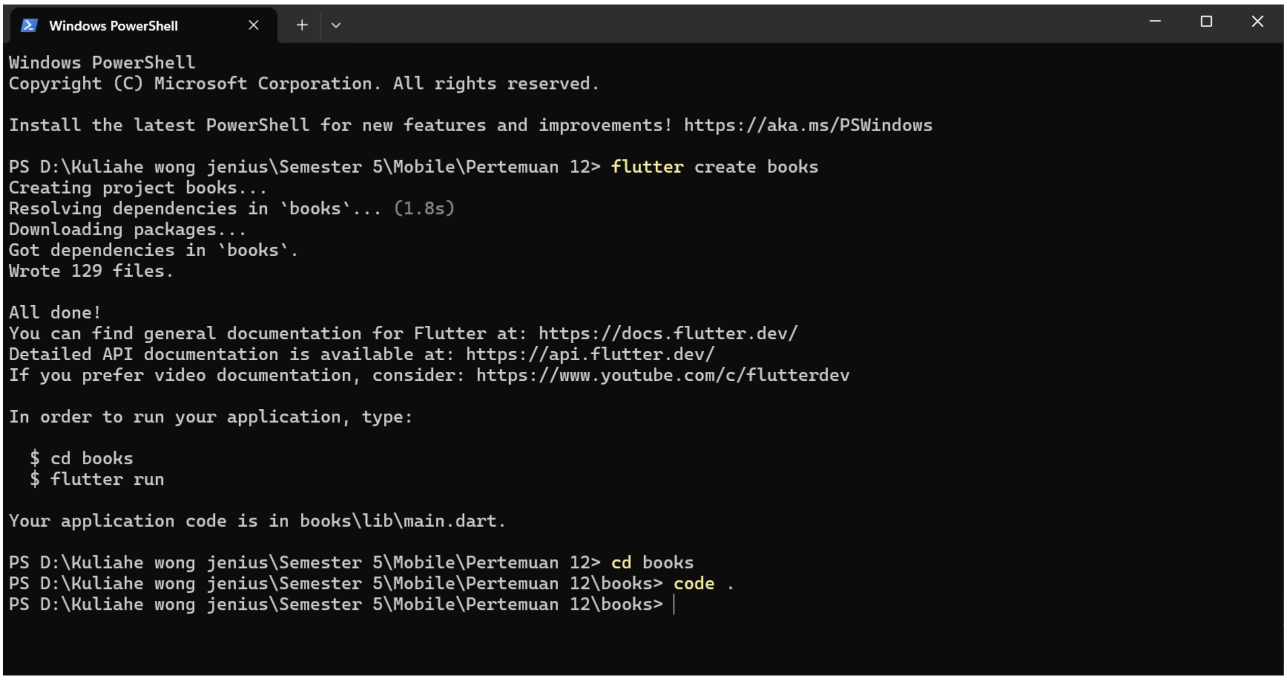
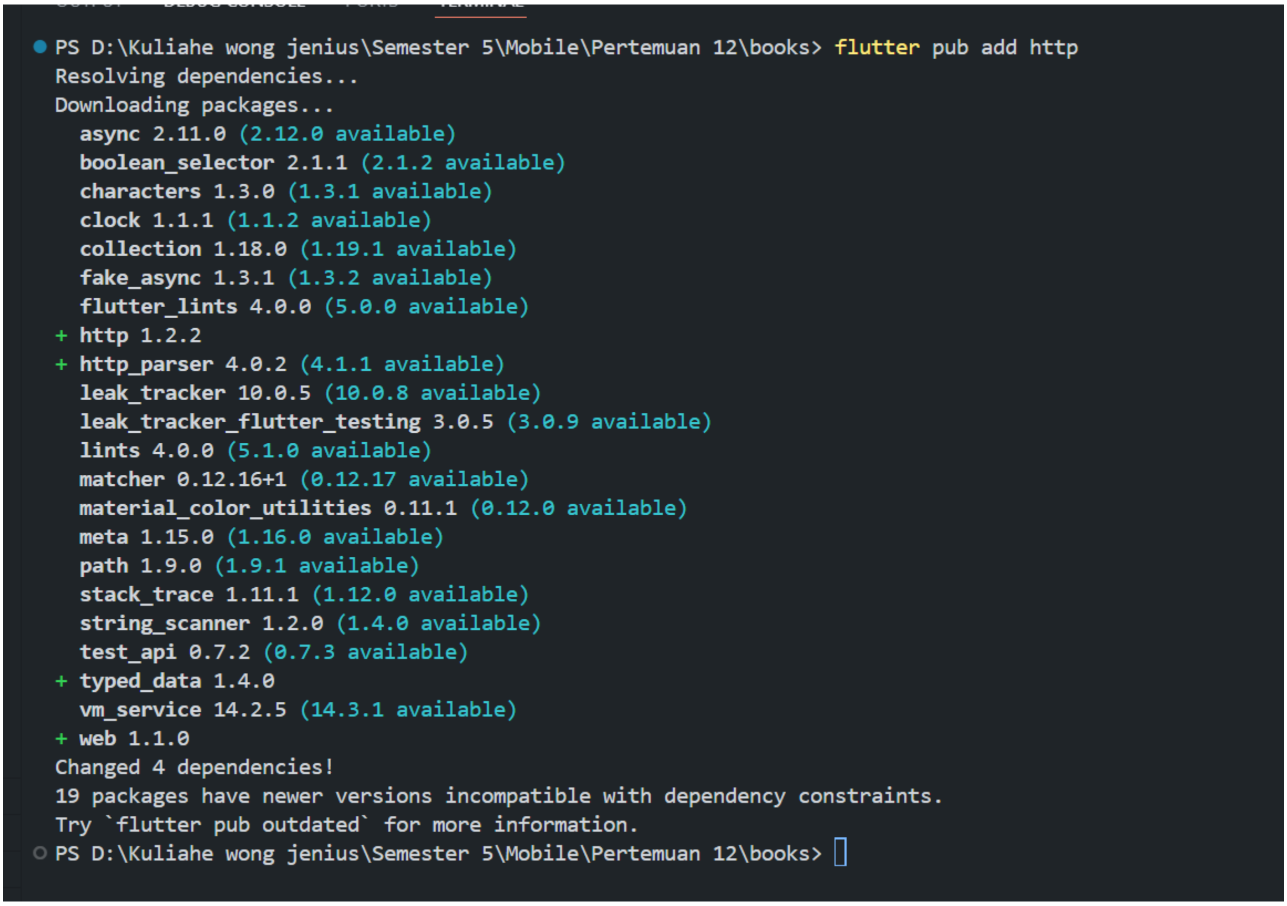
Pertemuan 12

Praktikum 1

1. Buat project baru

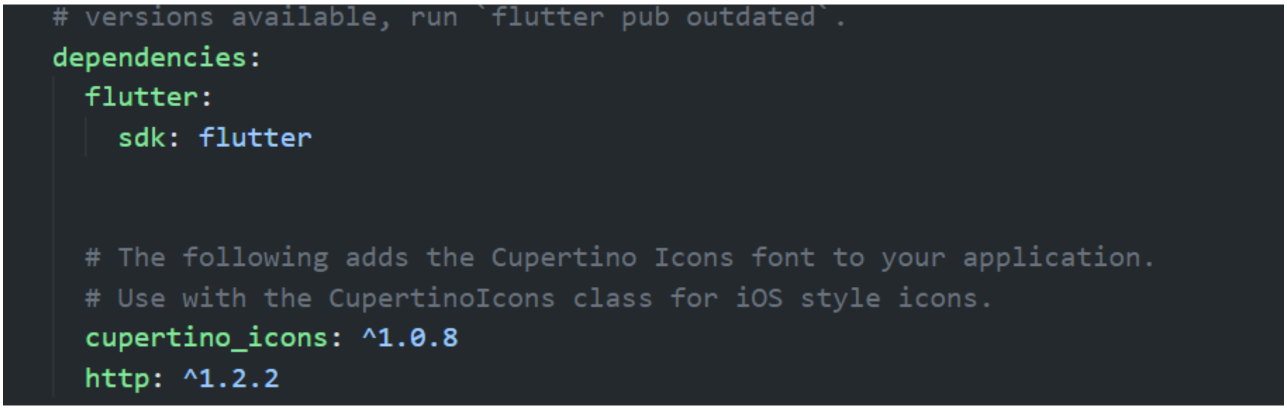


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Kuliahe wong jenius\Semester 5\Mobile\Pertemuan 12> flutter create books
Creating project books...
Resolving dependencies in `books`... (1.8s)
Downloading packages...
Got dependencies in `books`.
Wrote 129 files.

All done!
You can find general documentation for Flutter at: https://docs.flutter.dev/
Detailed API documentation is available at: https://api.flutter.dev/
If you prefer video documentation, consider: https://www.youtube.com/c/flutterdev

In order to run your application, type:

  $ cd books
  $ flutter run

Your application code is in books\lib\main.dart.

PS D:\Kuliahe wong jenius\Semester 5\Mobile\Pertemuan 12> cd books
PS D:\Kuliahe wong jenius\Semester 5\Mobile\Pertemuan 12\books> code .
PS D:\Kuliahe wong jenius\Semester 5\Mobile\Pertemuan 12\books>
```



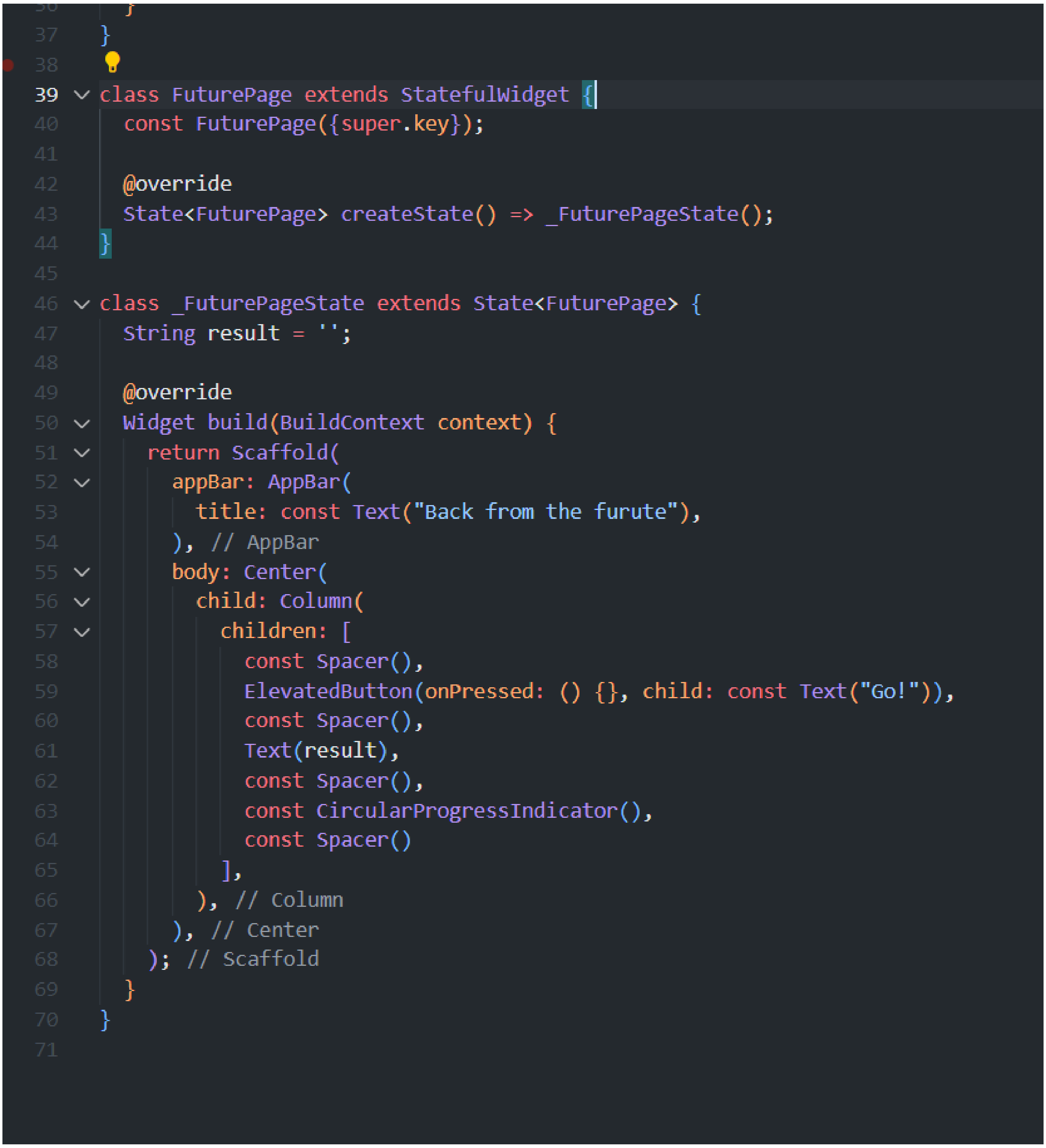
```
● PS D:\Kuliahe wong jenius\Semester 5\Mobile\Pertemuan 12\books> flutter pub add http
  Resolving dependencies...
  Downloading packages...
    async 2.11.0 (2.12.0 available)
    boolean_selector 2.1.1 (2.1.2 available)
    characters 1.3.0 (1.3.1 available)
    clock 1.1.1 (1.1.2 available)
    collection 1.18.0 (1.19.1 available)
    fake_async 1.3.1 (1.3.2 available)
    flutter_lints 4.0.0 (5.0.0 available)
  + http 1.2.2
  + http_parser 4.0.2 (4.1.1 available)
    leak_tracker 10.0.5 (10.0.8 available)
    leak_tracker_flutter_testing 3.0.5 (3.0.9 available)
    lints 4.0.0 (5.1.0 available)
    matcher 0.12.16+1 (0.12.17 available)
    material_color_utilities 0.11.1 (0.12.0 available)
    meta 1.15.0 (1.16.0 available)
    path 1.9.0 (1.9.1 available)
    stack_trace 1.11.1 (1.12.0 available)
    string_scanner 1.2.0 (1.4.0 available)
    test_api 0.7.2 (0.7.3 available)
  + typed_data 1.4.0
    vm_service 14.2.5 (14.3.1 available)
  + web 1.1.0
  Changed 4 dependencies!
  19 packages have newer versions incompatible with dependency constraints.
  Try `flutter pub outdated` for more information.
○ PS D:\Kuliahe wong jenius\Semester 5\Mobile\Pertemuan 12\books>
```

2. Cek di pusbspec.yaml



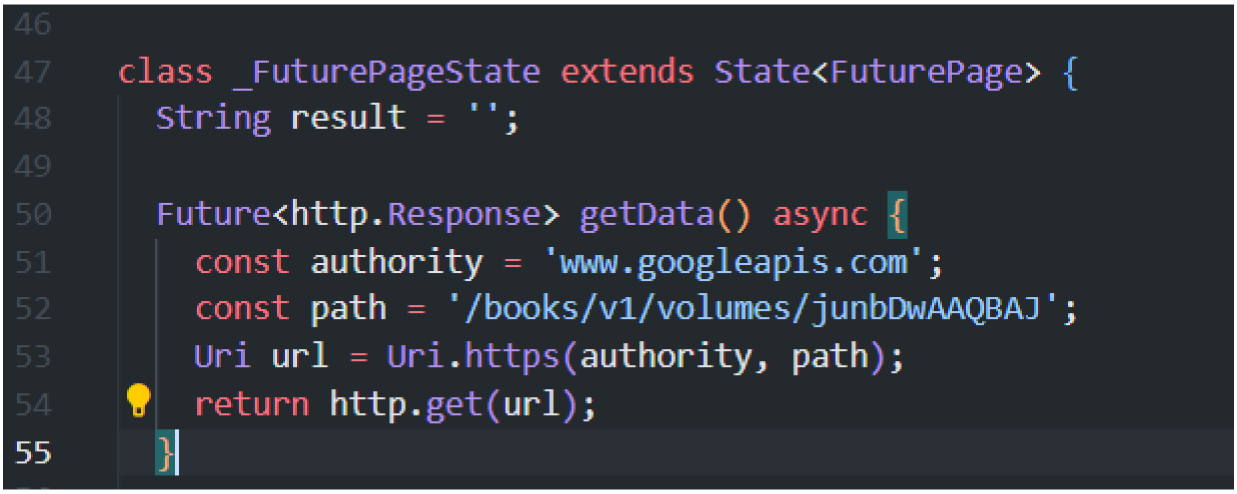
```yaml
# versions available, run `flutter pub outdated`.
dependencies:
  flutter:
    sdk: flutter


  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.8
  http: ^1.2.2
```

3. buka file main.dart

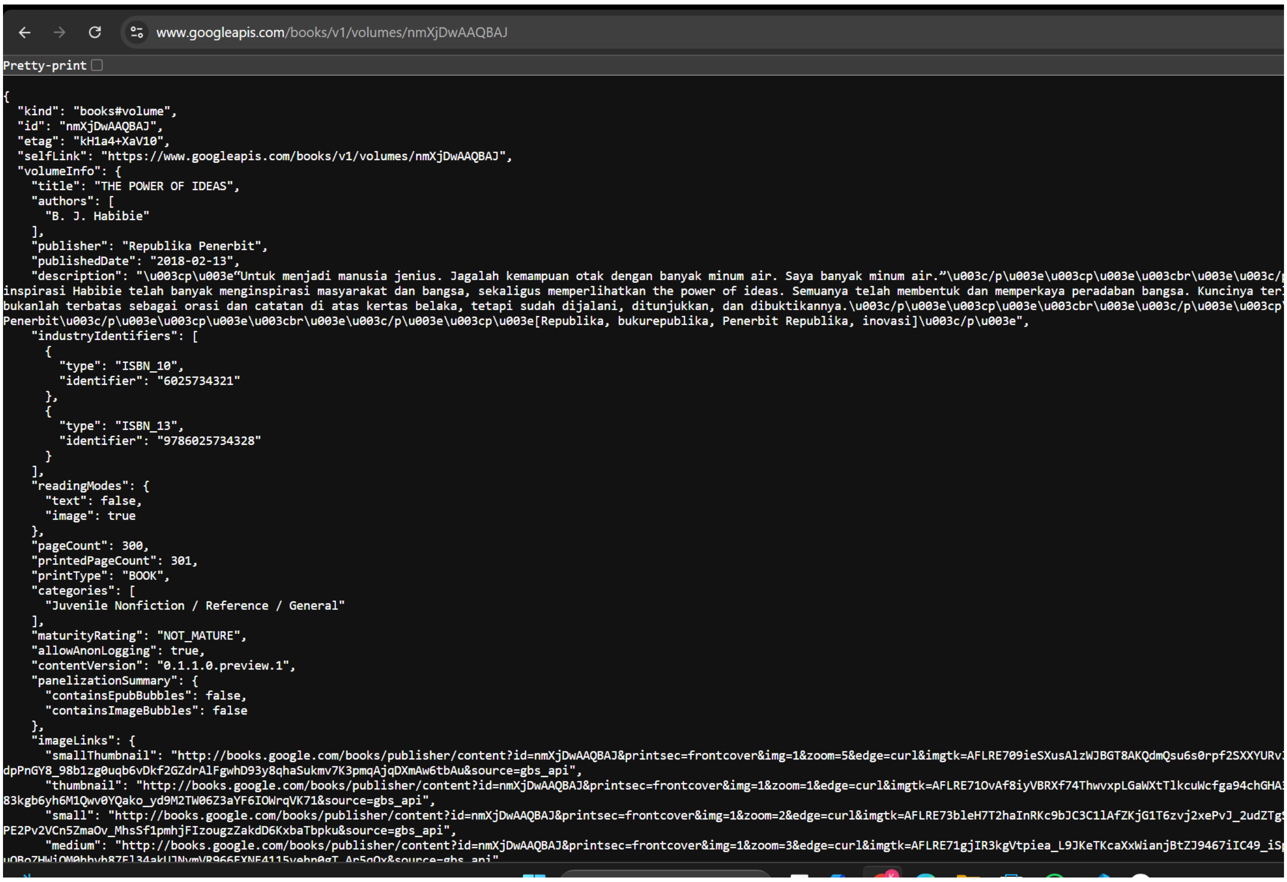


```dart
  }
}

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

  @override
  State<FuturePage> createState() => _FuturePageState();
}

class _FuturePageState extends State<FuturePage> {
  String result = '';

