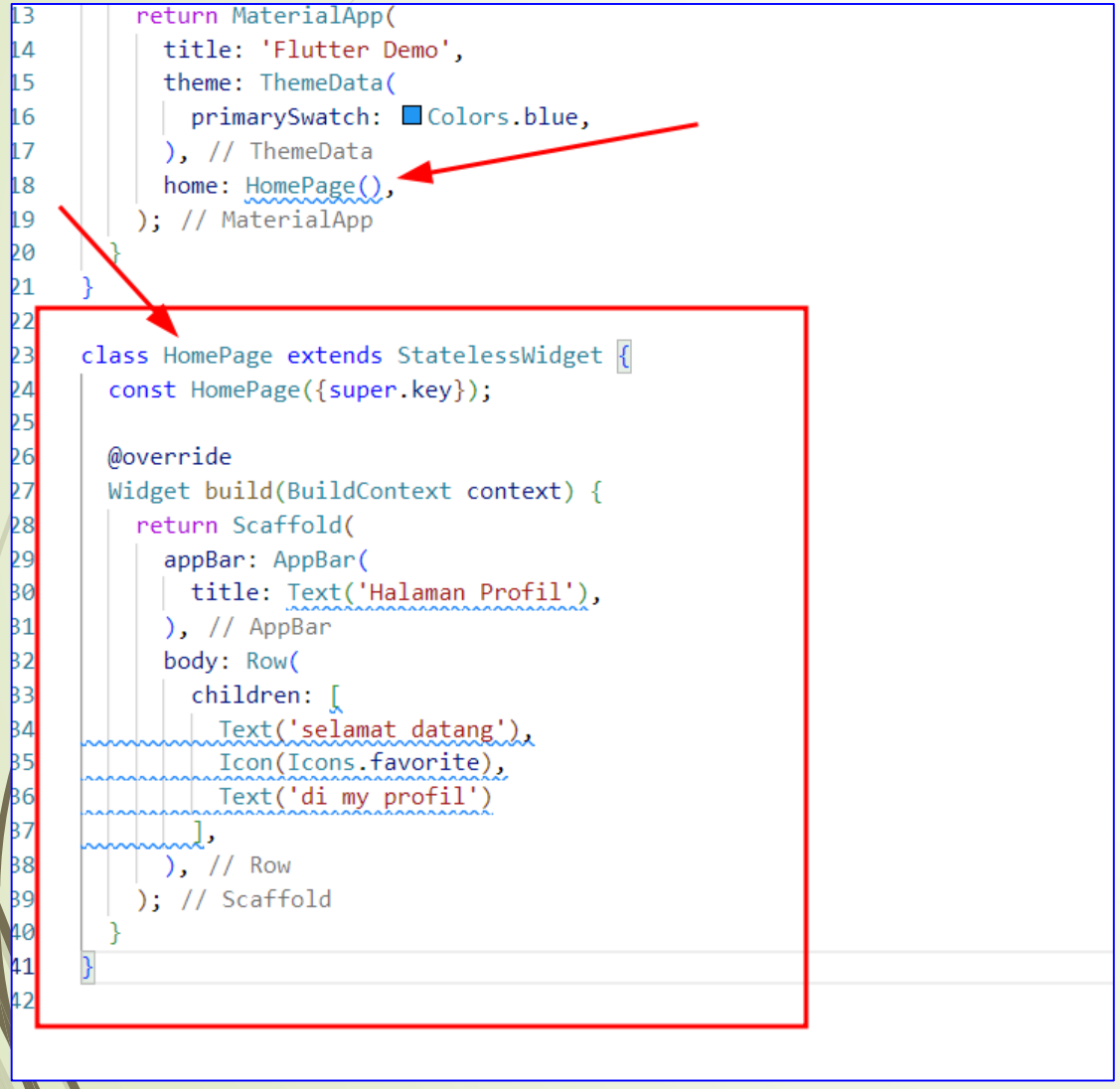
Arsitektur aplikasi flutter, hello world