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Back from the furute"),
      ), // AppBar
      body: Center(
        child: Column(
          children: [
            const Spacer(),
            ElevatedButton(onPressed: () {}, child: const Text("Go!")),
            const Spacer(),
            Text(result),
            const Spacer(),
            const CircularProgressIndicator(),
            const Spacer()
          ],
        ), // Column
      ), // Center
    ); // Scaffold
  }
}
```

4. Tambah method getData()



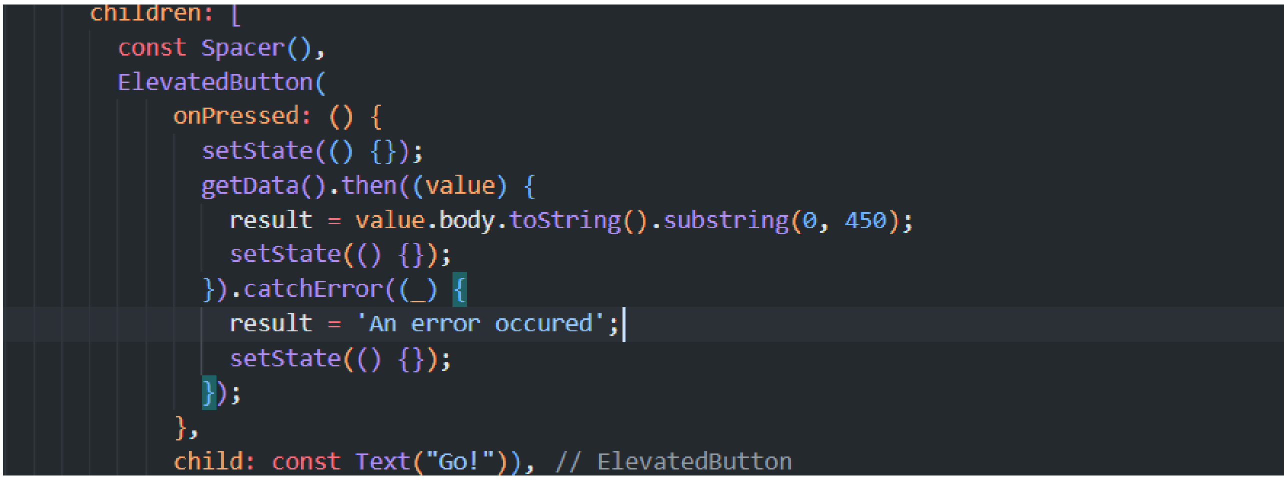
```dart

class _FuturePageState extends State<FuturePage> {
  String result = '';

  Future<http.Response> getData() async {
    const authority = 'www.googleapis.com';
    const path = '/books/v1/volumes/junbDwAAQBAJ';
    Uri url = Uri.https(authority, path);
    return http.get(url);
  }
```



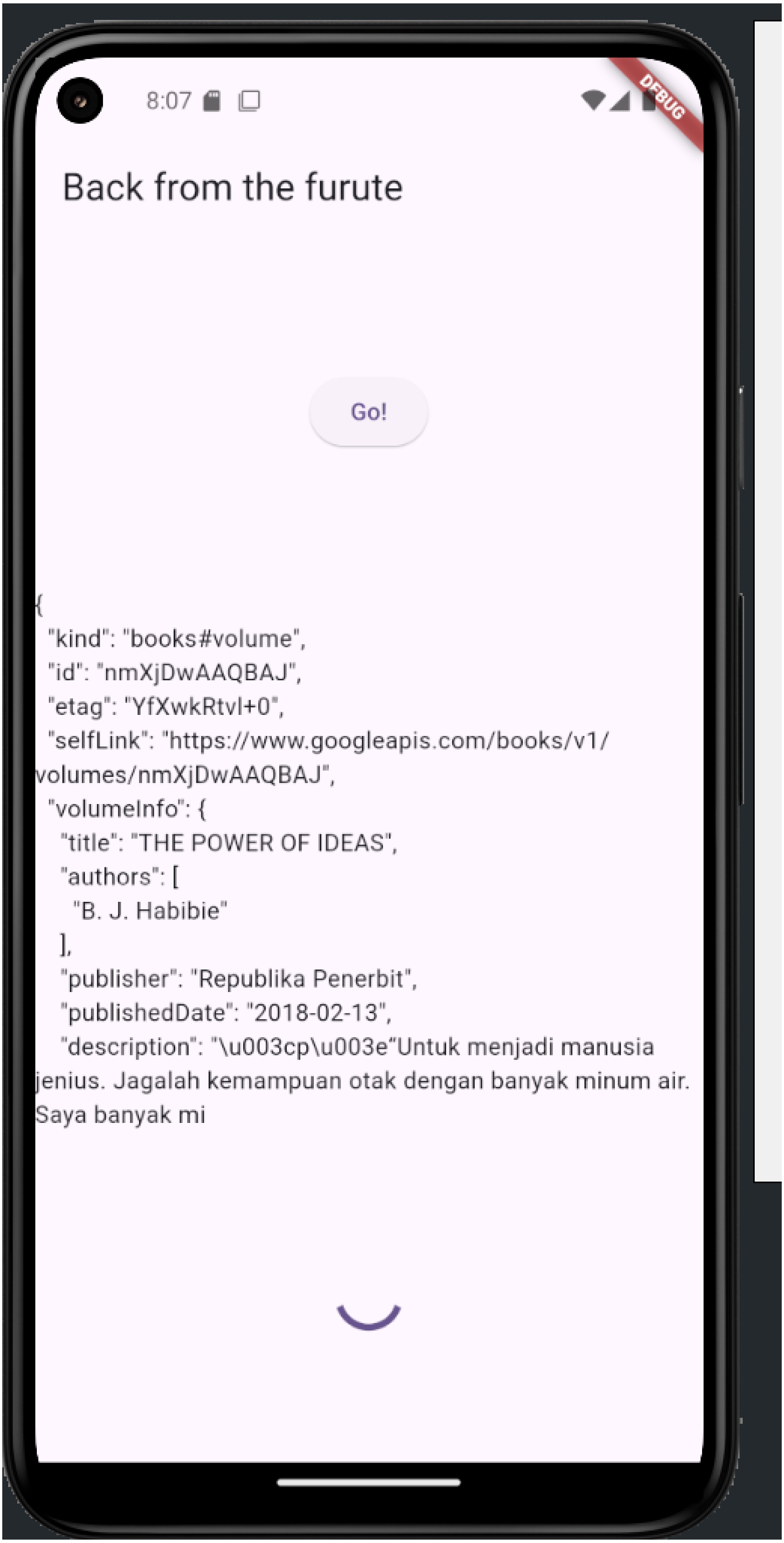
www.googleapis.com/books/v1/volumes/nmXjDwAAQBAJ

Pretty-print □

```
{
  "kind": "books#volume",
  "id": "nmXjDwAAQBAJ",
  "etag": "kH1a4+XaV10",
  "selfLink": "https://www.googleapis.com/books/v1/volumes/nmXjDwAAQBAJ",
  "volumeInfo": {
    "title": "THE POWER OF IDEAS",
    "authors": [
      "B. J. Habibie"
    ],
    "publisher": "Republika Penerbit",
    "publishedDate": "2018-02-13",
    "description": "\u003cp\u003e“Untuk menjadi manusia jenius. Jagalah kemampuan otak dengan banyak minum air. Saya banyak minum air.”\u003c/p\u003e\u003cp\u003e\u003cbr\u003e\u003c/p
inspirasi Habibie telah banyak menginspirasi masyarakat dan bangsa, sekaligus memperlihatkan the power of ideas. Semuanya telah membentuk dan memperkaya peradaban bangsa. Kuncinya ter
bukanlah terbatas sebagai orasi dan catatan di atas kertas belaka, tetapi sudah dijalani, ditunjukkan, dan dibuktikannya.\u003c/p\u003e\u003cp\u003e\u003cbr\u003e\u003c/p\u003e\u003cp
Penerbit\u003c/p\u003e\u003cp\u003e\u003cbr\u003e\u003c/p\u003e\u003cp\u003e[Republika, bukurepublika, Penerbit Republika, inovasi]\u003c/p\u003e",
    "industryIdentifiers": [
      {
        "type": "ISBN_10",
        "identifier": "6025734321"
      },
      {
        "type": "ISBN_13",
        "identifier": "9786025734328"
      }
    ],
    "readingModes": {
      "text": false,
      "image": true
    },
    "pageCount": 300,
    "printedPageCount": 301,
    "printType": "BOOK",
    "categories": [
      "Juvenile Nonfiction / Reference / General"
    ],
    "maturityRating": "NOT_MATURE",
    "allowAnonLogging": true,
    "contentVersion": "0.1.1.0.preview.1",
    "panelizationSummary": {
      "containsEpubBubbles": false,
      "containsImageBubbles": false
    },
    "imageLinks": {
      "smallThumbnail": "http://books.google.com/books/publisher/content?id=nmXjDwAAQBAJ&printsec=frontcover&img=1&zoom=5&edge=curl&imgtk=AFLRE709ieSXusAlzWJBGT8AKQdmQsu6s0rpf2SXXYURv
dpPnGY8_98b1zg0uqb6vDkf2GZdrAlFgwhD93y8qhaSukmv7K3pmqAjqDXmAw6tbAu&source=gbs_api",
      "thumbnail": "http://books.google.com/books/publisher/content?id=nmXjDwAAQBAJ&printsec=frontcover&img=1&zoom=1&edge=curl&imgtk=AFLRE71OvAf8iyVBRXf74ThwvxpLGaWXtTlkcuWcfga94chGHA
83kgb6yh6M1Qwv0YQako_yd9M2TW06Z3aYF6IOWrqVK71&source=gbs_api",
      "small": "http://books.google.com/books/publisher/content?id=nmXjDwAAQBAJ&printsec=frontcover&img=1&zoom=2&edge=curl&imgtk=AFLRE73bleH7T2haInRKc9bJC3C1lAfZKjG1T6zvj2xePvJ_2udZTg
PE2Pv2VCn5ZmaOv_MhsSf1pmhjFIzougzZakdD6KxbaTbpku&source=gbs_api",
      "medium": "http://books.google.com/books/publisher/content?id=nmXjDwAAQBAJ&printsec=frontcover&img=1&zoom=3&edge=curl&imgtk=AFLRE71gjIR3kgVtpiea_L9JKeTKcaXxWianjBtZJ9467iIC49_iS
uQBo7HWiOM0hbvh87El34akUJNvmVR966EXNE4115vehn0gT_Ar5qOx&source=gbs_api",
```



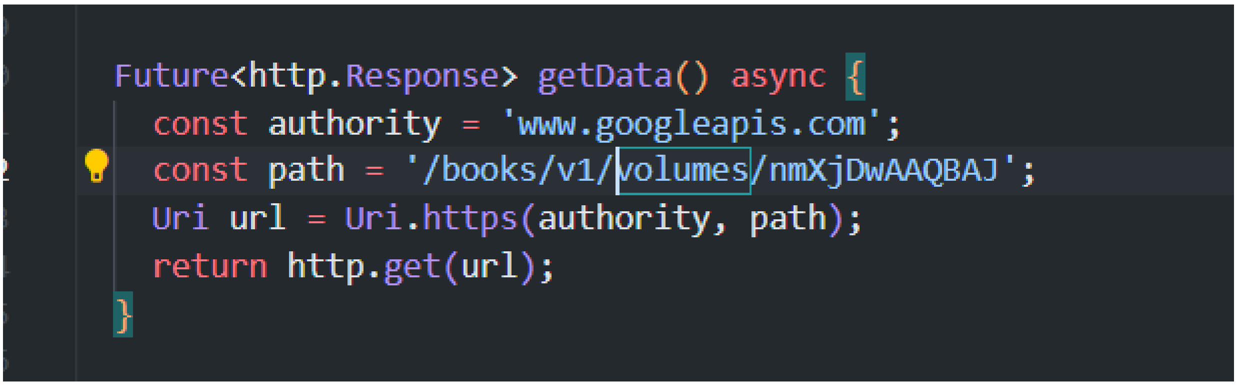
```
      children: [
        const Spacer(),
        ElevatedButton(
            onPressed: () {
              setState(() {});
              getData().then((value) {
                result = value.body.toString().substring(0, 450);
                setState(() {});
              }).catchError((_) {
                result = 'An error occured';
                setState(() {});
              });
            },
            child: const Text("Go!")), // ElevatedButton
```



Back from the furute

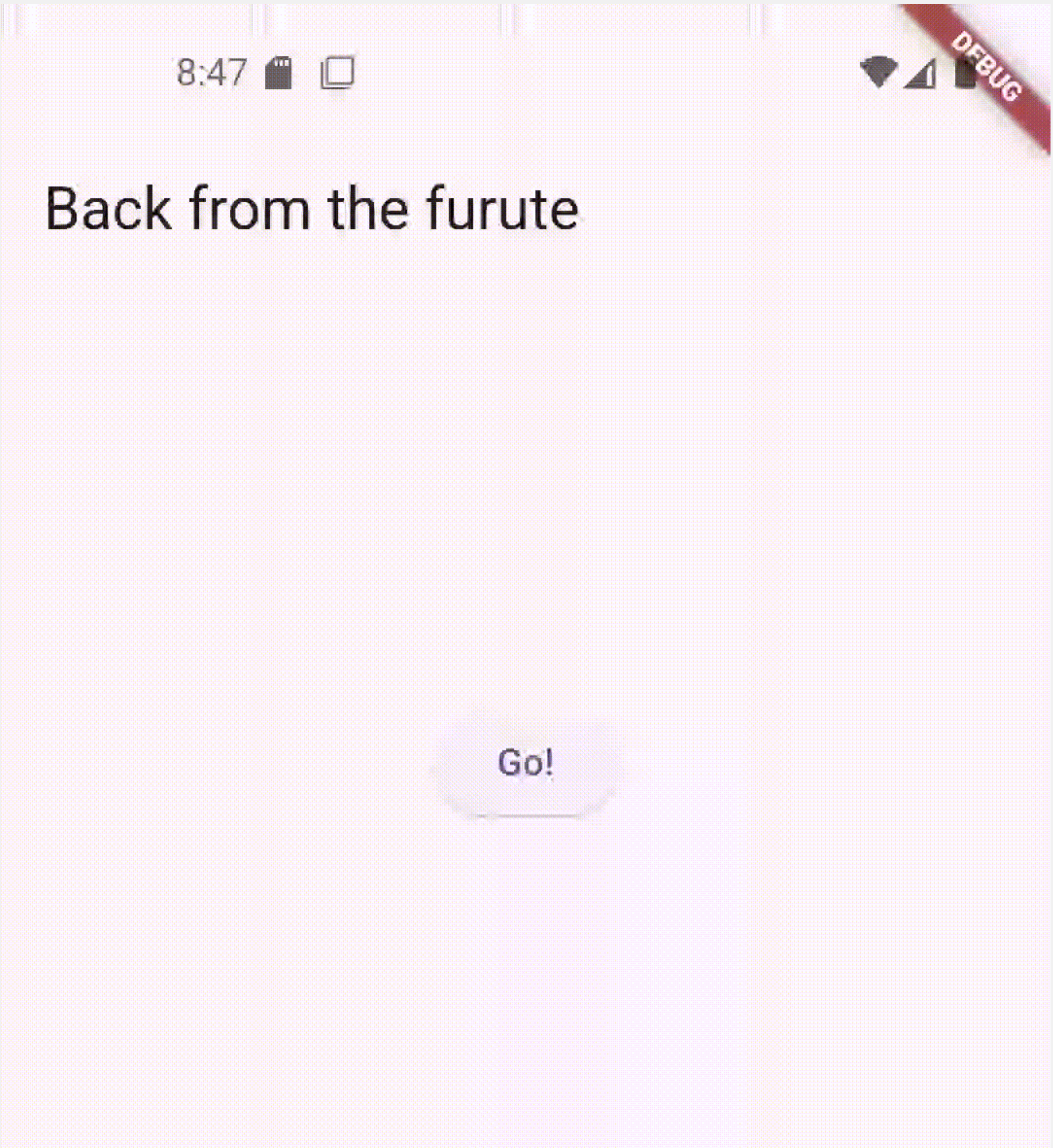
```
{
  "kind": "books#volume",
  "id": "nmXjDwAAQBAJ",
  "etag": "YfXwkRtvl+0",
  "selfLink": "https://www.googleapis.com/books/v1/
volumes/nmXjDwAAQBAJ",
  "volumeInfo": {
    "title": "THE POWER OF IDEAS",
    "authors": [
      "B. J. Habibie"
    ],
    "publisher": "Republika Penerbit",
    "publishedDate": "2018-02-13",
    "description": "\u003cp\u003e“Untuk menjadi manusia
jenius. Jagalah kemampuan otak dengan banyak minum air.
Saya banyak mi
```

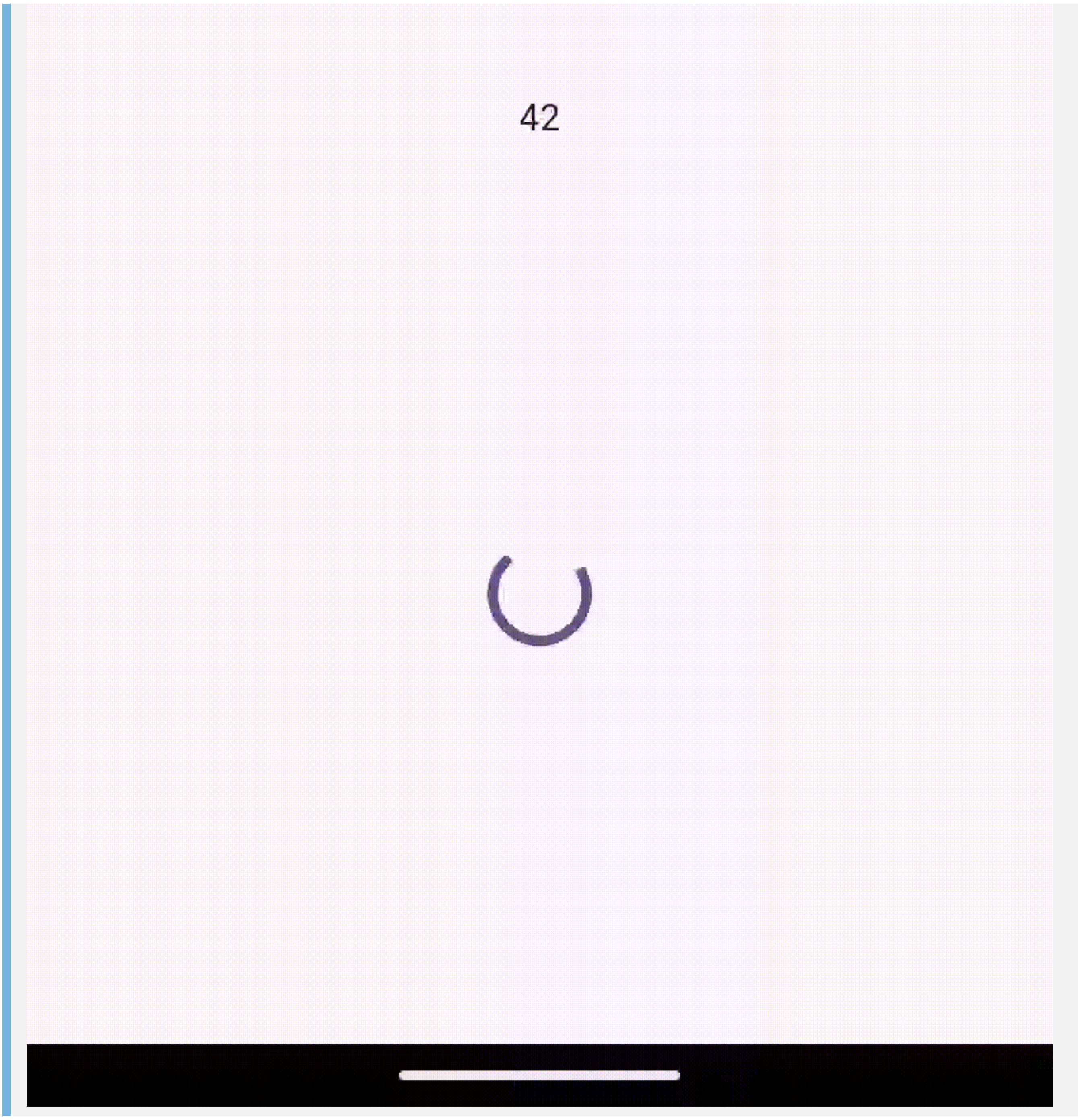
Soal 2



```
Future<http.Response> getData() async {
  const authority = 'www.googleapis.com';
  const path = '/books/v1/volumes/nmXjDwAAQBAJ';
  Uri url = Uri.https(authority, path);
  return http.get(url);
}
```

Soal 3

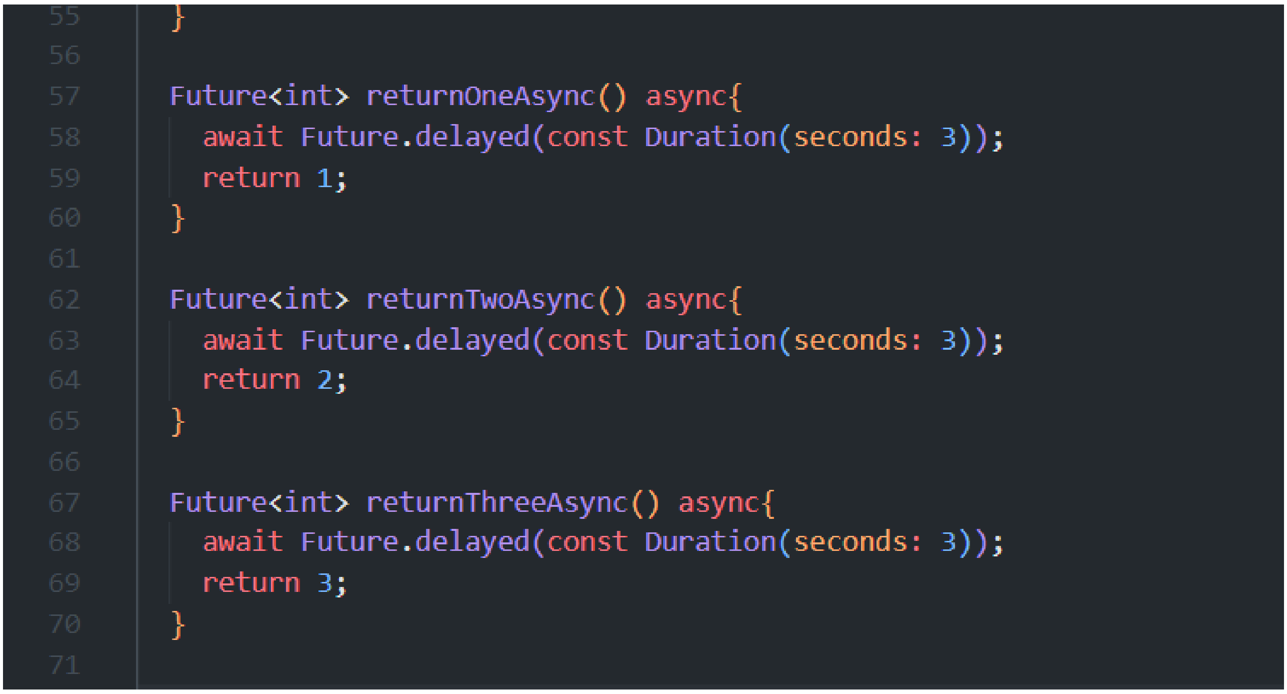
> substring : digunakan untuk mengambil string dengan indeks dari 0 sampai 450 Catcherror : digunakan untuk error handling ketika terjadi kesalahan saat load data
>
> 
>
> Back from the furute



42

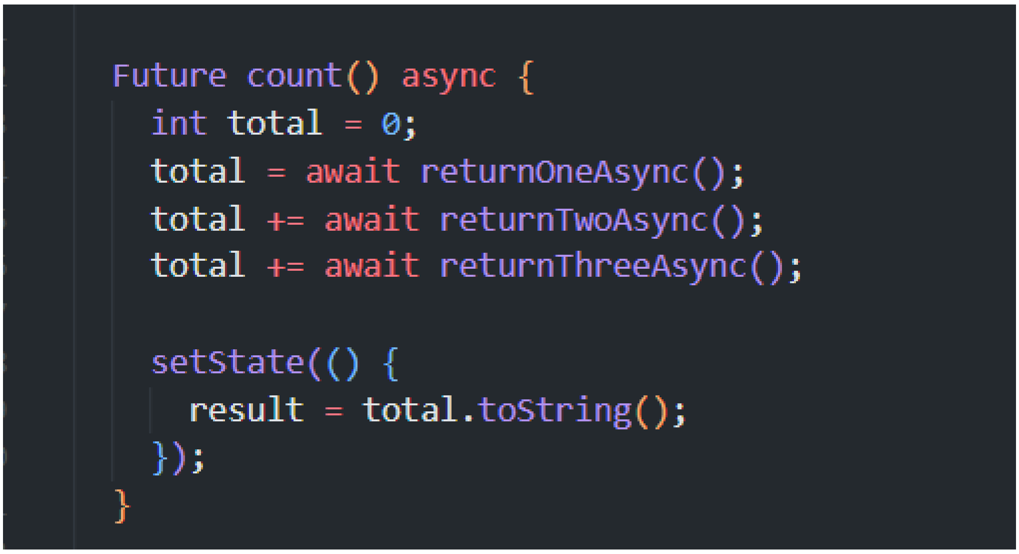
Praktikum 2

1. Buka file main.dart



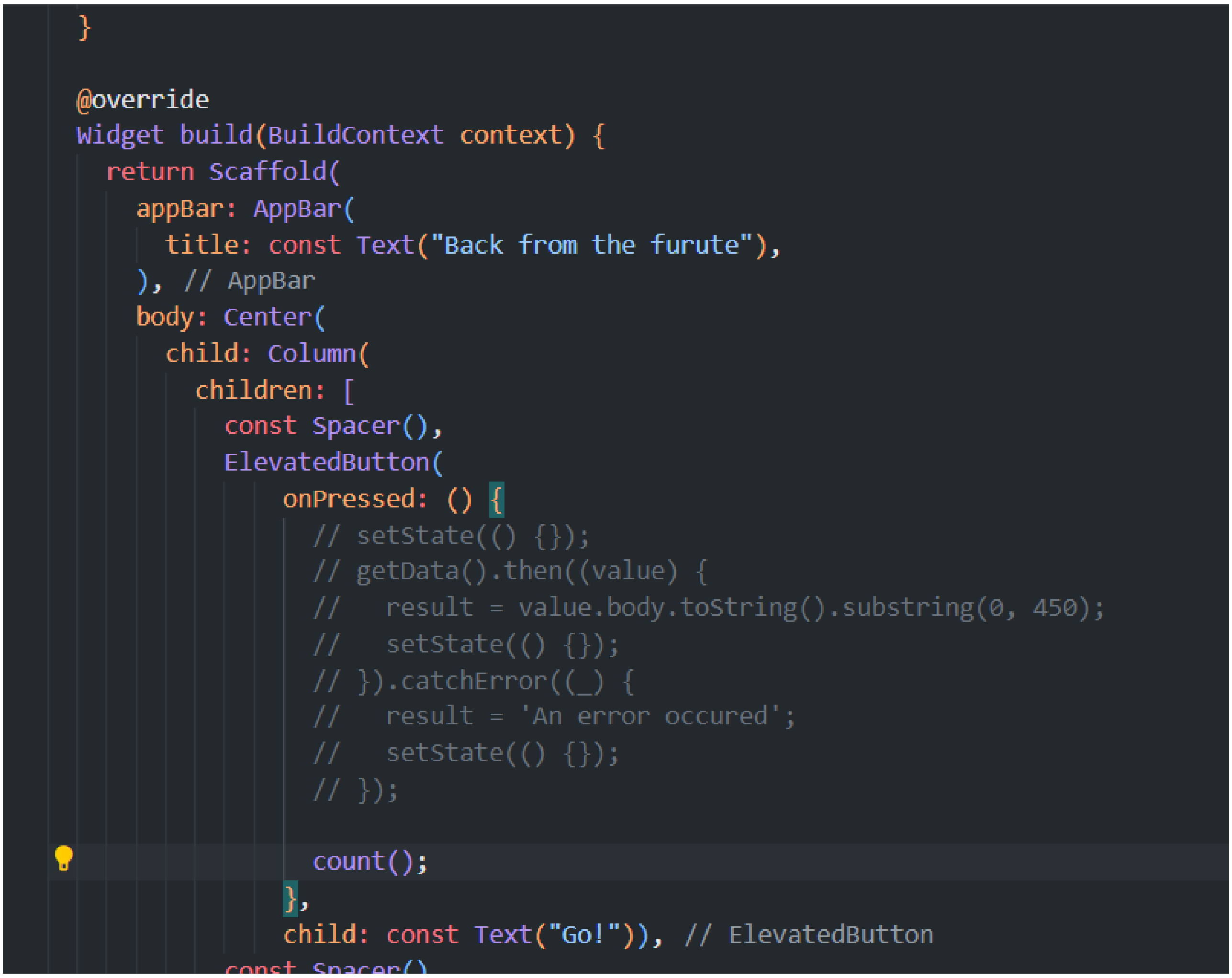
```
  }

  Future<int> returnOneAsync() async{
    await Future.delayed(const Duration(seconds: 3));
    return 1;
  }

  Future<int> returnTwoAsync() async{
    await Future.delayed(const Duration(seconds: 3));
    return 2;
  }

  Future<int> returnThreeAsync() async{
    await Future.delayed(const Duration(seconds: 3));
    return 3;
  }

```

2. Tambah method count()

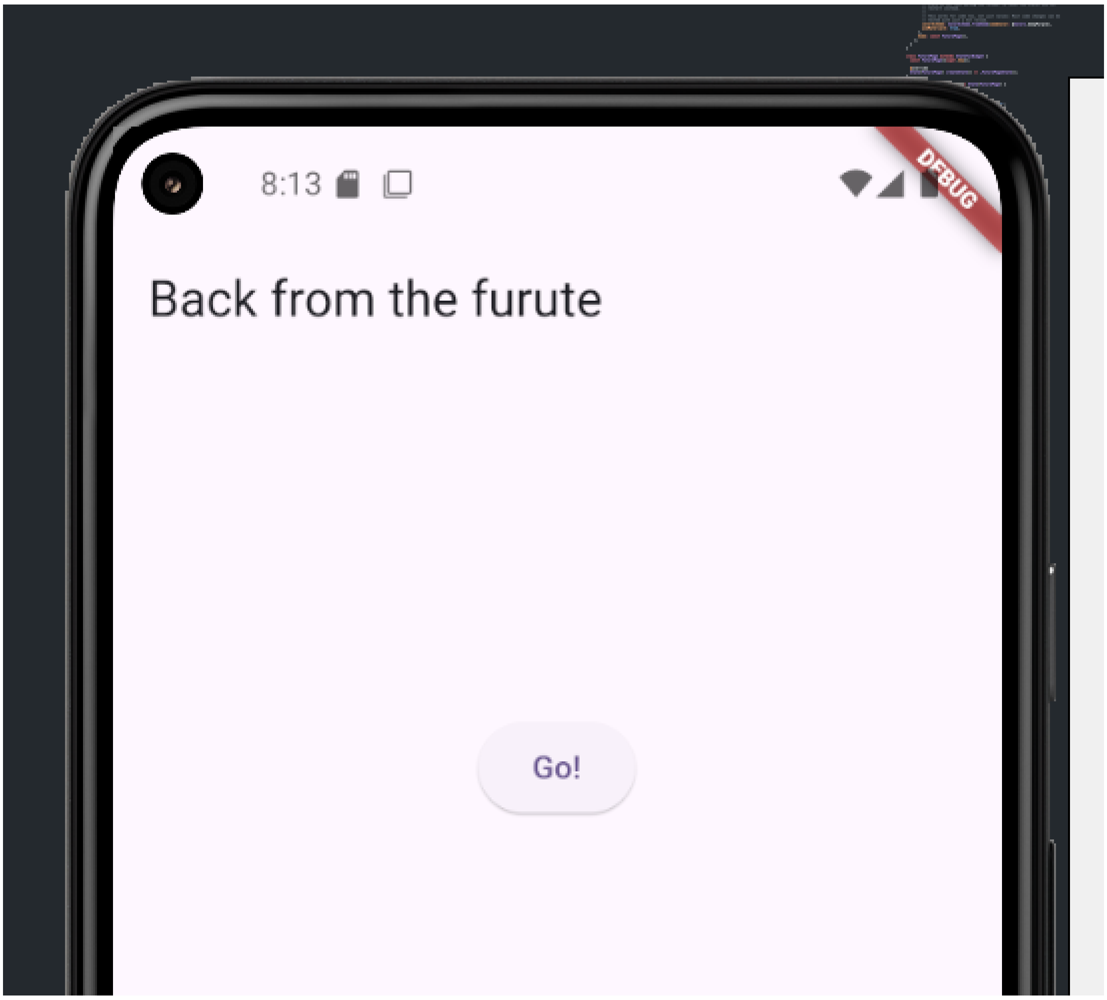


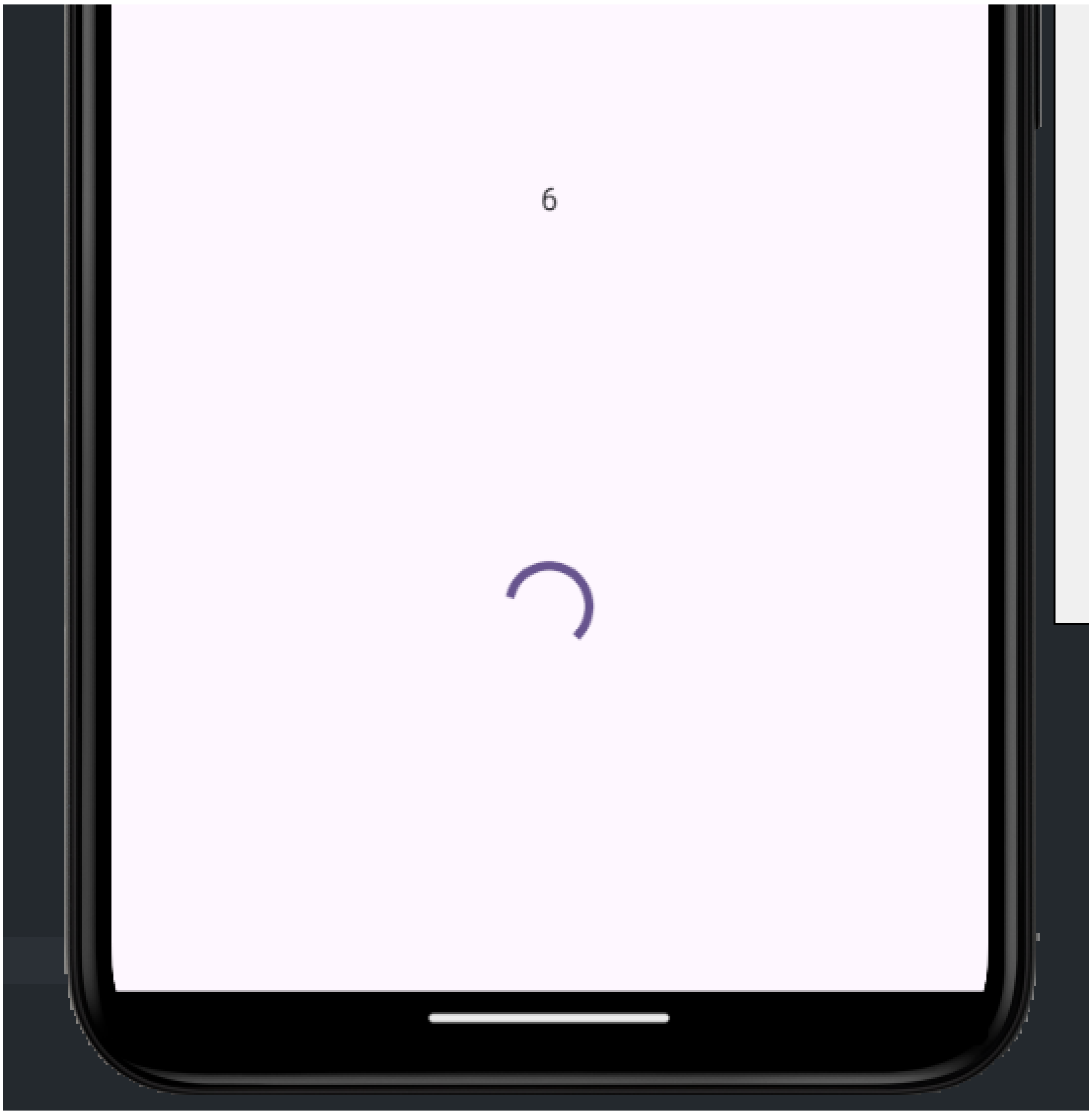
```
  Future count() async {
    int total = 0;
    total = await returnOneAsync();
    total += await returnTwoAsync();
    total += await returnThreeAsync();

    setState(() {
      result = total.toString();
    });
  }
```

3. Panggil count()



```
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text("Back from the furute"),
    ), // AppBar
    body: Center(
      child: Column(
        children: [
          const Spacer(),
          ElevatedButton(
              onPressed: () {
                // setState(() {});
                // getData().then((value) {
                //   result = value.body.toString().substring(0, 450);
                //   setState(() {});
                // }).catchError((_) {
                //   result = 'An error occured';
                //   setState(() {});
                // });