Arsitektur aplikasi flutter, hello world

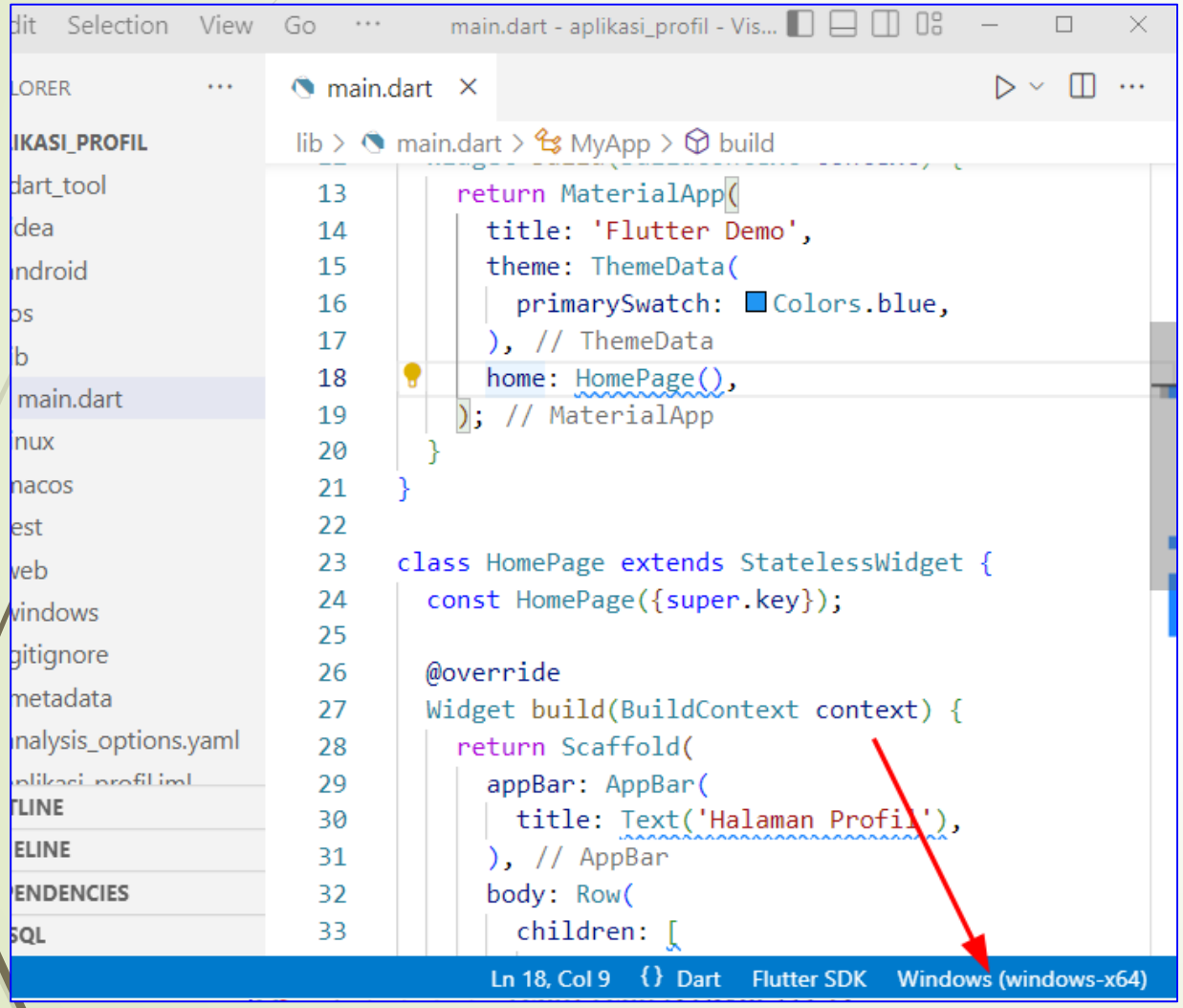


Arsitektur aplikasi flutter, hello world

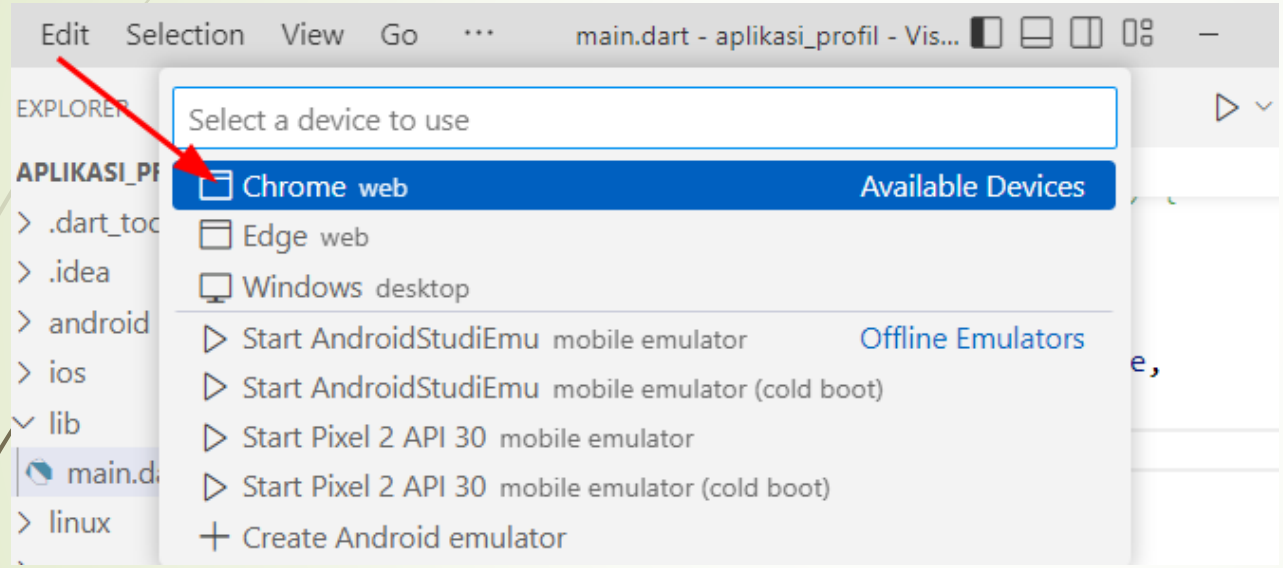


```
13      return MaterialApp(
14        title: 'Flutter Demo',
15        theme: ThemeData(
16          primarySwatch: Colors.blue,
17        ), // ThemeData
18        home: HomePage(),
19      ); // MaterialApp
20    }
21  }
22
23  class HomePage extends StatelessWidget {
24    const HomePage({super.key});
25
26    @override
27    Widget build(BuildContext context) {
28      return Scaffold(
29        appBar: AppBar(
30          title: Text('Halaman Profil'),
31        ), // AppBar
32        body: Row(
33          children: [
34            Text('selamat datang'),
35            Icon(Icons.favorite),
36            Text('di my profil')
37          ],
38        ), // Row
39      ); // Scaffold
40    }
41  }
42
```

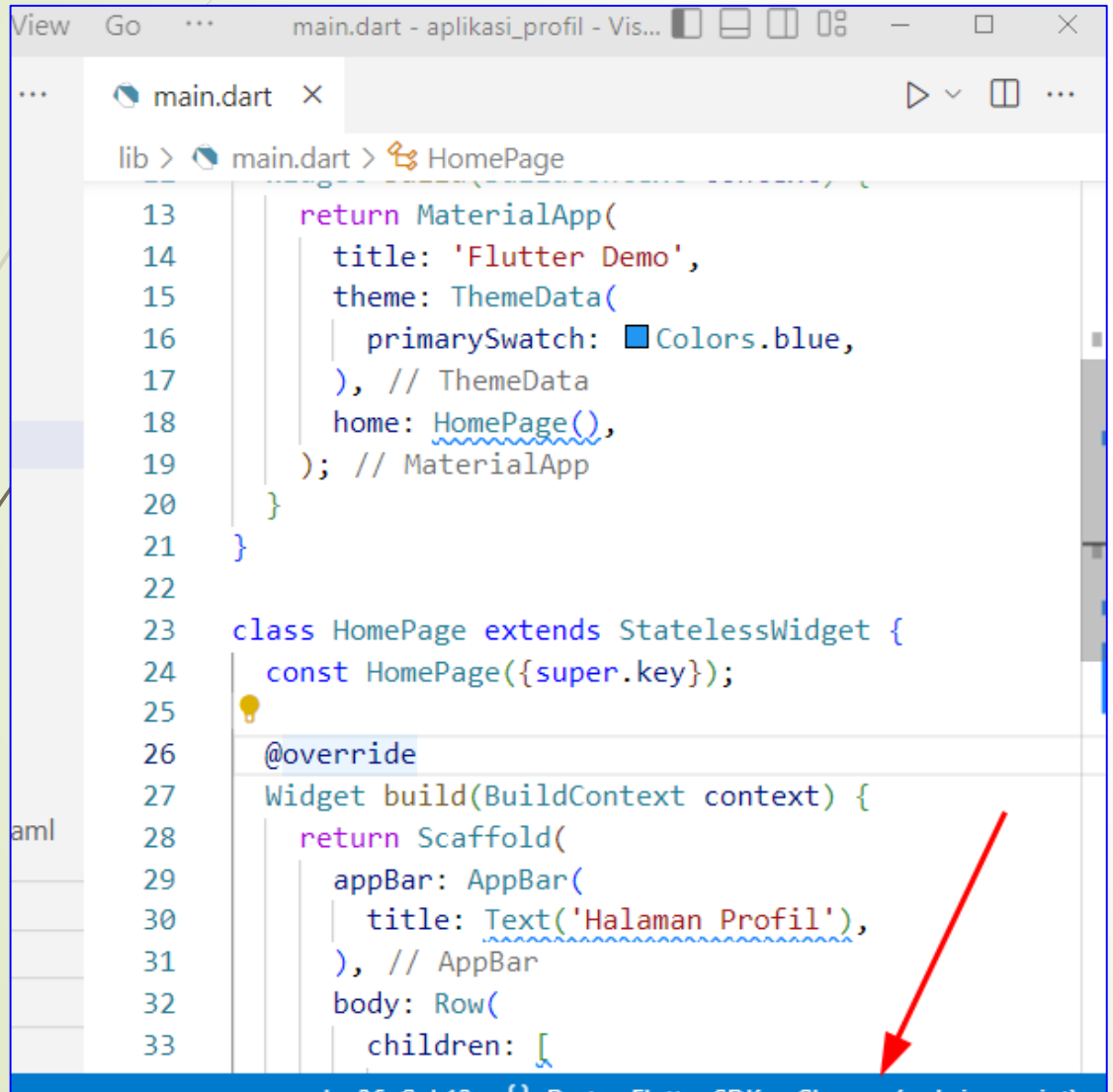
Arsitektur aplikasi flutter, hello world



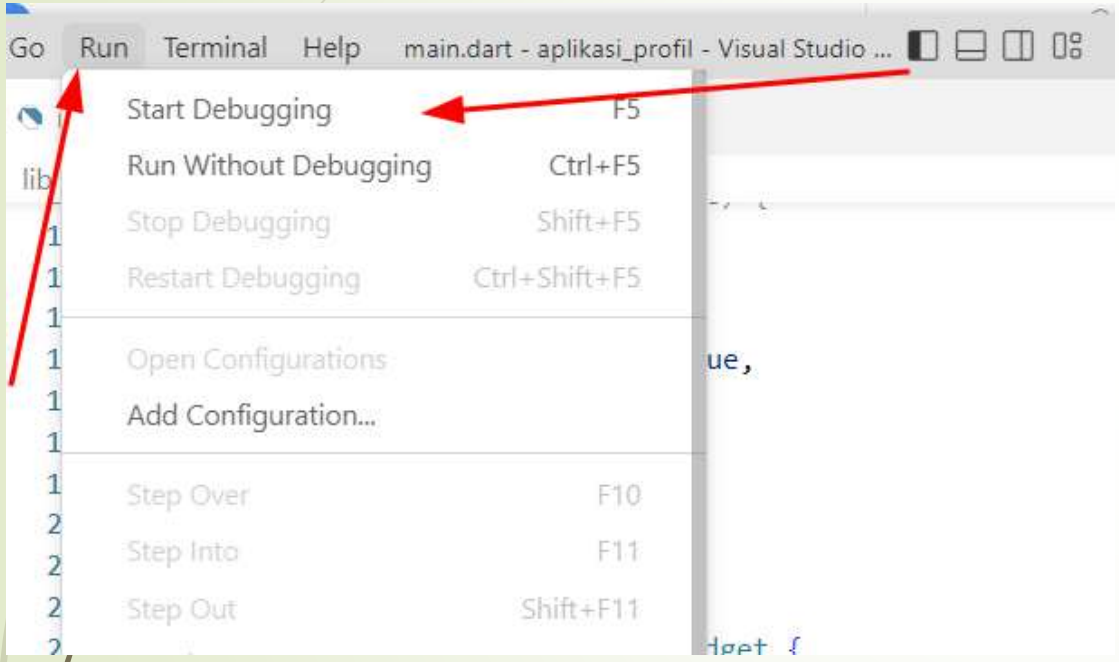
Arsitektur aplikasi flutter, hello world