                count();
              },
              child: const Text("Go!")), // ElevatedButton
          const Spacer()
```

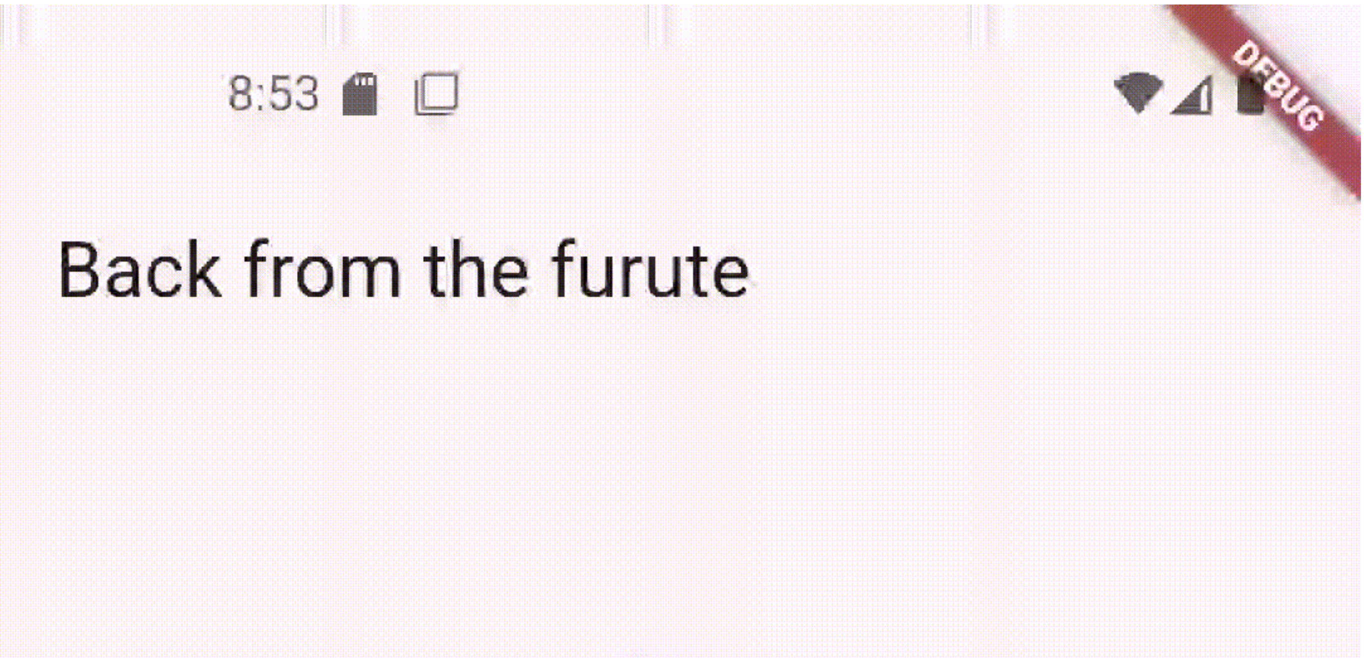


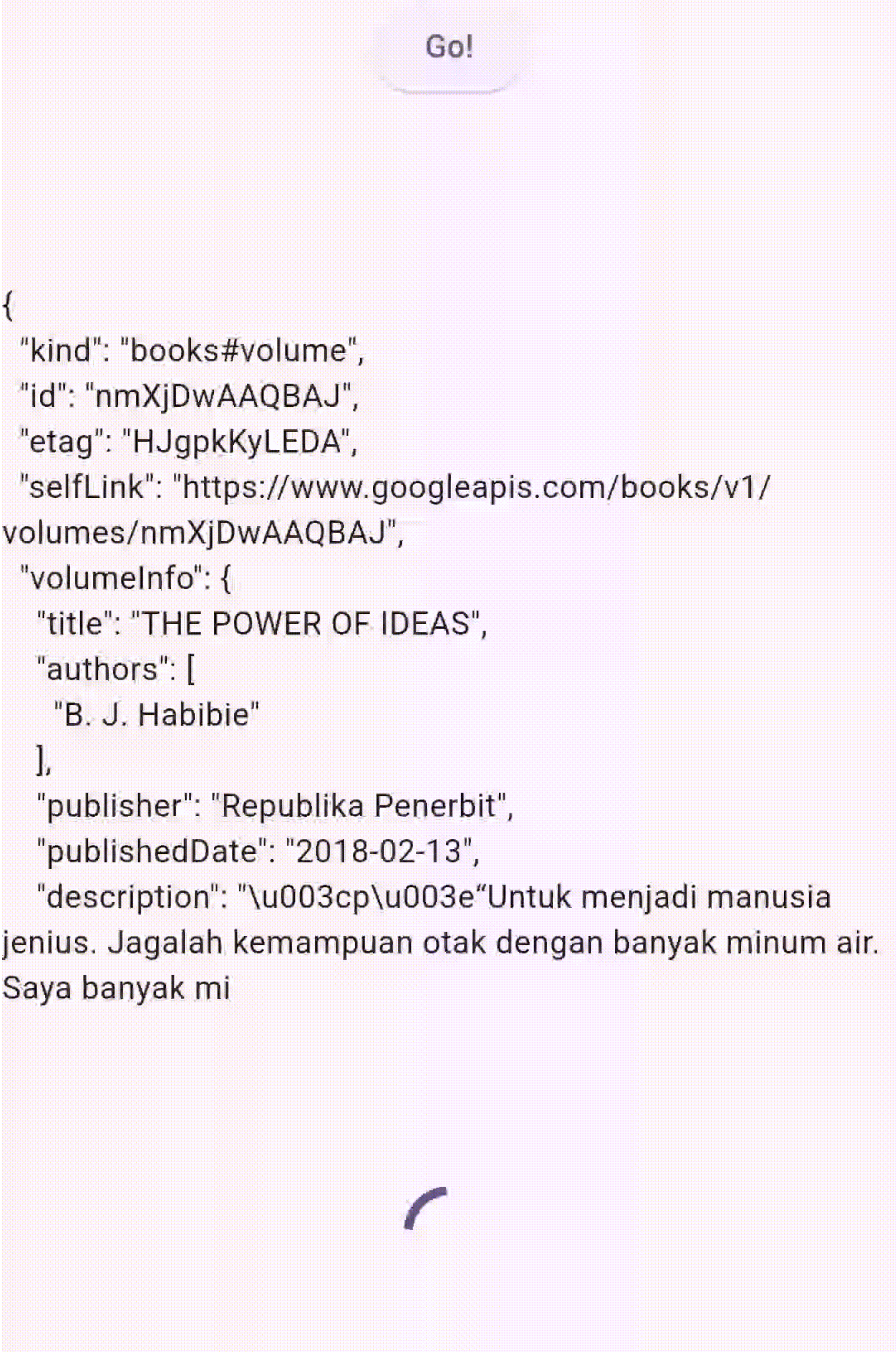


4. Run

Soal 4

> kode dari langkah 1 dan adalah contoh pembuatan function async dengan memberikan delay 3 detik dan mereturn nilai int lalau pada kode langkah 2 membuat fungsi async yang menunggu dari fungsi async sebelumnya lalu menjumlahkannya dan mengatur state

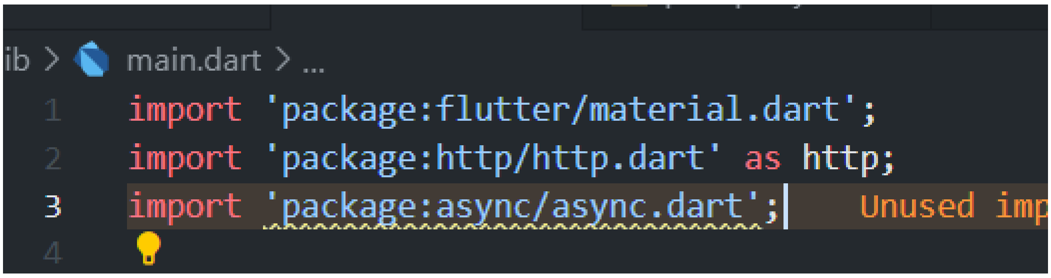




```
{
  "kind": "books#volume",
  "id": "nmXjDwAAQBAJ",
  "etag": "HJgpkKyLEDA",
  "selfLink": "https://www.googleapis.com/books/v1/
volumes/nmXjDwAAQBAJ",
  "volumeInfo": {
    "title": "THE POWER OF IDEAS",
    "authors": [
      "B. J. Habibie"
    ],
    "publisher": "Republika Penerbit",
    "publishedDate": "2018-02-13",
    "description": "\u003cp\u003e“Untuk menjadi manusia
jenius. Jagalah kemampuan otak dengan banyak minum air.
Saya banyak mi
```

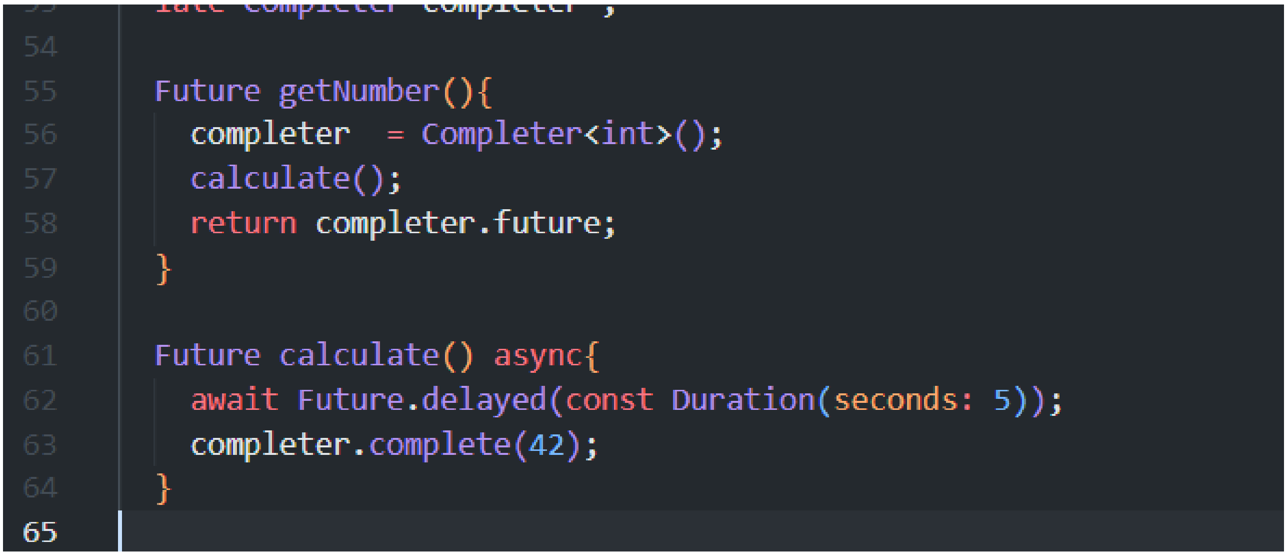
Praktikum 3

1. buka main.dart



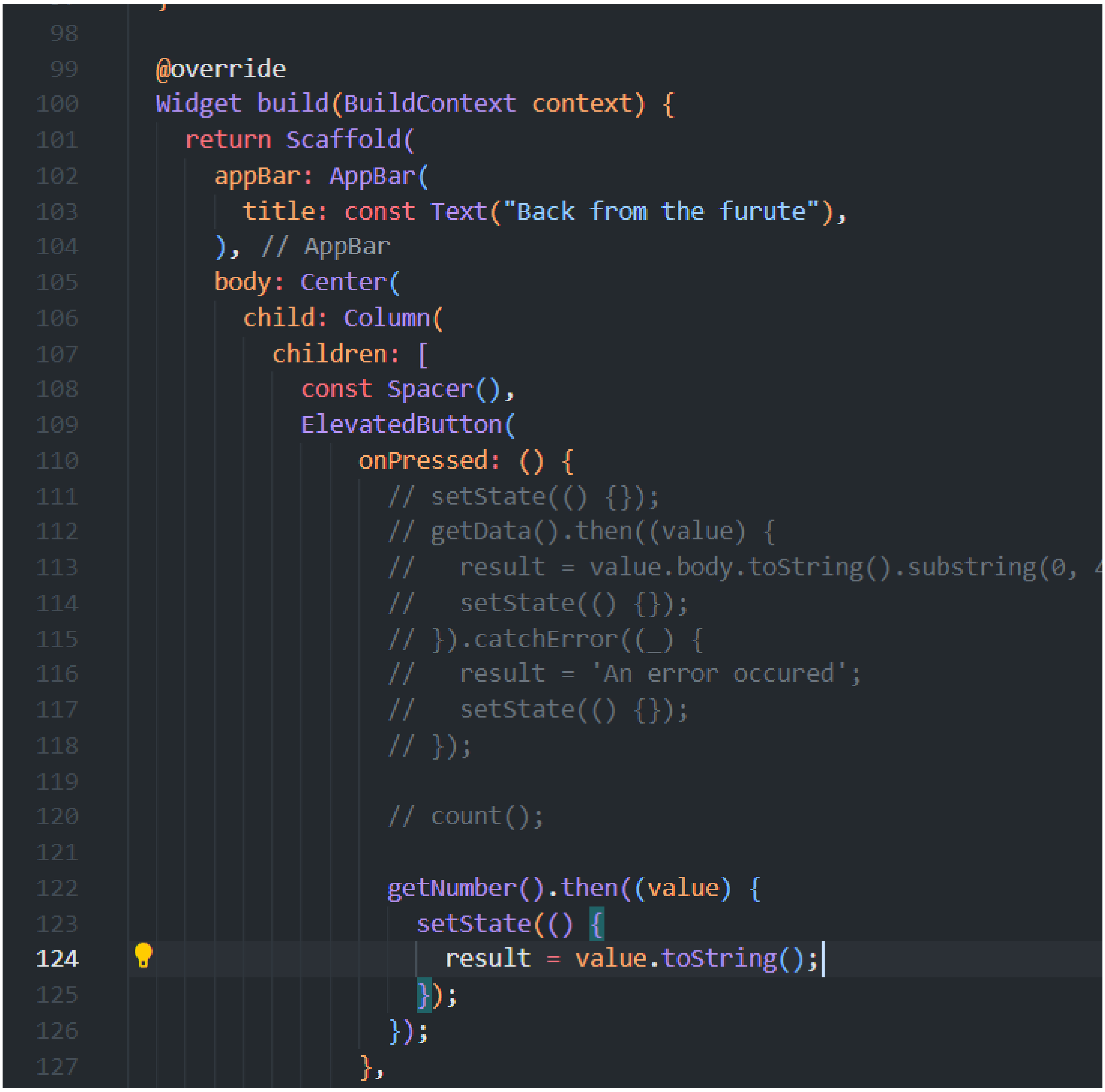
```dart
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'package:async/async.dart';
```

2. tambahkan variabel dan method

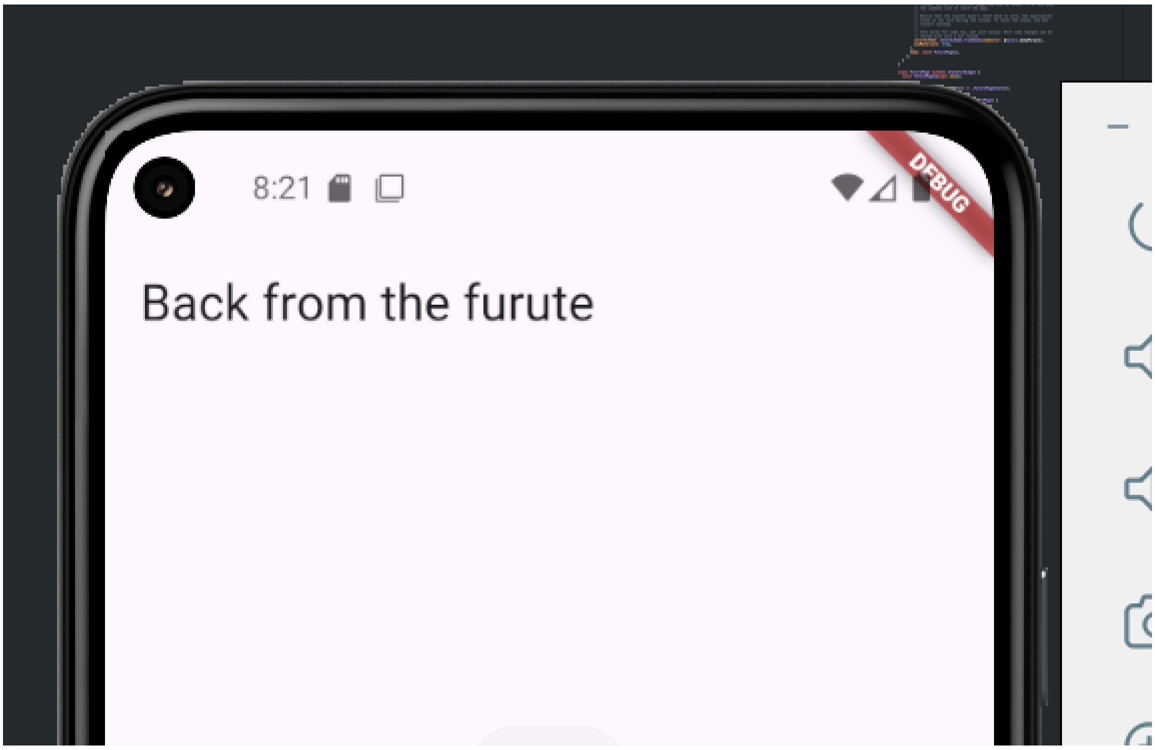


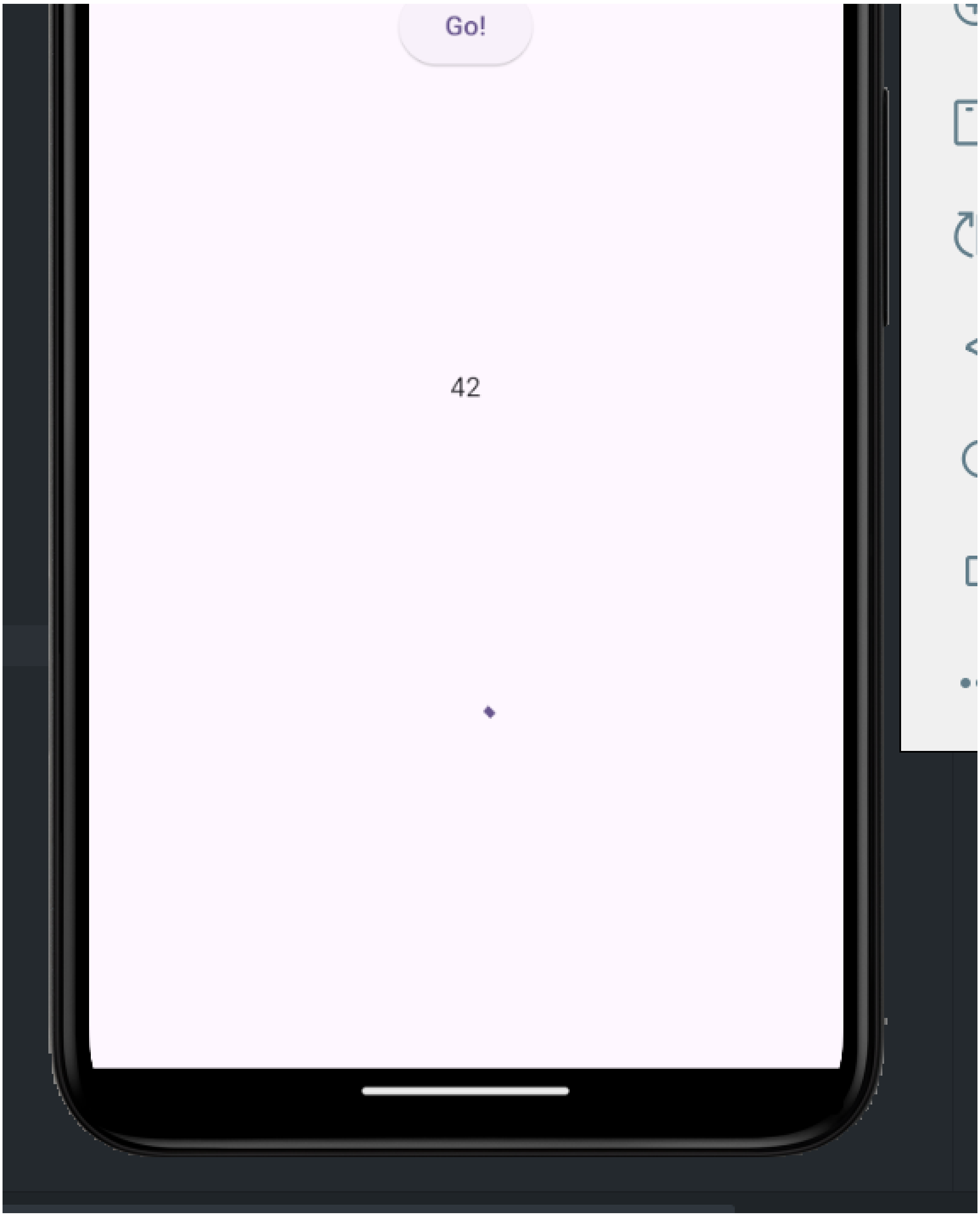
```dart
late Completer completer;

Future getNumber(){
  completer  = Completer<int>();
  calculate();
  return completer.future;
}

Future calculate() async{
  await Future.delayed(const Duration(seconds: 5));
  completer.complete(42);
}

```

3. Ganti isi kode onPressed()

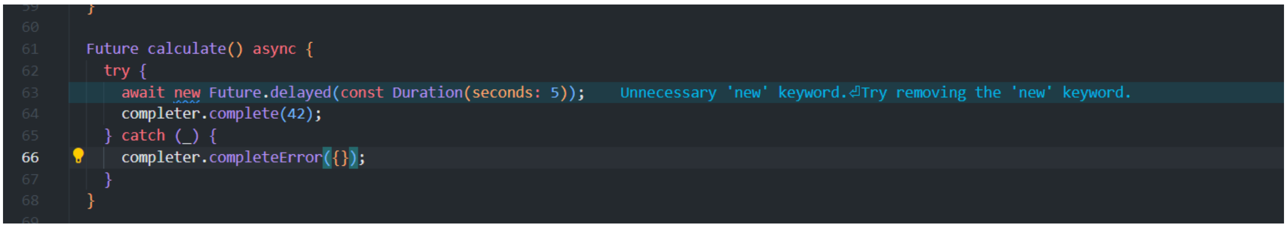


```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text("Back from the furute"),
    ), // AppBar
    body: Center(
      child: Column(
        children: [
          const Spacer(),
          ElevatedButton(
            onPressed: () {
              // setState(() {});
              // getData().then((value) {
              //   result = value.body.toString().substring(0,
              //   setState(() {});
              // }).catchError((_) {
              //   result = 'An error occured';
              //   setState(() {});
              // });