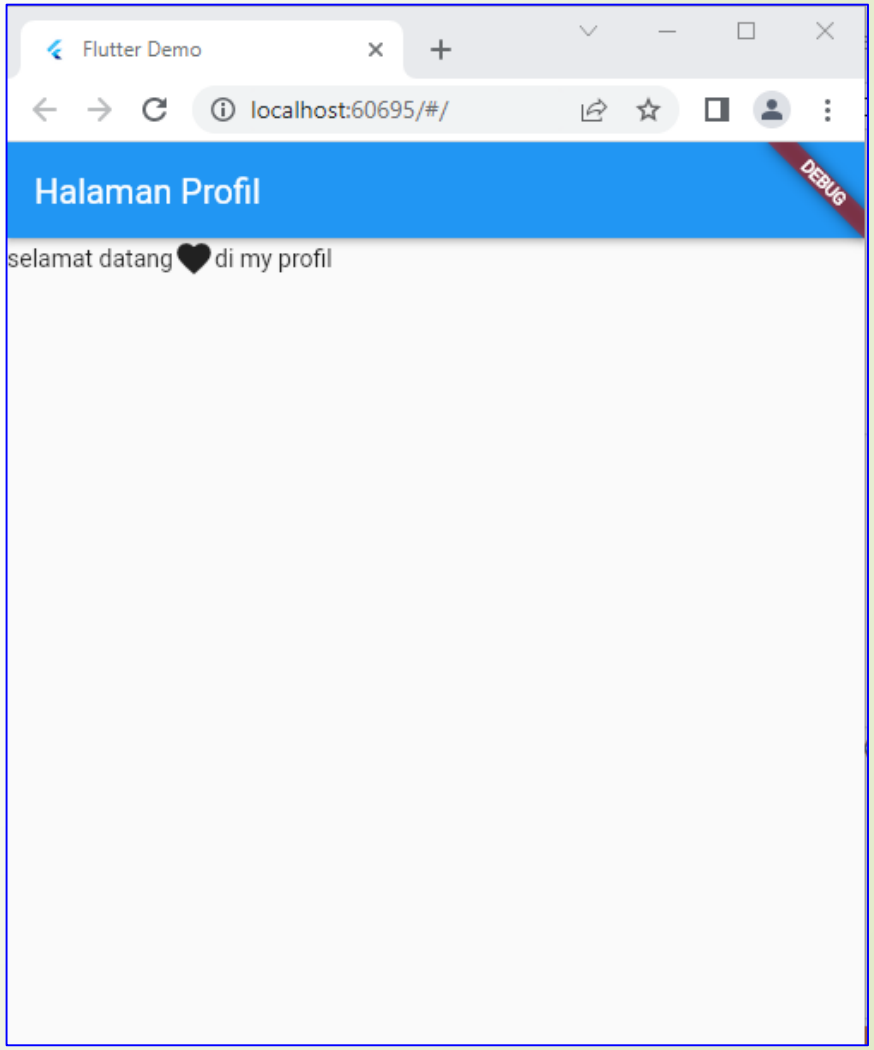


Arsitektur aplikasi flutter, hello world



Arsitektur aplikasi flutter, hello world

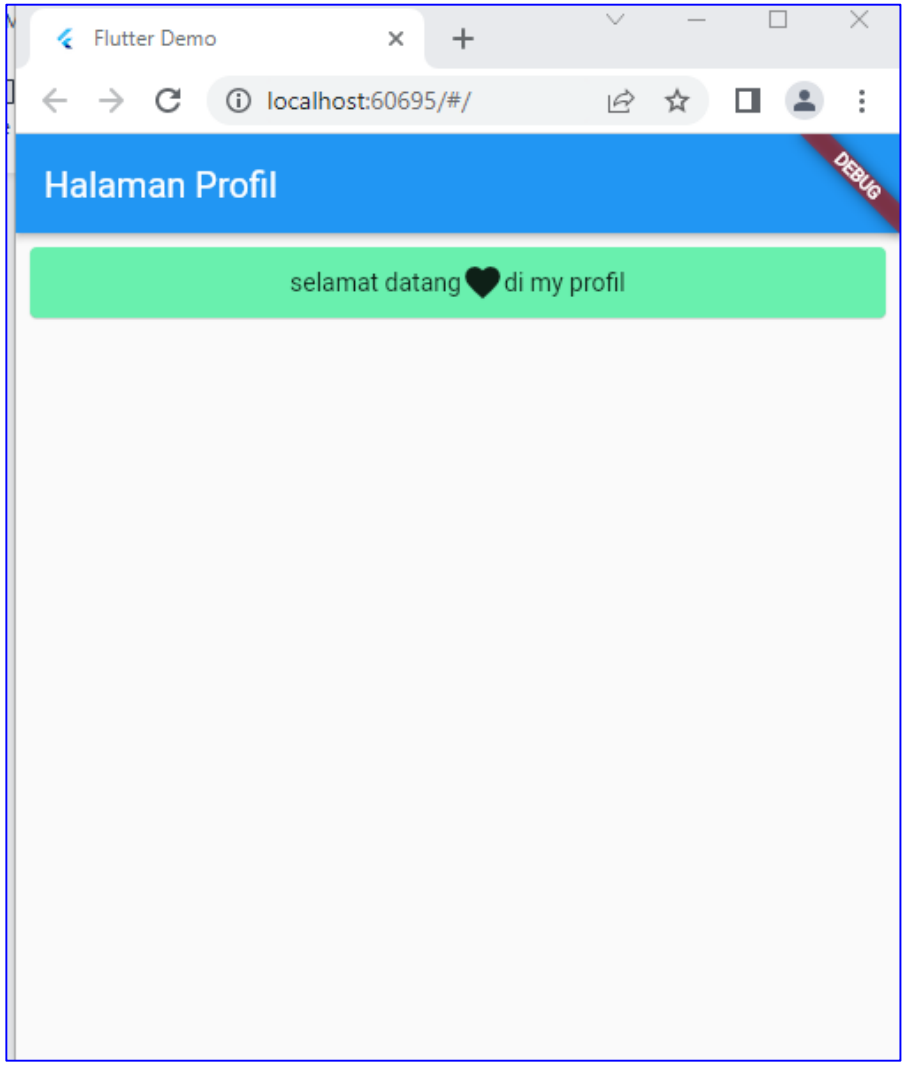




Arsitektur aplikasi flutter, hello world

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Halaman Profil'),
      ),
      body: Card(
        margin: EdgeInsets.all(8),
        color: Colors.greenAccent,
        child: Padding(
          padding: const EdgeInsets.all(8.0),
          child: Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Text('selamat datang'),
              Icon(Icons.favorite),
              Text('di my profil')
            ],
          ),
        ),
      ),
    );
  }
}
```

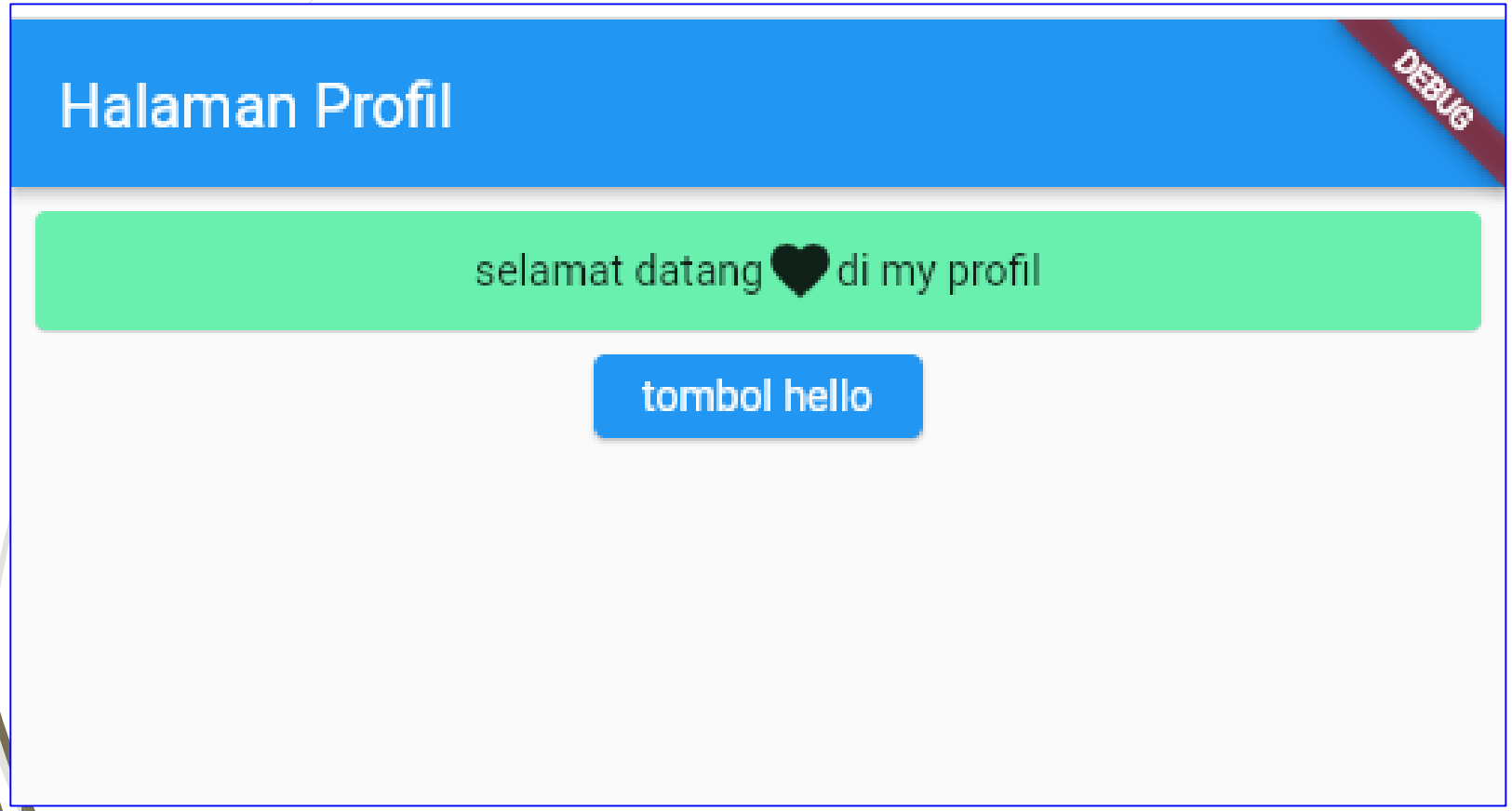
Arsitektur aplikasi flutter, hello world



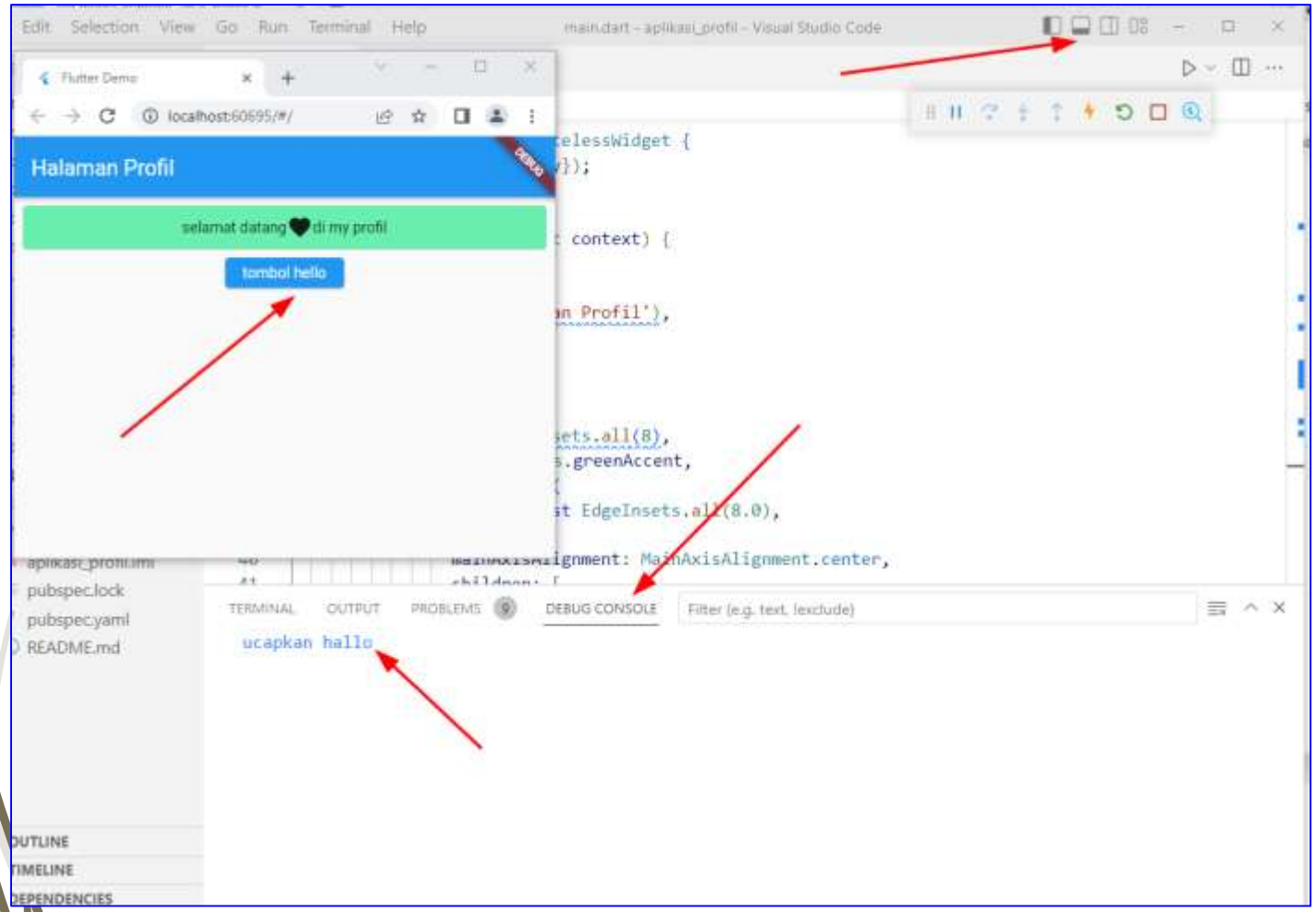
Arsitektur aplikasi flutter, hello world

```dart
class HomePage extends StatelessWidget {
  const HomePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Halaman Profil'),
      ), // AppBar
      body: Column(
        children: [
          Card(
            margin: EdgeInsets.all(8),
            color: Colors.greenAccent,
            child: Padding(
              padding: const EdgeInsets.all(8.0),
              child: Row(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
                  Text('selamat datang'),
                  Icon(Icons.favorite),
                  Text('di my profil')
                ],
              ), // Row
            ), // Padding
          ), // Card
          ElevatedButton(
            onPressed: () {
              print('ucapkan hallo');
            },
            child: Text('tombol hello'),
          ), // ElevatedButton
        ],
      ), // Column
    ); // Scaffold
  }
}
```

Arsitektur aplikasi flutter, hello world



Arsitektur aplikasi flutter, hello world



Pengenalan Tipe Data

- **String**
  String adalah kumpulan huruf, angka, dan tanda baca lainnya. Jadi isinya bebas asalkan tulisan.

```
void main() {
  String data = "Hello World!";
  print(data);
}
```

Pengenalan Tipe Data

- **Integer**
  integer merupakan tipe data yang merepresentasikan angka tanpa desimal.

```
void main() {
  int a = 10;
  print(a);
}
```

Pengenalan Tipe Data

- **Numerik**
  num, merupakan integer atau angka floating poin. Tipe data ini bisa menjadi integer ataupun double.