              // count();

              getNumber().then((value) {
                setState(() {
                  result = value.toString();
                });
              });
            },
```





4. Run
5. Ganti method calculate()

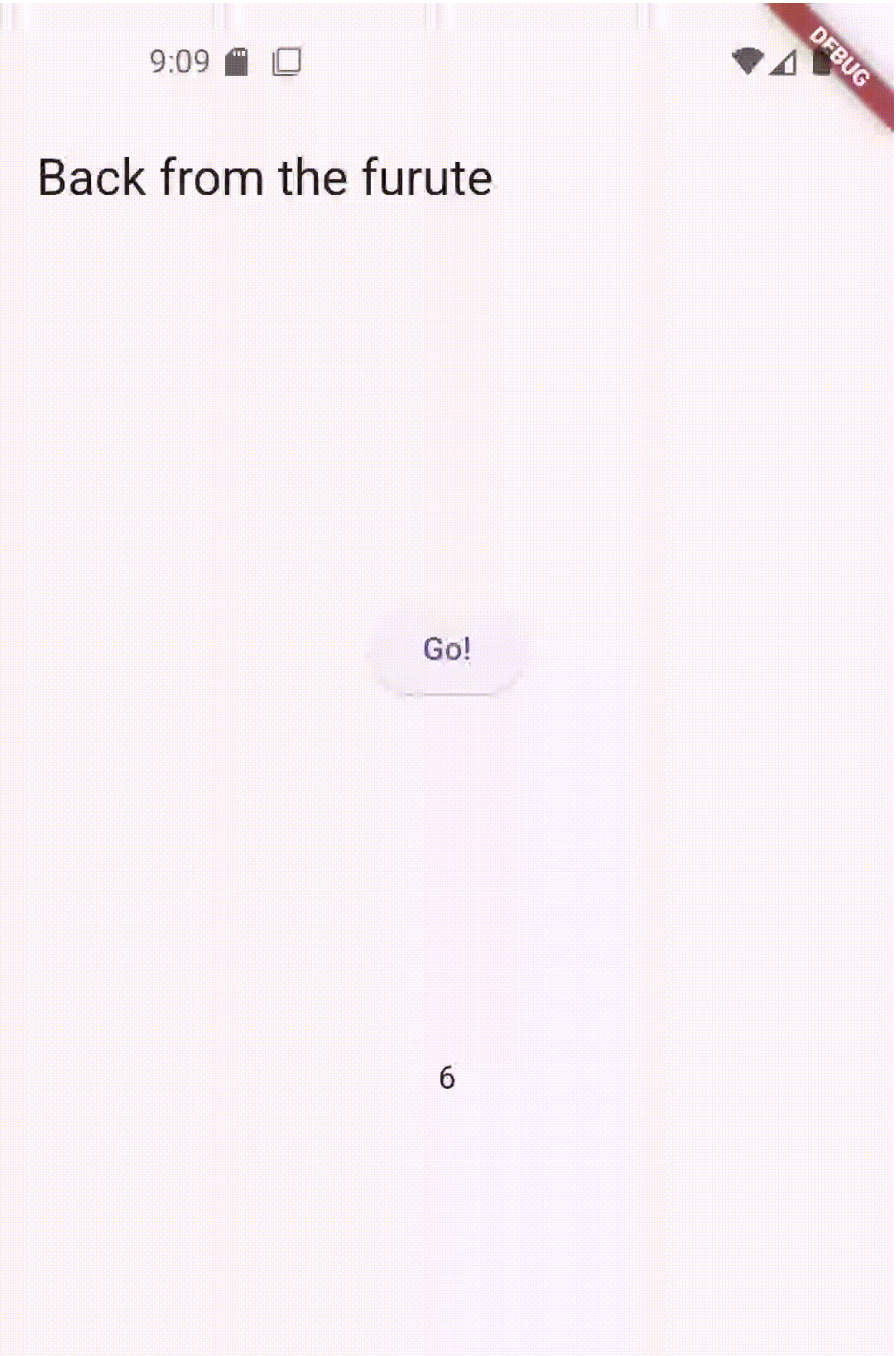


```
  }

  Future calculate() async {
    try {
      await new Future.delayed(const Duration(seconds: 5));    Unnecessary 'new' keyword.⏎Try removing the 'new' keyword.
      completer.complete(42);
    } catch (_) {
      completer.completeError({});
    }
  }
```

6. pindah ke onPressed()

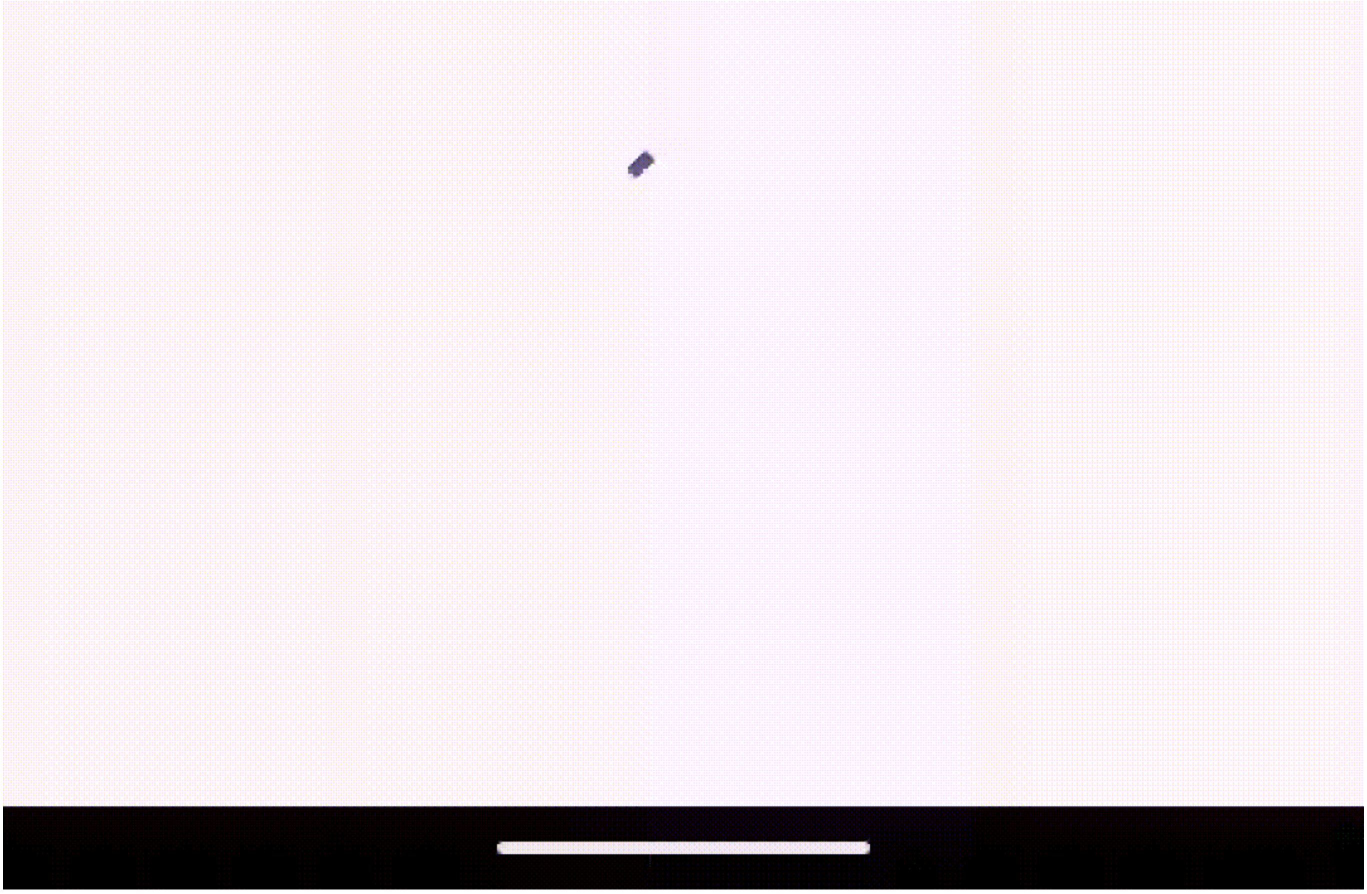
Soal 5

> Pada langkah 2 membuat sebuah variabel dengan tipe data Completer dan membuat fungsi future getNumber yang menjalankan fungsi future calculate dengan delayed 5 detik dan mereturn nilai complete 42. Dengan menggunakan completer maka bisa mereturn nilai value jika berhasil atau error jika gagal



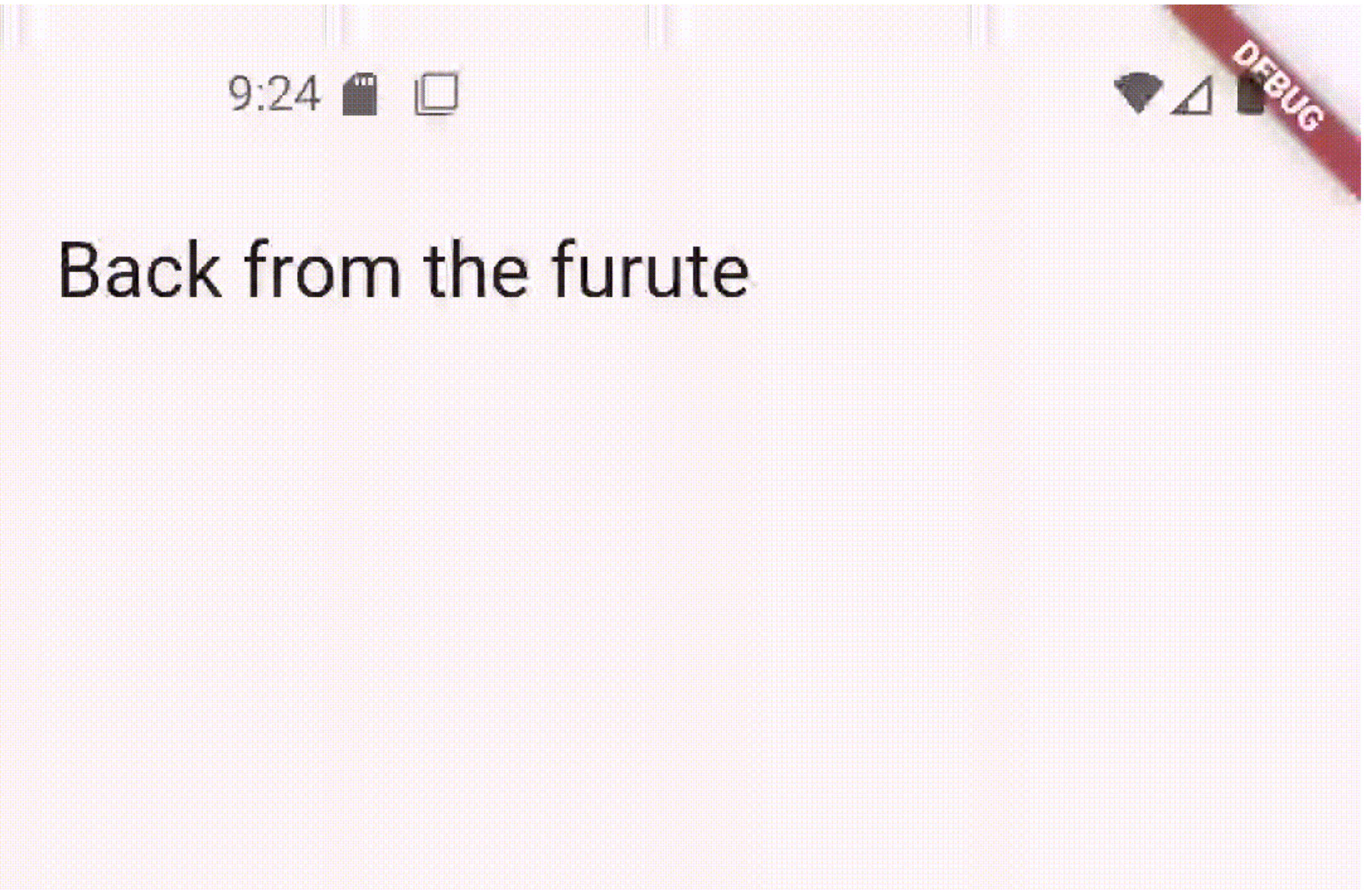
Back from the furute

6

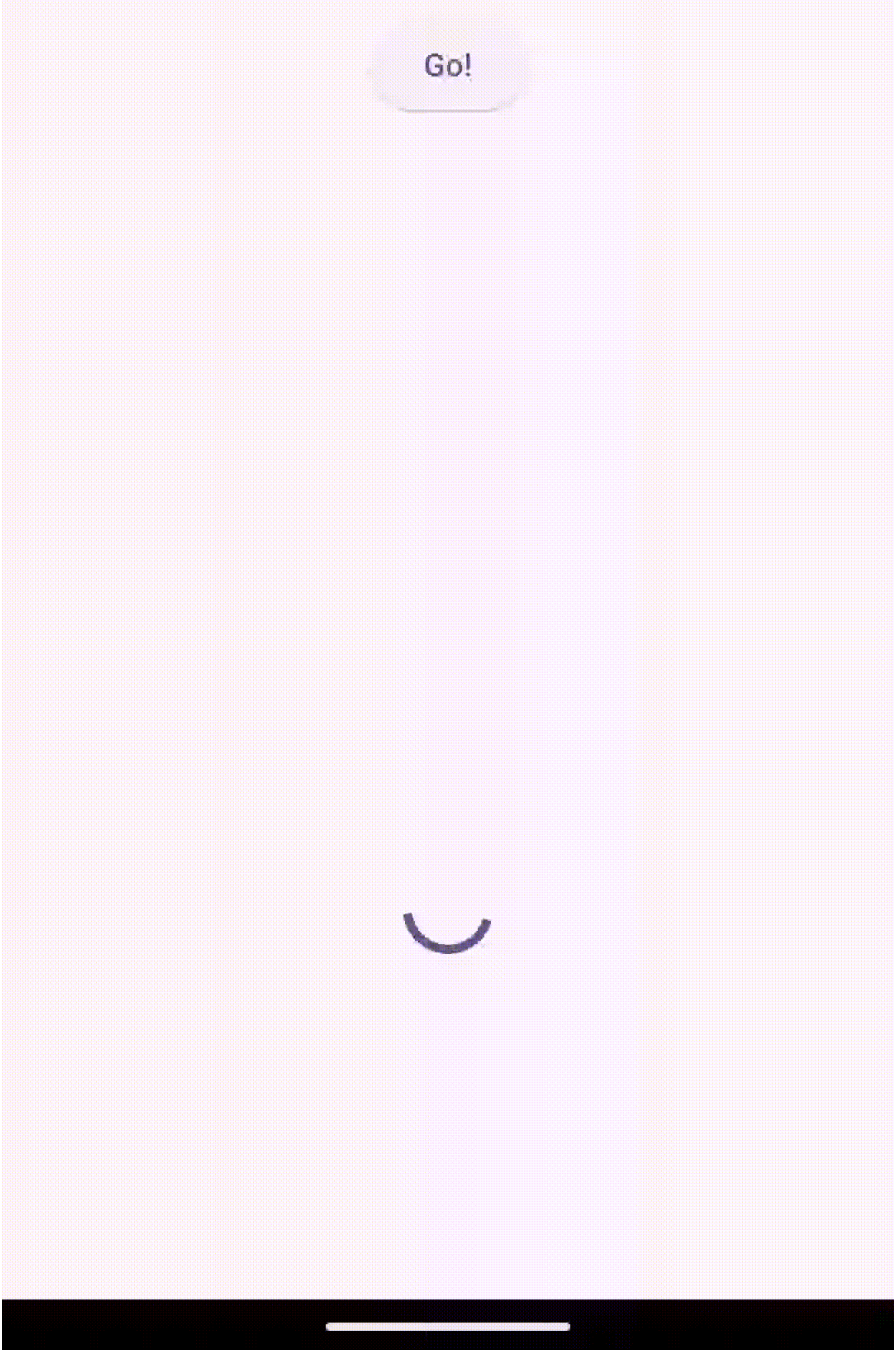


Soal 6

> perbedaannya terletak pada handling error, jika variabel dengan tipe data Completer dapat mereturn error maka kita harus menambahkan handling error juga agar program memberi flow apabila terjadi error

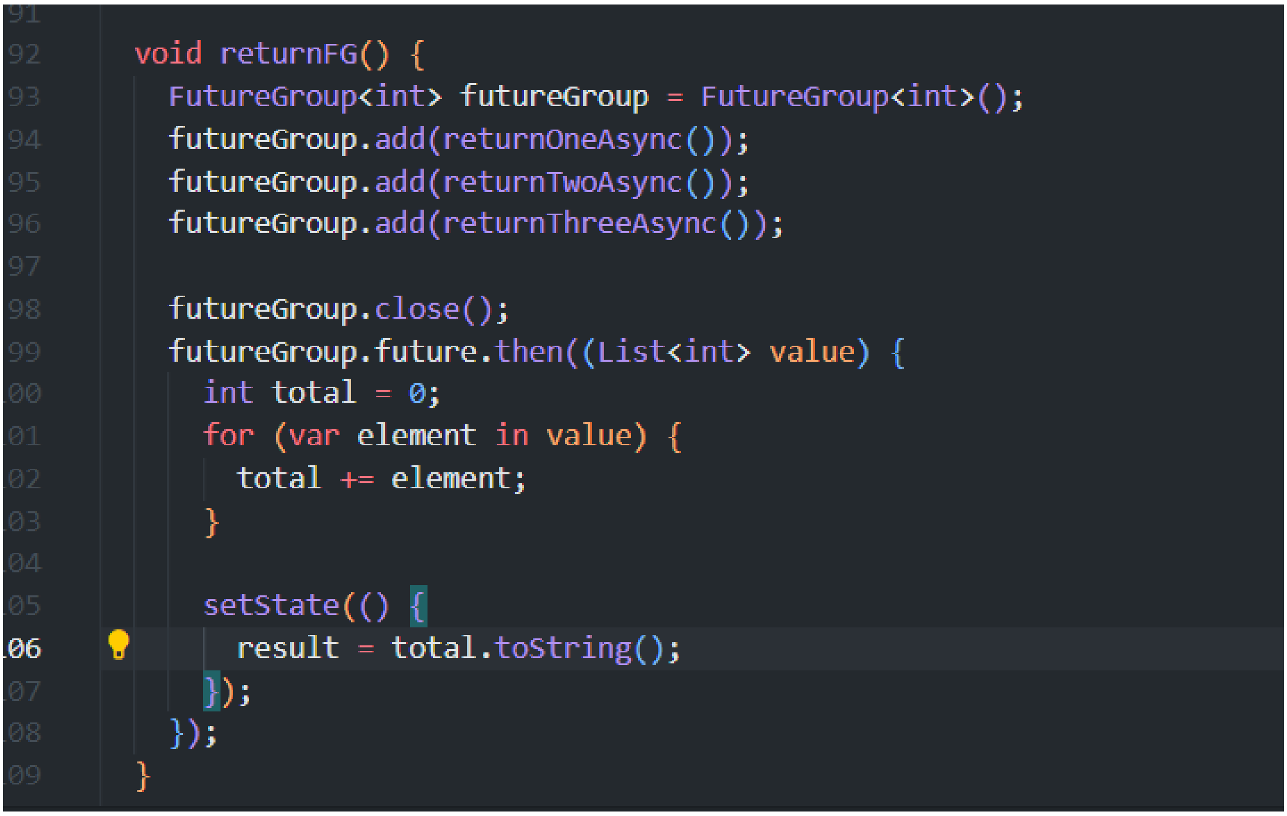


Back from the furute


Go!

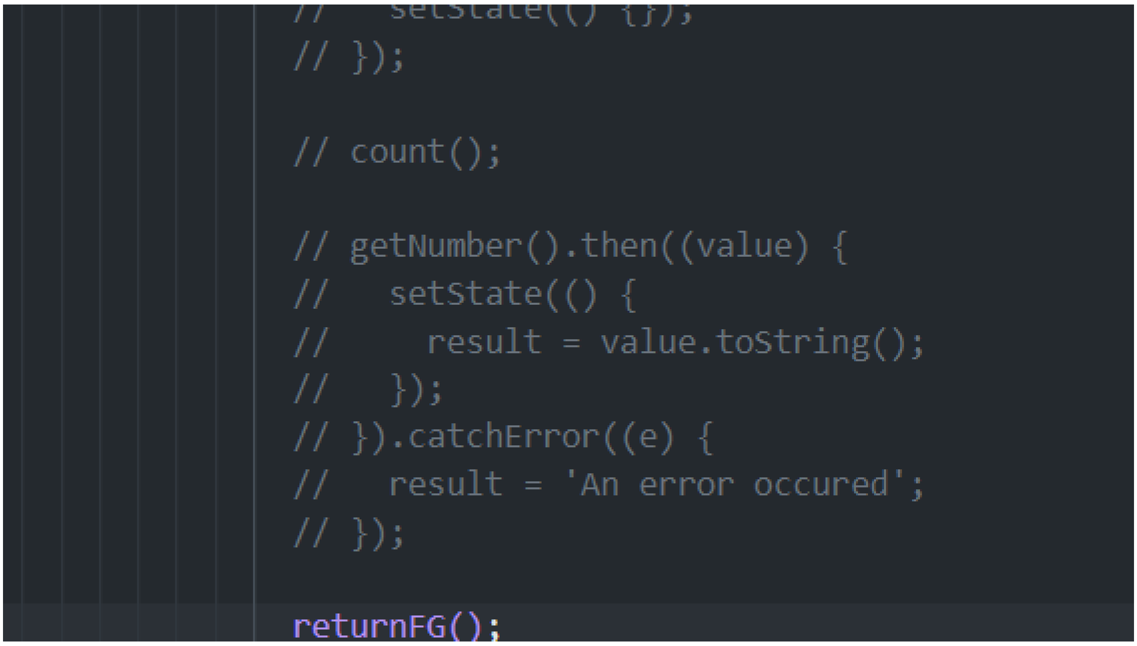
Praktikum 4

1. Buka file main.dart

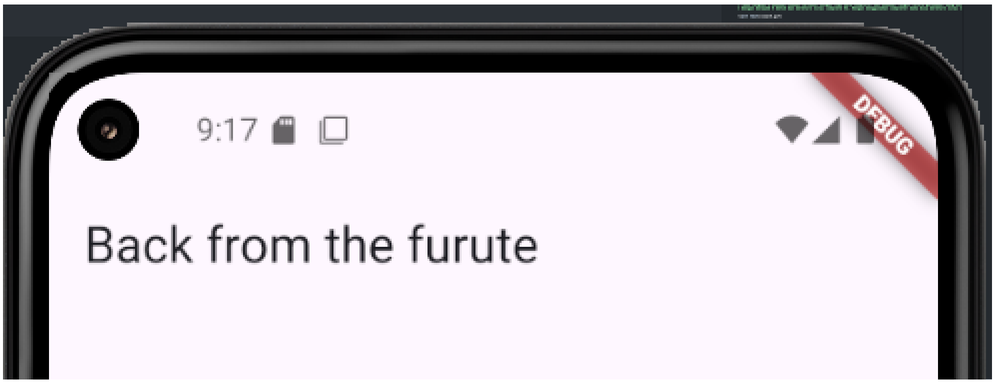


```dart
void returnFG() {
  FutureGroup<int> futureGroup = FutureGroup<int>();
  futureGroup.add(returnOneAsync());
  futureGroup.add(returnTwoAsync());
  futureGroup.add(returnThreeAsync());

  futureGroup.close();
  futureGroup.future.then((List<int> value) {
    int total = 0;
    for (var element in value) {
      total += element;
    }

    setState(() {
      result = total.toString();
    });
  });
}
```

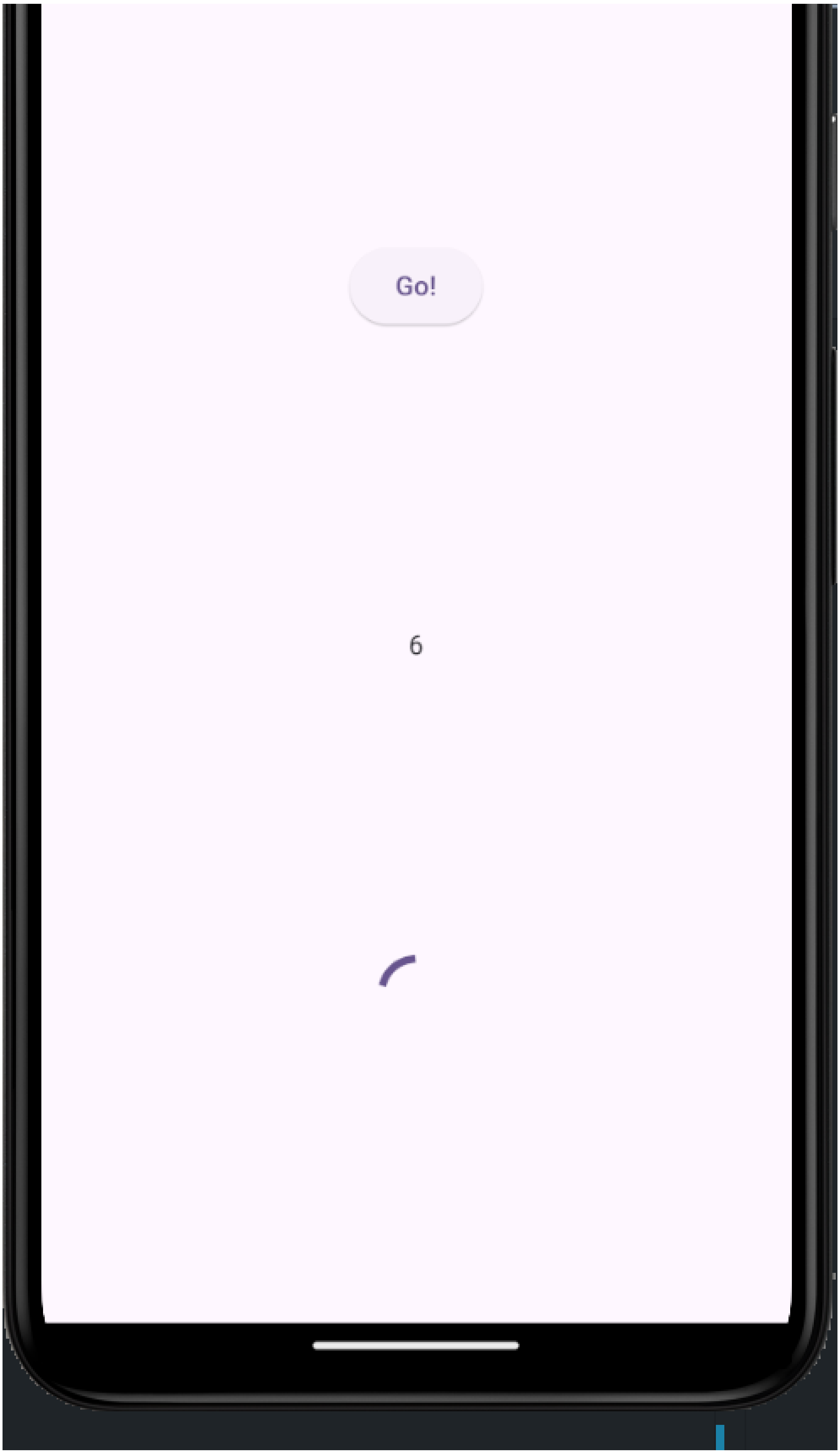
2. Edit onPresssed()



```dart
//   setState(() {});
// });

// count();

// getNumber().then((value) {
//   setState(() {
//     result = value.toString();
//   });
// }).catchError((e) {
//   result = 'An error occured';
// });

returnFG();
```

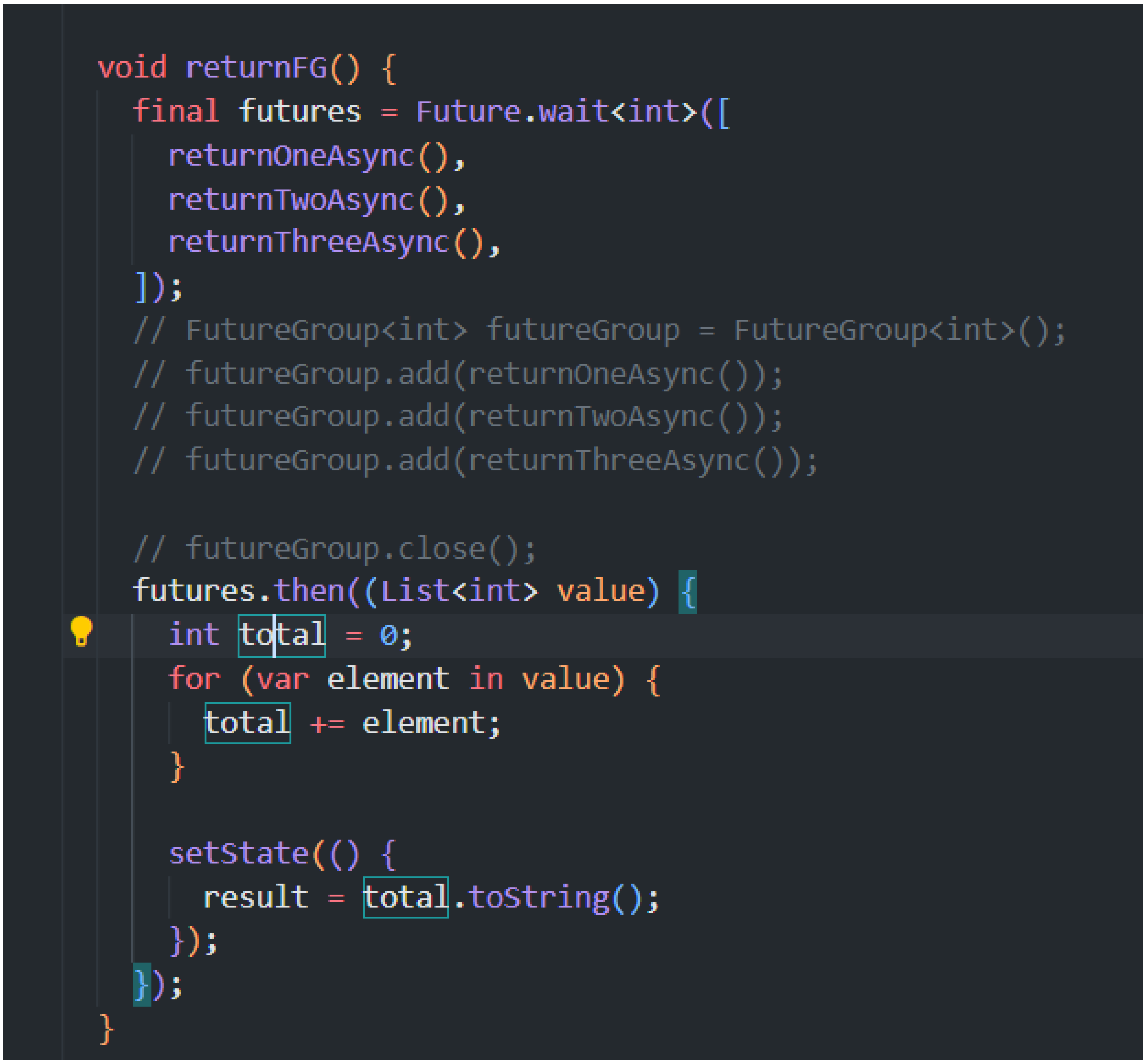


Back from the furute



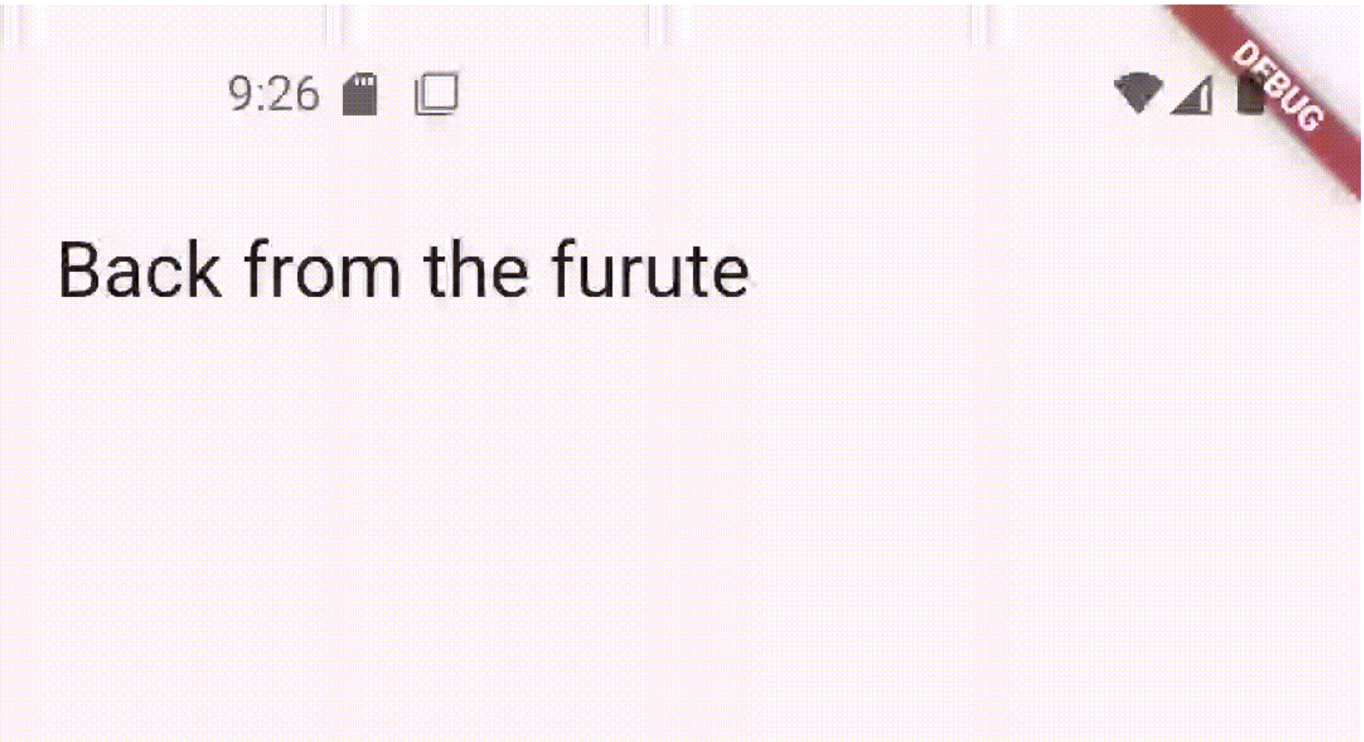
3. run

4. Ganti variabel futureGroup

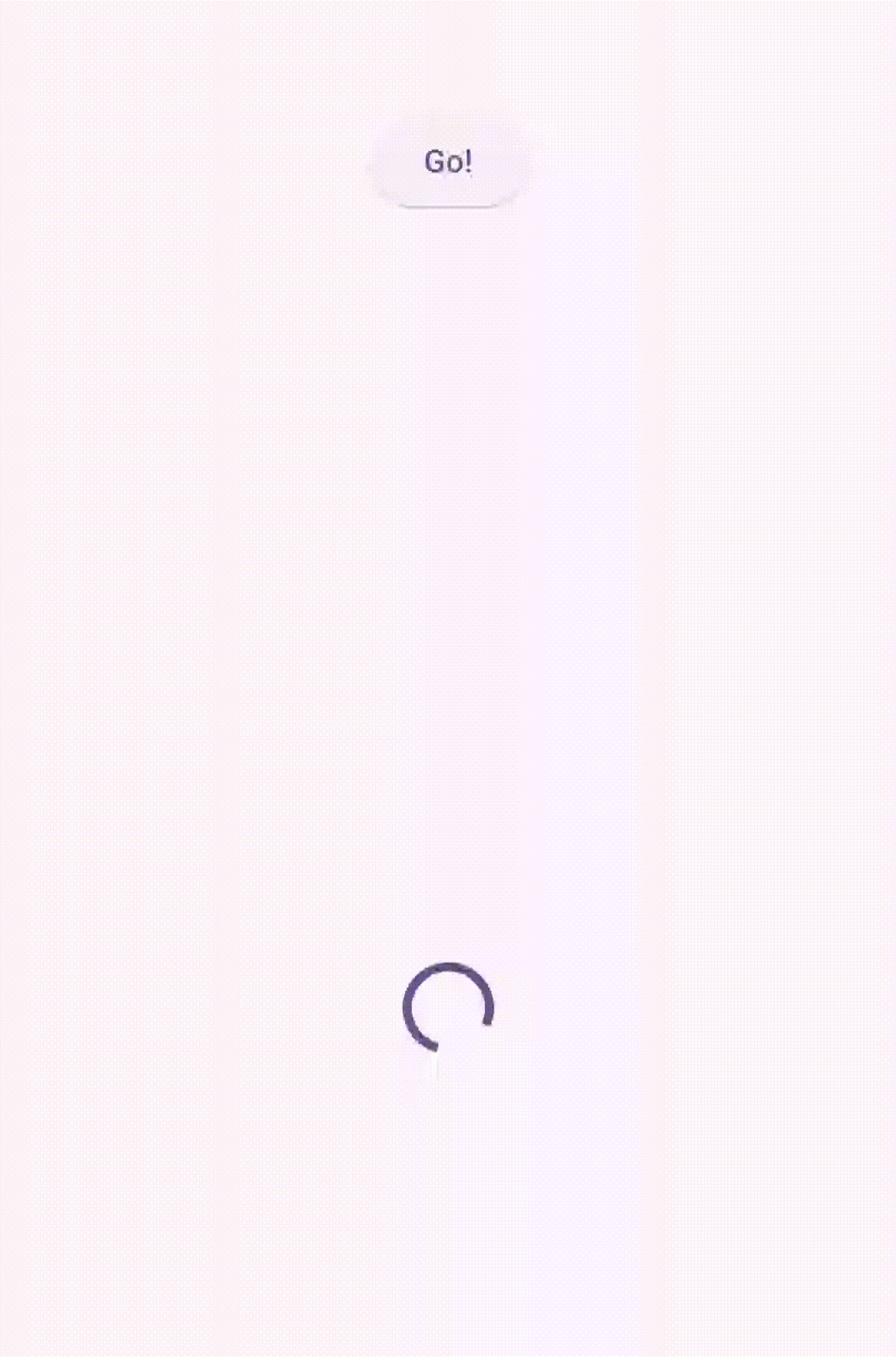


```
void returnFG() {
  final futures = Future.wait<int>([
    returnOneAsync(),
    returnTwoAsync(),
    returnThreeAsync(),
  ]);
  // FutureGroup<int> futureGroup = FutureGroup<int>();
  // futureGroup.add(returnOneAsync());
  // futureGroup.add(returnTwoAsync());
  // futureGroup.add(returnThreeAsync());

  // futureGroup.close();
  futures.then((List<int> value) {
    int total = 0;
    for (var element in value) {
      total += element;
    }

    setState(() {
      result = total.toString();
    });
  });
}
```

Soal 7



Back from the furute

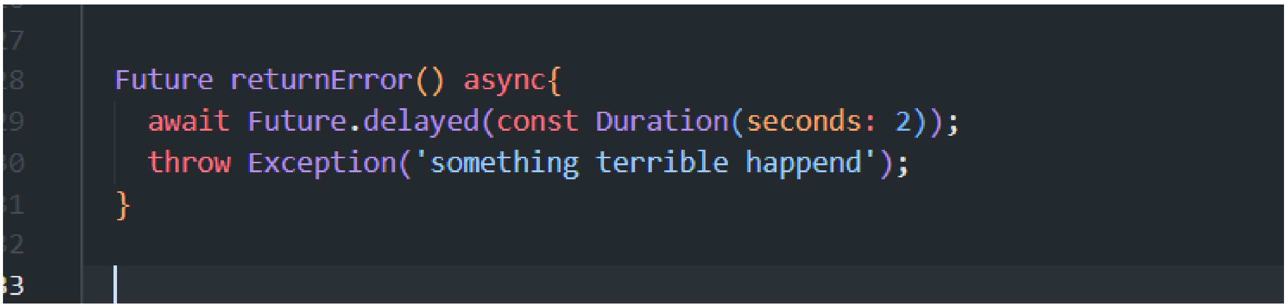

Go!

Soal 8

> Perbedaan langkah 1 dan 4 adalah dalam menghandling beberapa future, pada langkah 1 perlu menggunakan futuregroup dengan tipe datra int dan menambahkannya seperti pada list, dan pada langkah 4 langsung seperti pendeklarasian list dengan isian dari beberapa fungsi async

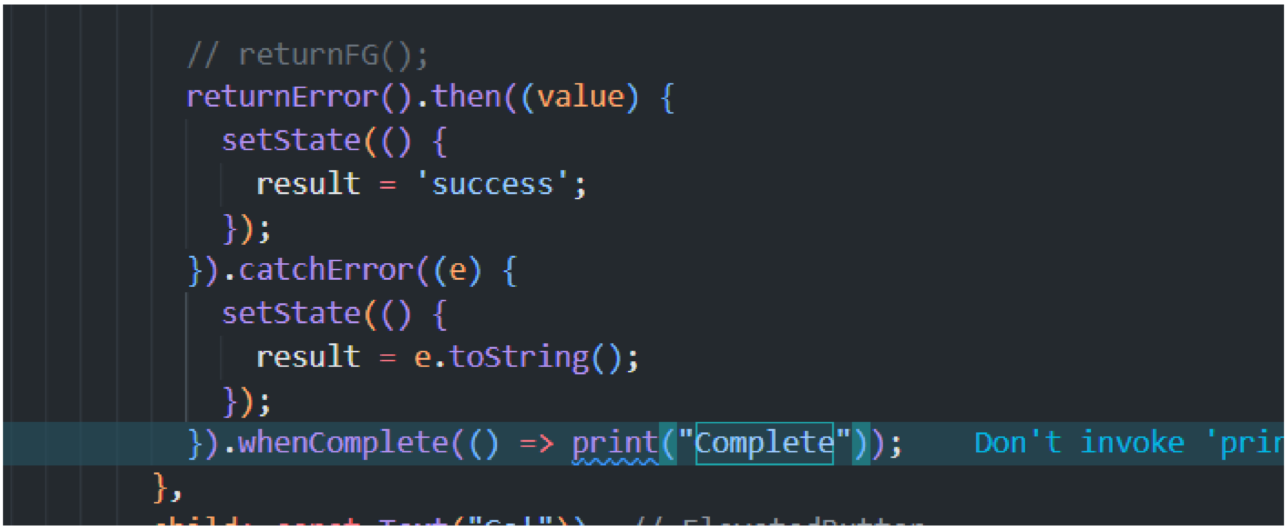
Praktikum 5

1. Buka main.dart

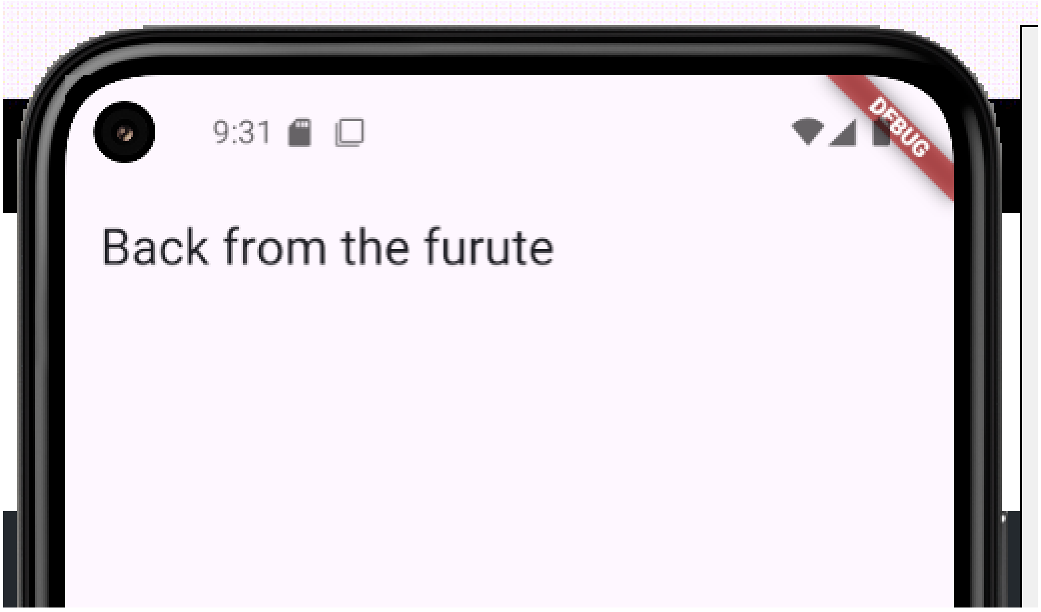