```
void main() {
  num a = 0.1;
  num b = 0.2;
  num c = a + b;
  print(c);
}
```

Pengenalan Tipe Data

- **Double**
  double merupakan angka decimal

```
void main() {
  double a = 0.1;
  print(a);
}
```

Pengenalan Tipe Data

- **Dynamic**
  dynamic merupakan tipe data yang isinya bisa berubah, dynamic bisa berupa integer, string, double, num, object, dan lain sebagainya.

```
void main() {
  dynamic a = 0.1;
  a = "Hello World!";
  print(a);
}
```

Pengenalan Tipe Data

- **List**
  List merupakan tumpukan data. Jika kalian familiar dengan **PHP** atau **Javascript** , List merupakan sebuah **Array**.

isi data dari List bisa dideskripsikan secara statis, contohnya **List<double>, List<String>, List<int>** , dan terserah lagi.

Jika kalian tidak mendeklarasikan tipe data dari isi List, maka hasilnya akan menjadi **List<dynamic>.**

```
void main() {
  List data = [1, 2, 3, "a", "b", "c"];
  print(data);
}
```

Pengenalan Tipe Data

- **Map**
  Map merupakan pasangan kunci dan isi. Kunci dan isi tersebut biasa dipanggil dengan **key** dan **value.** Setiap key dan value dapat dideklarasikan tipe datanya. Strukturnya sama dengan **JSON (Javascript Object Notation)**

```
void main() {
  Map<String, dynamic> data = {
    "url": "https://gepcode.com",
    "domain": "gepcode.com",
    "penulis": "Gilang Pratama",
    "totalAdmin": 1,
    "supportSeo": 100,
  };

  print(data);
}
```

Pengenalan Tipe Data

- **Boolean**
  Merupakan tipe data yang berisikan kebenaran. (Yes/No)

```
void main() {
  Map<String, dynamic> data = {
    "url": "https://gepcode.com",
    "domain": "gepcode.com",
    "penulis": "Gilang Pratama",
    "totalAdmin": 1,
    "supportSeo": 100,
  };

  print(data);
}
```

Pengenalan Tipe Data

- **Boolean**
  Merupakan tipe data yang berisikan kebenaran. (Yes/No)

```
void main() {
  var penulis = Person("Gilang Pratama", 27);

  print(penulis);
  print(penulis.name);
  print(penulis.age);
}
```

Pengenalan Tipe Data

- **Function**
  Function merupakan tipe data yang biasa digunakan untuk callback

```
void main() {
  cariBudi("Malang", (ada) {
    print(ada);
  });
  cariBudi("Banyuwangi", (ada) {
    print(ada);
  });
}

void cariBudi(String lokasi, void Function(String ada) cari) {
  if (lokasi == "Banyuwagi") {
    cari("Ada");
  } else {
    cari("Tidak ada");
  }
}
```

If Else

```
void main() {
  var teamA = 90;
  var teamB = 80;
  if (teamA > teamB) {
    print('Team A Menang');
  } else if (teamB > teamA) {
    print('Team B Menang');
  } else {
    print('Kedua Team Seri');
  }
}
```

Perbandingan nilai variabel yang ditentukan

If Else

Perbandingan nilai variabel yang ditentukan

```
void main() {
  //deklarasikan variabel
  var teamA = 50;
  var teamB = 85;
  //if statement
  if(teamA > teamB){
    print('Team A Menang');
  } else {
    print('Team B Menang');
  }
}
```

Pemrograman Dart

```dart
class PersegiPanjang {
  late double panjang; //initialisasi tipe data panjang
  late double lebar; //initialisasi tipe data lebar
  void setPanjang(double value) {
    if (value < 0) {
      // validasi jika nilai di bawah 0
      value *= -1; // akan di kalikan -1, misal -1 * -2 hasilnya 2
    }
    panjang = value; //alias
  }

  double getPanjang() {
    //get panjang
    return panjang; // mengembalikan nilai get panjang
  }

  void setLebar(double value) {
    if (value < 0) {
      // validasi jika nilai di bawah 0
      value *= -1; // akan di kalikan -1, misal -1 * -2 hasilnya 2
    }
    lebar = value; //alias
  }
```

- Buatlah file **oop.dart** dan masukkan kode berikut

Pemrograman Dart lanjutan...

```
  double getLebar() {
    //get panjang
    return lebar; // mengembalikan nilai get panjang
  }

  double hitungLuas() {
    return panjang * lebar; //mereturn hasil
  }
}
```

Pemrograman Dart

```dart
import 'oop.dart';

Run | Debug | Profile
void main(List<String> args) {
  PersegiPanjang kotak; // inisialisasi persegi panjang
  double luasKotak; // inisialisasi tipe data luas kotak

  // ignore: unnecessary_new
  kotak =
      new PersegiPanjang(); //menginisialisasi atau mengaliskan
      //kotak sebagai persegi panjang/ pointer menunjuk object persegipanjang
  kotak.setPanjang(
      12.0); // set nilai panjang(pembeda dengan method getter dan
      //setter adalah adanya setPanjang)
  kotak.setLebar(
      5.0); //set nilai lebar (pembeda dengan method getter
      //dan setter adalah adanya setLebar)

  luasKotak = kotak.hitungLuas(); // allias luaskotak
  print(luasKotak); // mencetak luas kotak
}
```

Buatlah file aksi.dart lalu masukkan kode berikut

Panggil file **oop.dart** yang telah dibuat dengan menggunakan perintah **import**

Pemrograman Dart

```
E:\BIRO\FIKOM\pemrograman_mobile\profil\lib>dart aksi.dart
60.0
```

- Jalankan perintah dart berikut untuk melihat hasil perhitungan Dari rumus yang telah dibuat

```
kotak.setPanjang(
    12.0); // set nilai panjang(pembeda dengan method getter dan
    //setter adalah adanya setPanjang)
kotak.setLebar(
    5.0); //set nilai lebar (pembeda dengan method getter
    //dan setter adalah adanya setLebar)

luasKotak = kotak.hitungLuas(); // allias luaskotak
print(luasKotak); // mencetak luas kotak
```

```dart
import 'dart:ui';

import 'package:flutter/material.dart';

Run | Debug | Profile
void main() => runApp(const MyApp());

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  static const String _title = 'Materi Flutter';