```
Future returnError() async{
  await Future.delayed(const Duration(seconds: 2));
  throw Exception('something terrible happend');
}
```

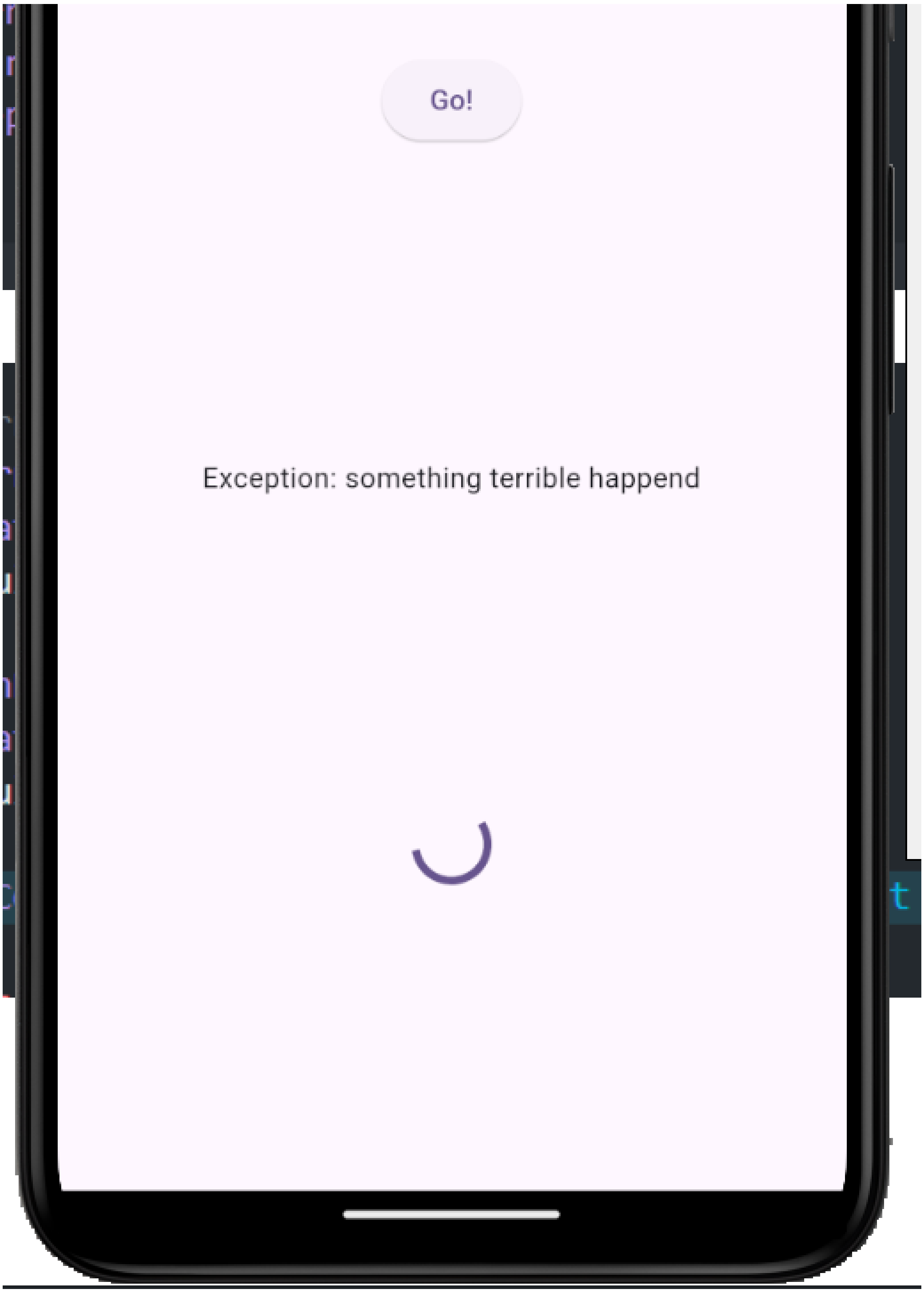
2. Elevated button



```
  // returnFG();
  returnError().then((value) {
    setState(() {
      result = 'success';
    });
  }).catchError((e) {
    setState(() {
      result = e.toString();
    });
  }).whenComplete(() => print("Complete"));    Don't invoke 'prir
},
child: const Text("Go!")), // ElevatedButton
```



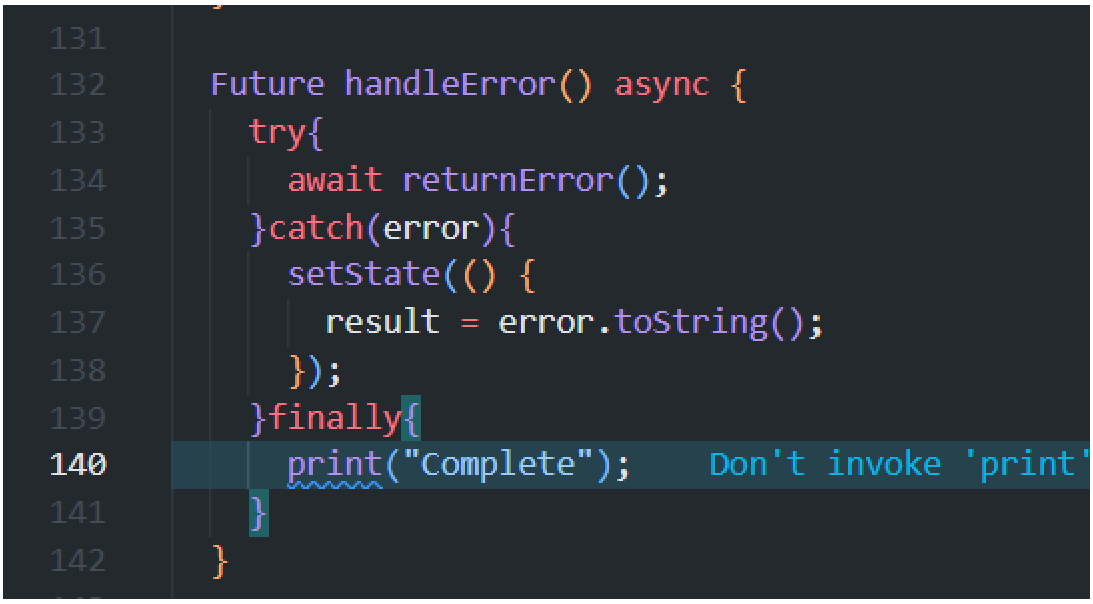
Back from the furute



Exception: something terrible happend

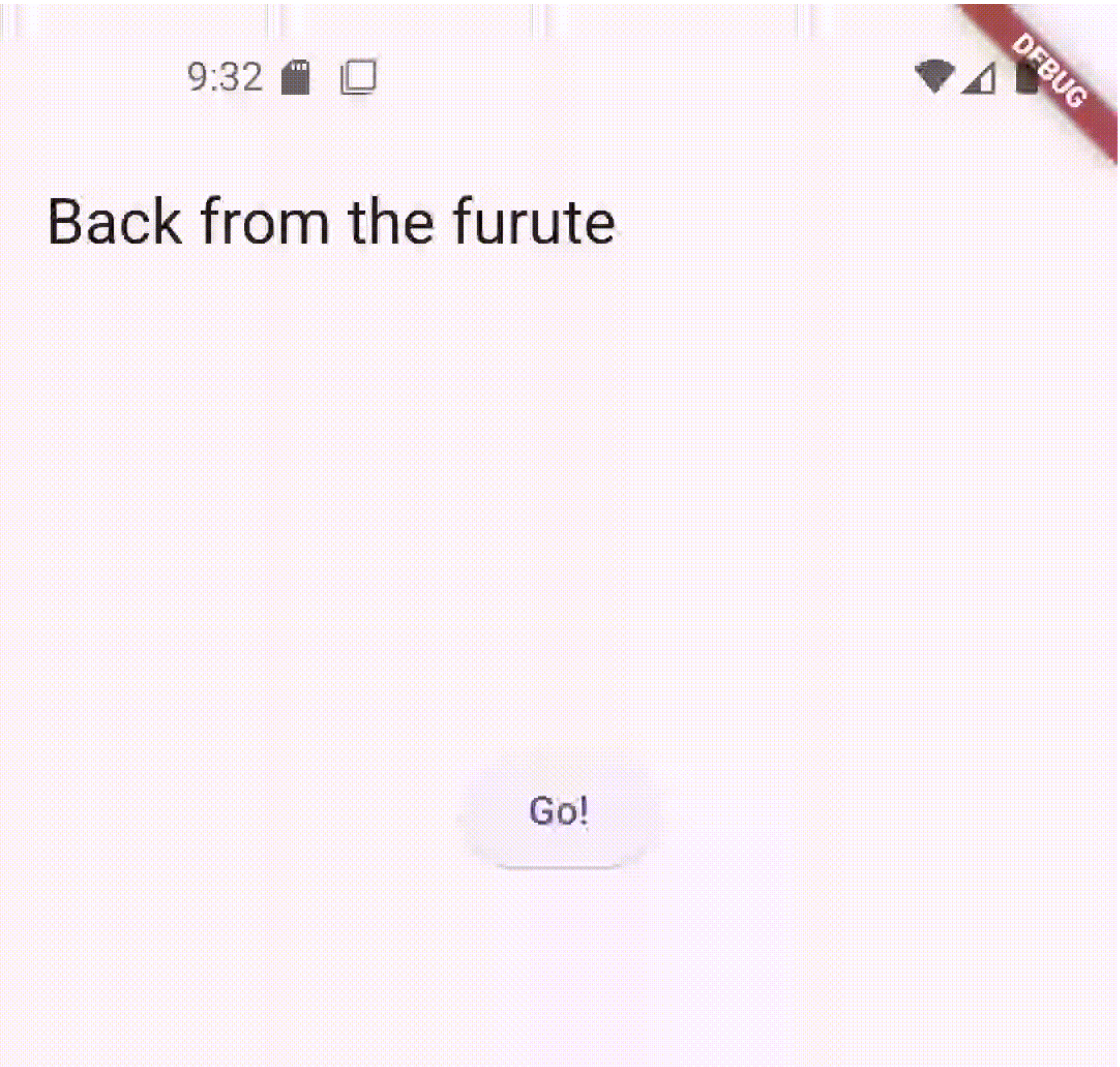
3. Run

4. Tambah method handleError()

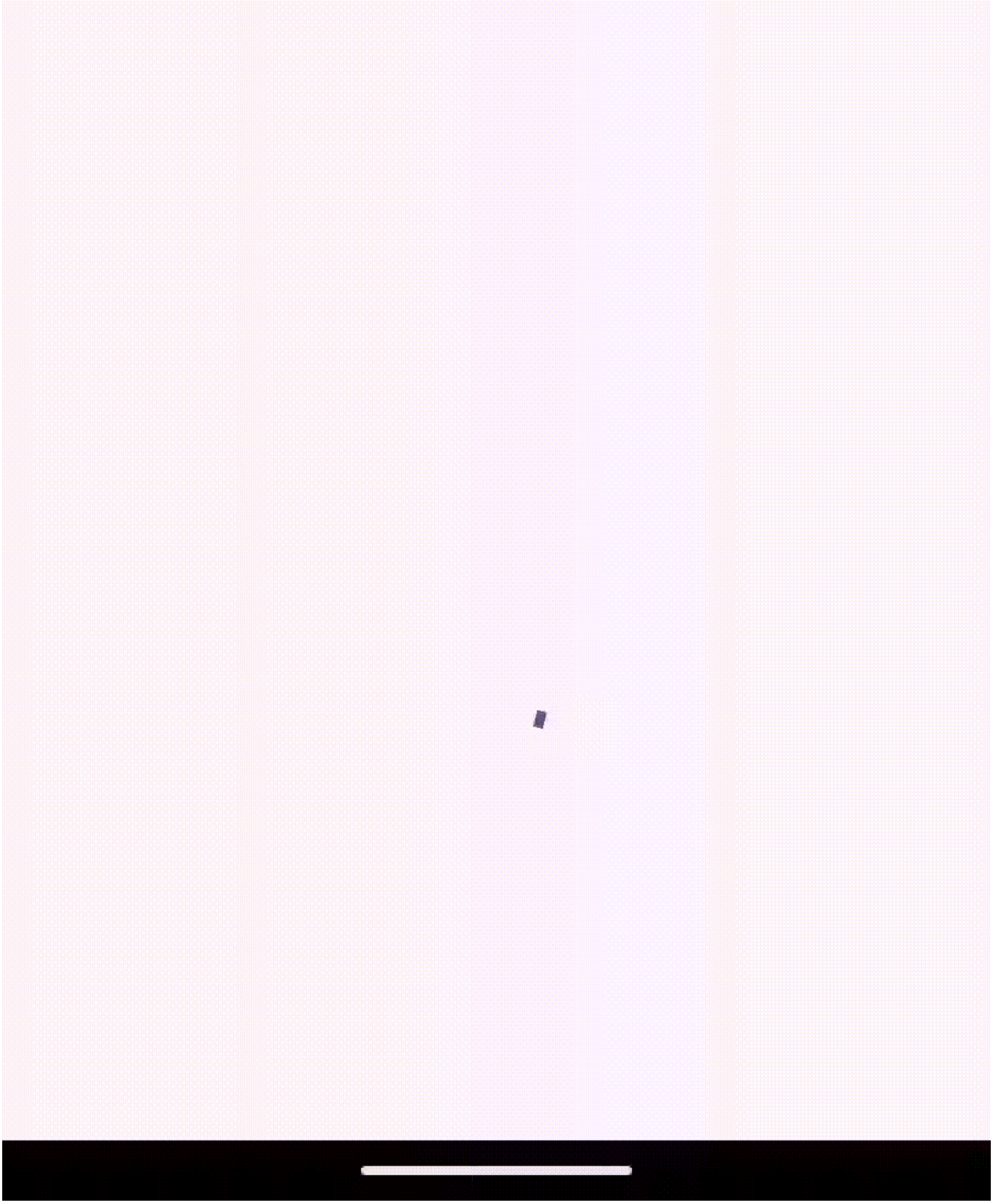