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: _title,
      home: Scaffold(
        appBar: AppBar(title: const Text(_title)),
        body: const MyStatefulWidget(),
      ), // Scaffold
    ); // MaterialApp
  }
}
```

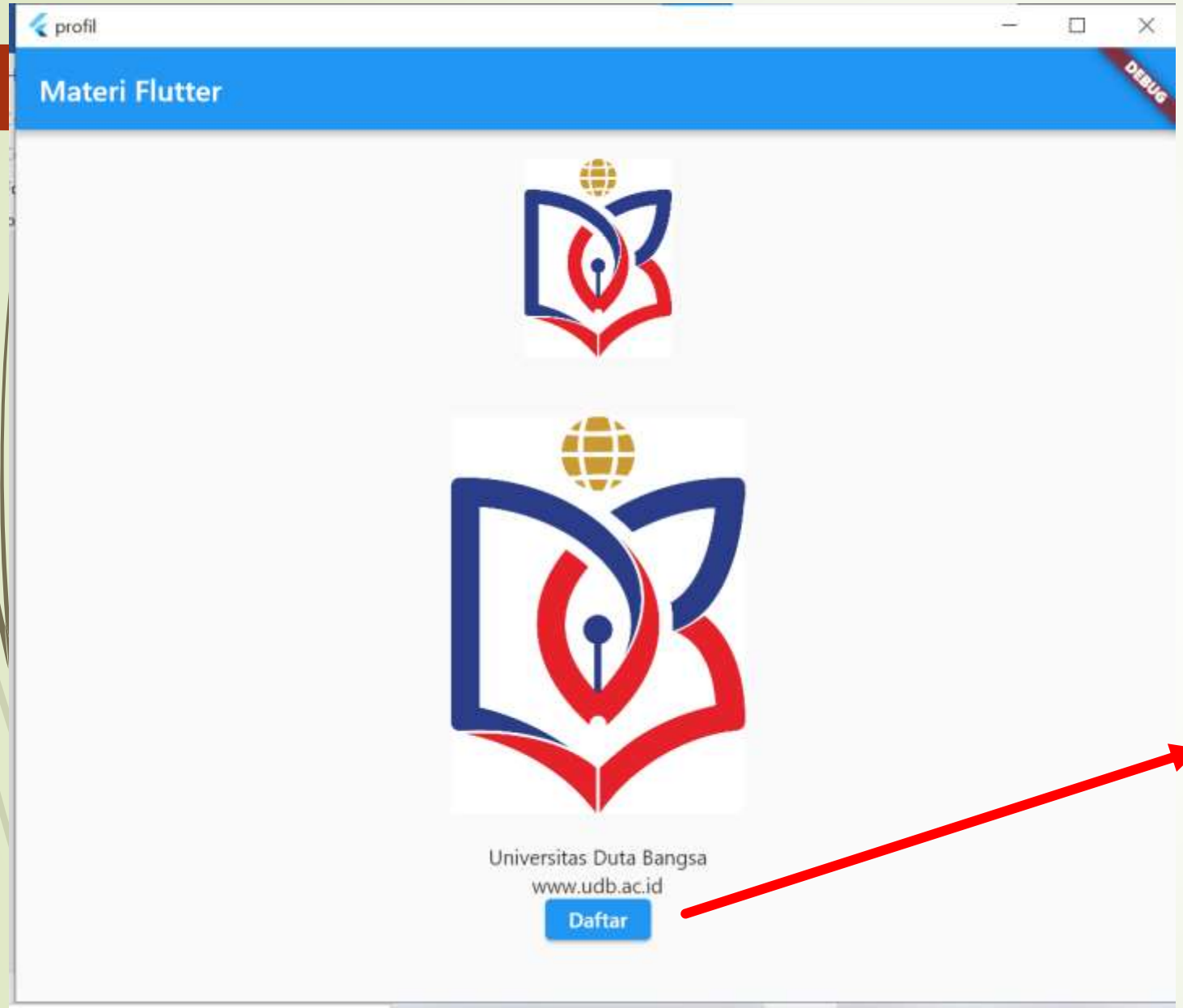
Widget Text, Button dan Image

Widget Text, Button dan Image

```
class MyStatefulWidget extends StatefulWidget {
  const MyStatefulWidget({Key? key}) : super(key: key);

  @override
  State<MyStatefulWidget> createState() => _MyStatefulWidgetState();
}

class _MyStatefulWidgetState extends State<MyStatefulWidget> {
  @override
  Widget build(BuildContext context) {
    return Center(
      child: Column(
        children: <Widget>[
          Container(
            margin: const EdgeInsets.all(20),
            child: const Image(
              width: 100,
              image: NetworkImage(
                  'https://udb.ac.id/storage/app/media/uploaded-files/LOGO.jpg'),
            ), // Image
          ), // Container
```

Widget Text, Button dan Image

```
      Container(
        margin: const EdgeInsets.all(20),
        child: const Image(
          width: 200,
          image: NetworkImage(
              'https://udb.ac.id/storage/app/media/uploaded-files/LOGO.jpg'),
        ), // Image
      ), // Container
      Text("Universitas Duta Bangsa"),
      Text("www.udb.ac.id"),
      ElevatedButton(
        onPressed: () {
          print('Selamat, Pendaftaran Anda Telah kami Proses');
        },
        child: Text('Daftar'),
      ), // ElevatedButton
    ], // <Widget>[]
  ), // Column
  ); // Center
 }
}
```


profil
Materi Flutter
DEBUG
Universitas Duta Bangsa
www.udb.ac.id
Daftar


PROBLEMS 33
OUTPUT
DEBUG CONSOLE
Filter (e.g. text, !exclude)
flutter: Selamat, Pendaftaran Anda Telah kami Proses

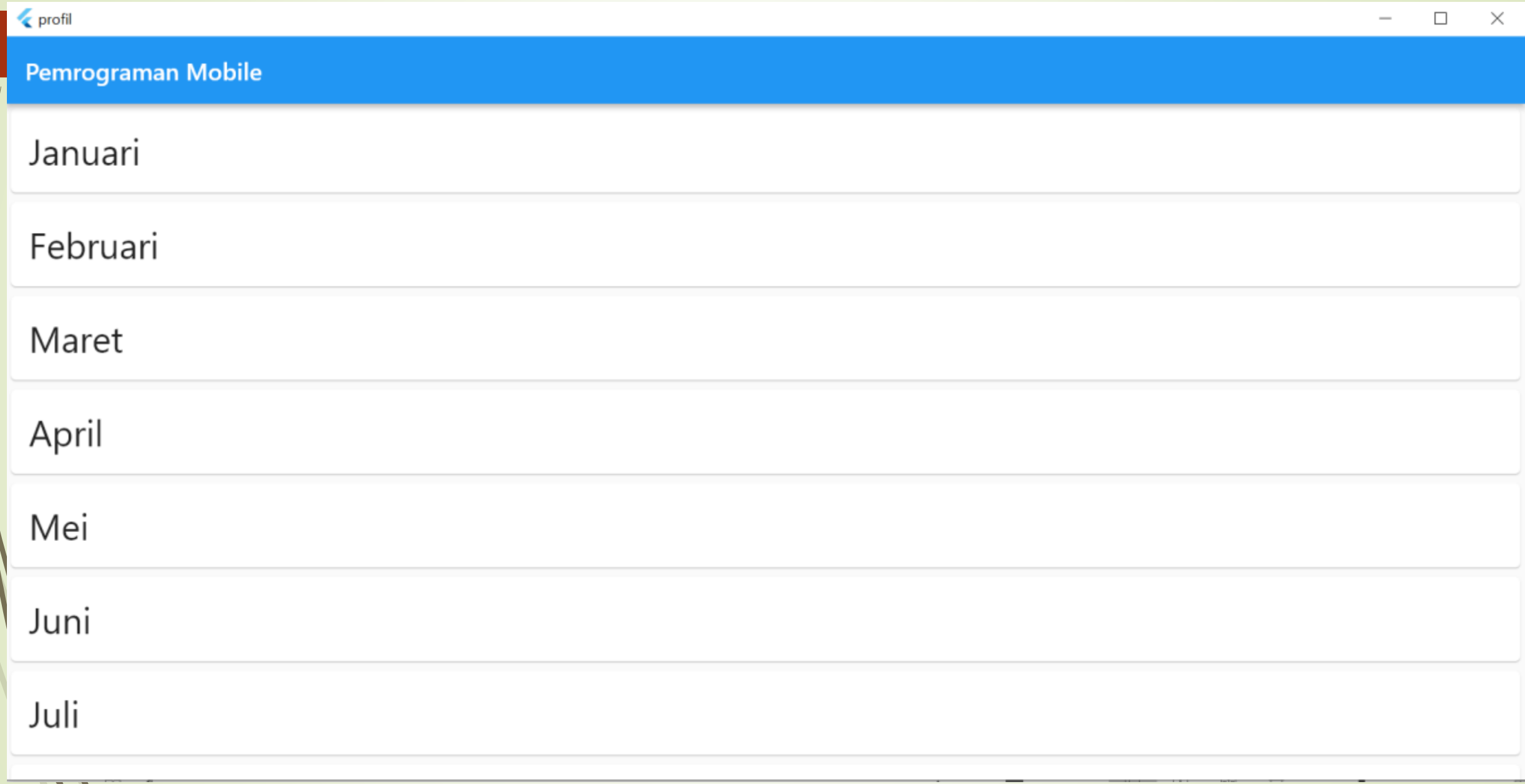
Pengenalan ListView

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

Run | Debug | Profile
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: BelajarListView(),
    ); // MaterialApp
  }
}
```

Array view

```
class BelajarListView extends StatelessWidget {
  final List bulan = [
    "Januari",
    "Fabruari",
    "Maret",
    "April",
    "Mei",
    "Juni",
    "Juli",
    "Agustus",
    "September",
    "Oktober",
    "November",
    "Desember"
  ];
```

```dart
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Pemrograman Mobile"),
      ), // AppBar
      body: ListView.builder(
        itemBuilder: (context, index) {
          return Card(
            child: Padding(
              padding: const EdgeInsets.all(15.0),
              child: Text(bulan[index], style: TextStyle(fontSize: 30)),
            ), // Padding
          ); // Card
        },
        itemCount: bulan.length,
      ), // ListView.builder
    ); // Scaffold
  }
}
```


profil
Pemrograman Mobile
Januari
Februari
Maret
April
Mei
Juni
Juli

Gesture Flutter

- Gesture detector merupakan widget yang berfungsi untuk mendeteksi gesture atau aksi dari pengguna atau user
- Gesture Detector adalah widget yang berguna untuk memberikan informasi kepada pengguna saat menyentuh pada aplikasi. Biasanya gesture detector digunakan saat pesan singkat pada pengguna

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

Run | Debug | Profile
void main() {
  runApp(const MaterialApp(
    title: "Gesture Detector",
    debugShowCheckedModeBanner: false,
    home: MyApp(),
  )); // MaterialApp
}

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

```dart
  @override
  // ignore: library_private_types_in_public_api
  _MyAppState createState() => _MyAppState();
}

class _MyAppState extends State<MyApp> {
  late String _title;

  @override
  void initState() {
    super.initState();
    _title = "Gesture Detector";
  }
```

```
void _changeTitle(String title) {
  setState(() {
    _title = title;
  });
}
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      centerTitle: true,
      title: Text(_title),
    ), // AppBar
    body: ListView(
      children: <Widget>[
        GestureDetector(
          onTap: () {
            _changeTitle("onTap Clicked");
          },
```

```
    child: _buildItem("Click with tap"),
  ), // GestureDetector
  GestureDetector(
    onDoubleTap: () {
      _changeTitle("onDoubleTap Clicked");
    },
    child: _buildItem("Click with double tap"),
  ), // GestureDetector
  GestureDetector(
    onLongPress: () {
      _changeTitle("onLongPress Clicked");
    },
```

```
    child: _buildItem("Click with long press"),
  ), // GestureDetector
  GestureDetector(
    onVerticalDragDown: (dragDetails) {
      _changeTitle("Vertical Drag Clicked");
    },
    child: _buildItem("Click with vertical drag"),
  ), // GestureDetector
  GestureDetector(
    onHorizontalDragEnd: (dragDetails) {
      _changeTitle("Horizontal Drag Clicked");
    },
```

```
            child: _buildItem("Click with horizontal drag"),
          ), // GestureDetector
          GestureDetector(
            onScaleEnd: (scaleDetails) {
              _changeTitle("Scale Clicked");
            },
            child: _buildItem("Click with scale left"),
          ), // GestureDetector
        ], // <Widget>[]
      ), // ListView
    ); // Scaffold
}
```

```
Widget _buildItem(String title) {
  return Container(
    margin: const EdgeInsets.all(8.0),
    padding: const EdgeInsets.all(8.0),
    color: Colors.blue,
    child: Center(
      child: Text(
        title,
        style: const TextStyle(
          color: Colors.white,
          fontWeight: FontWeight.bold,
        ), // TextStyle
      ), // Text
    ), // Center
  ); // Container
}
}
```


profil
Gesture Detector
Click with tap
Click with double tap
Click with long press
Click with vertical drag
Click with horizontal drag
Click with scale left
onTap Clicked
Click with tap
Click with double tap
Click with long press
Click with vertical drag
Click with horizontal drag
Click with scale left


profil
Gesture Detector
Click with tap
Click with double tap
Click with long press
Click with vertical drag
Click with horizontal drag
Click with scale left
profil
onDoubleTap Clicked
Click with tap
Click with double tap
Click with long press
Click with vertical drag
Click with horizontal drag
Click with scale left

Terimakasih