```
Future handleError() async {
  try{
    await returnError();
  }catch(error){
    setState(() {
      result = error.toString();
    });
  }finally{
    print("Complete");     Don't invoke 'print'
  }
}
```

Soal 9

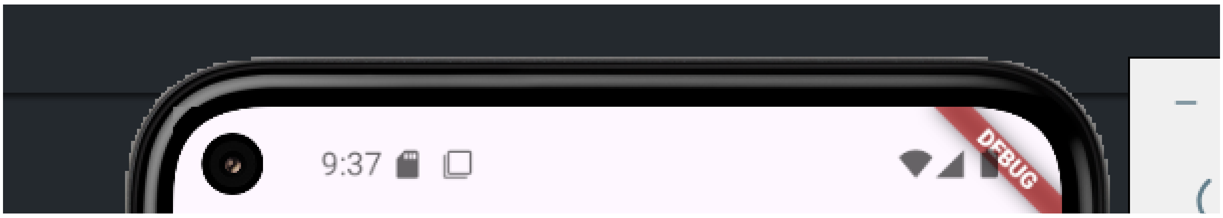


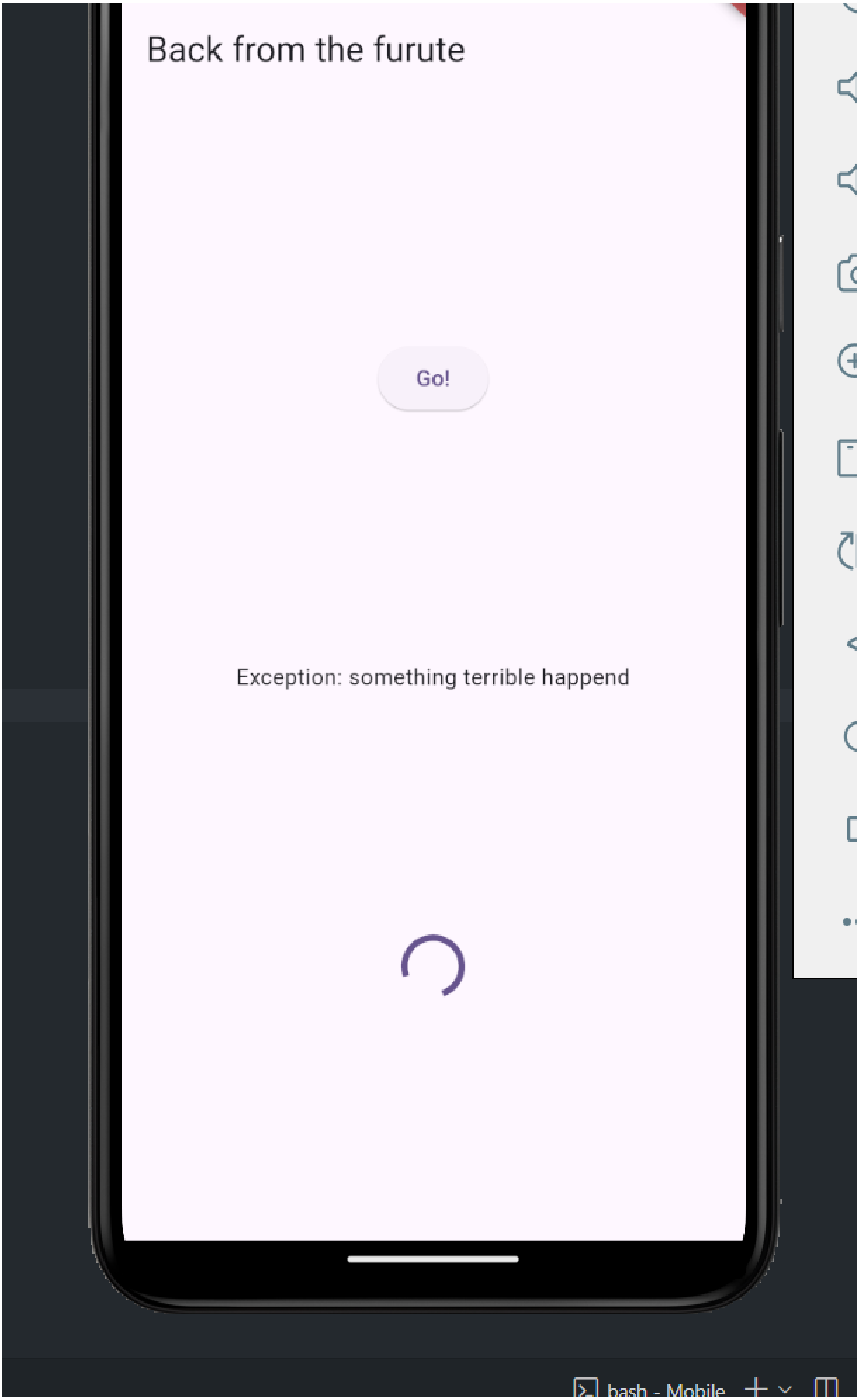
Back from the furute



Soal 10

> Perbedaan pada try catch yang di bungkus dalam sebuah fungsi atau chaining dari async



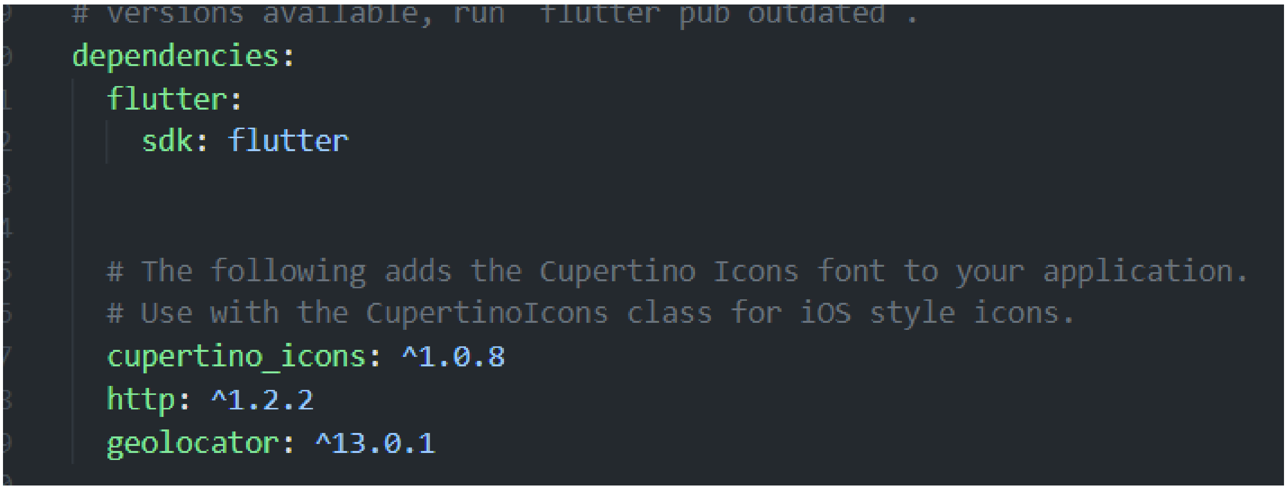


Back from the furute

Exception: something terrible happend

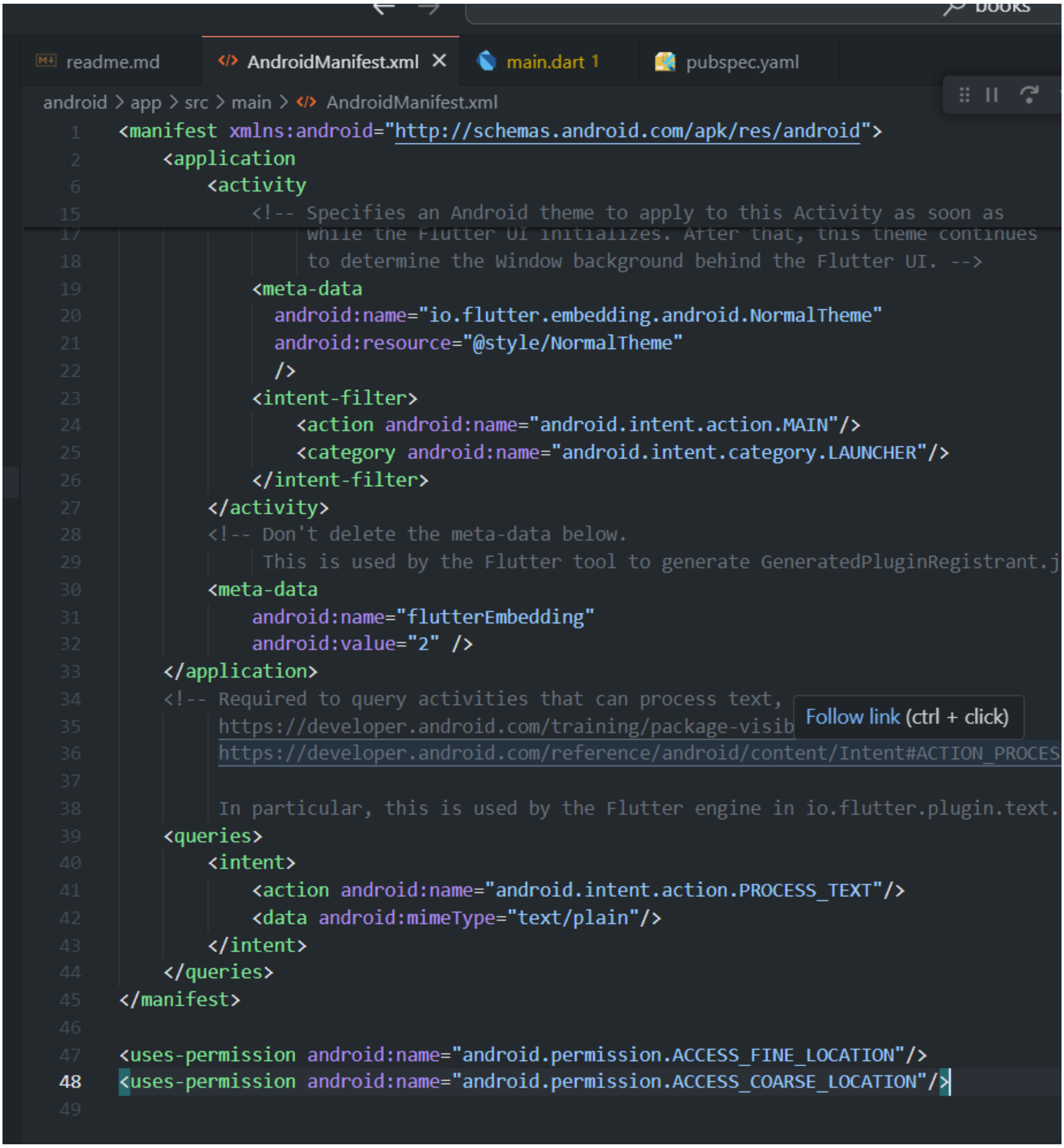
Praktikum 6

1. Install plugin geolocator



```yaml
# versions available, run `flutter pub outdated`.
dependencies:
  flutter:
    sdk: flutter


  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.8
  http: ^1.2.2
  geolocator: ^13.0.1
```

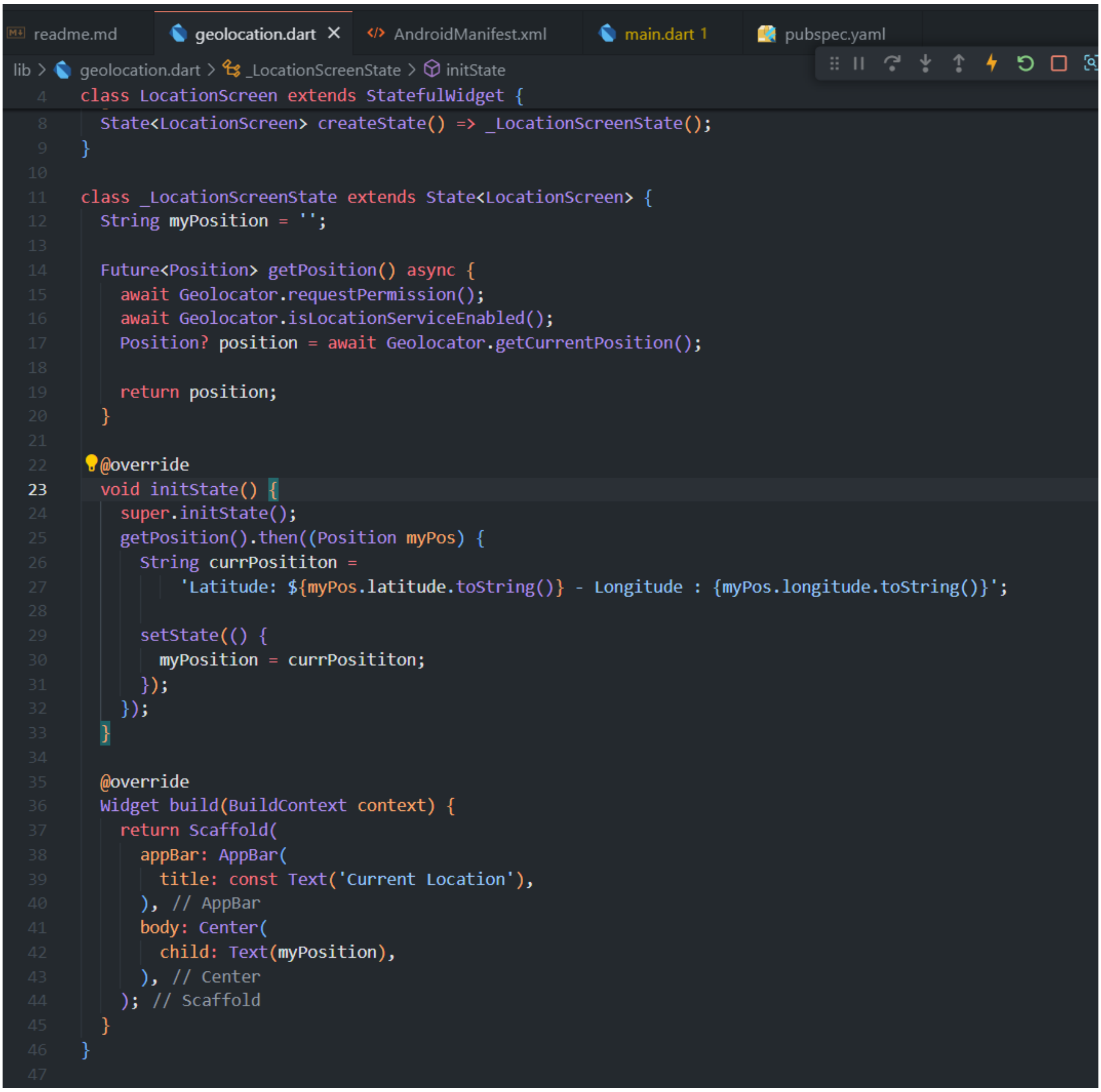
2. Tambah permisson GPS



books
readme.md
AndroidManifest.xml
main.dart 1
pubspec.yaml
android > app > src > main > AndroidManifest.xml
Follow link (ctrl + click)

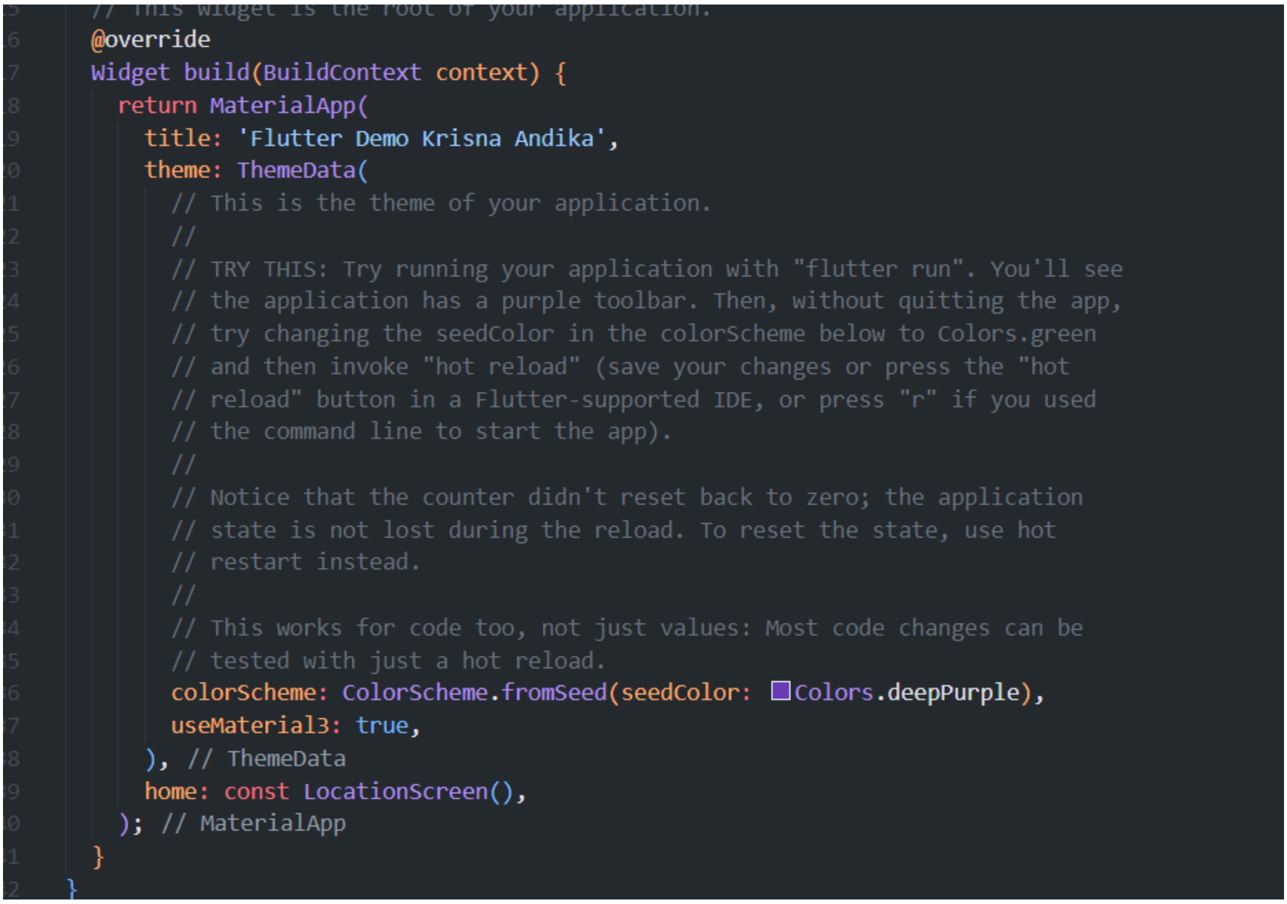
```xml
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
    <application
        <activity
            <!-- Specifies an Android theme to apply to this Activity as soon as
                 while the Flutter UI initializes. After that, this theme continues
                 to determine the Window background behind the Flutter UI. -->
            <meta-data
              android:name="io.flutter.embedding.android.NormalTheme"
              android:resource="@style/NormalTheme"
              />
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
        <!-- Don't delete the meta-data below.
             This is used by the Flutter tool to generate GeneratedPluginRegistrant.j
        <meta-data
            android:name="flutterEmbedding"
            android:value="2" />
    </application>
    <!-- Required to query activities that can process text,
         https://developer.android.com/training/package-visib
         https://developer.android.com/reference/android/content/Intent#ACTION_PROCES

         In particular, this is used by the Flutter engine in io.flutter.plugin.text.
    <queries>
        <intent>
            <action android:name="android.intent.action.PROCESS_TEXT"/>
            <data android:mimeType="text/plain"/>
        </intent>
    </queries>
</manifest>

<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
```

3. Buat file geolocation.dart



```dart
class LocationScreen extends StatefulWidget {
  State<LocationScreen> createState() => _LocationScreenState();
}

class _LocationScreenState extends State<LocationScreen> {
  String myPosition = '';

  Future<Position> getPosition() async {
    await Geolocator.requestPermission();
    await Geolocator.isLocationServiceEnabled();
    Position? position = await Geolocator.getCurrentPosition();

    return position;
  }

  @override
  void initState() {
    super.initState();
    getPosition().then((Position myPos) {
      String currPosititon =
          'Latitude: ${myPos.latitude.toString()} - Longitude : {myPos.longitude.toString()}';

      setState(() {
        myPosition = currPosititon;
      });
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Current Location'),
      ), // AppBar
      body: Center(
        child: Text(myPosition),
      ), // Center
    ); // Scaffold
  }
}
```

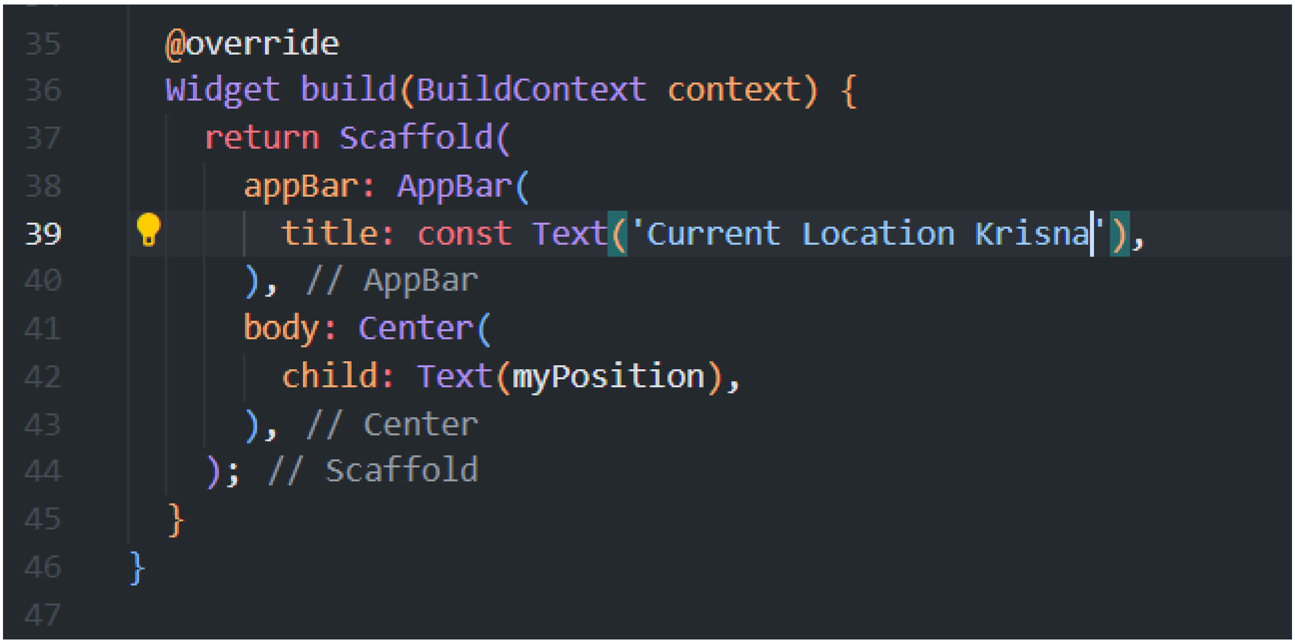
4. Edit main.dart



```
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo Krisna Andika',
      theme: ThemeData(
        // This is the theme of your application.
        //
        // TRY THIS: Try running your application with "flutter run". You'll see
        // the application has a purple toolbar. Then, without quitting the app,
        // try changing the seedColor in the colorScheme below to Colors.green
        // and then invoke "hot reload" (save your changes or press the "hot
        // reload" button in a Flutter-supported IDE, or press "r" if you used
        // the command line to start the app).
        //
        // Notice that the counter didn't reset back to zero; the application
        // state is not lost during the reload. To reset the state, use hot
        // restart instead.
        //
        // This works for code too, not just values: Most code changes can be
        // tested with just a hot reload.
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ), // ThemeData
      home: const LocationScreen(),
    ); // MaterialApp
  }
}
```

5. Tambahkan animasi loading

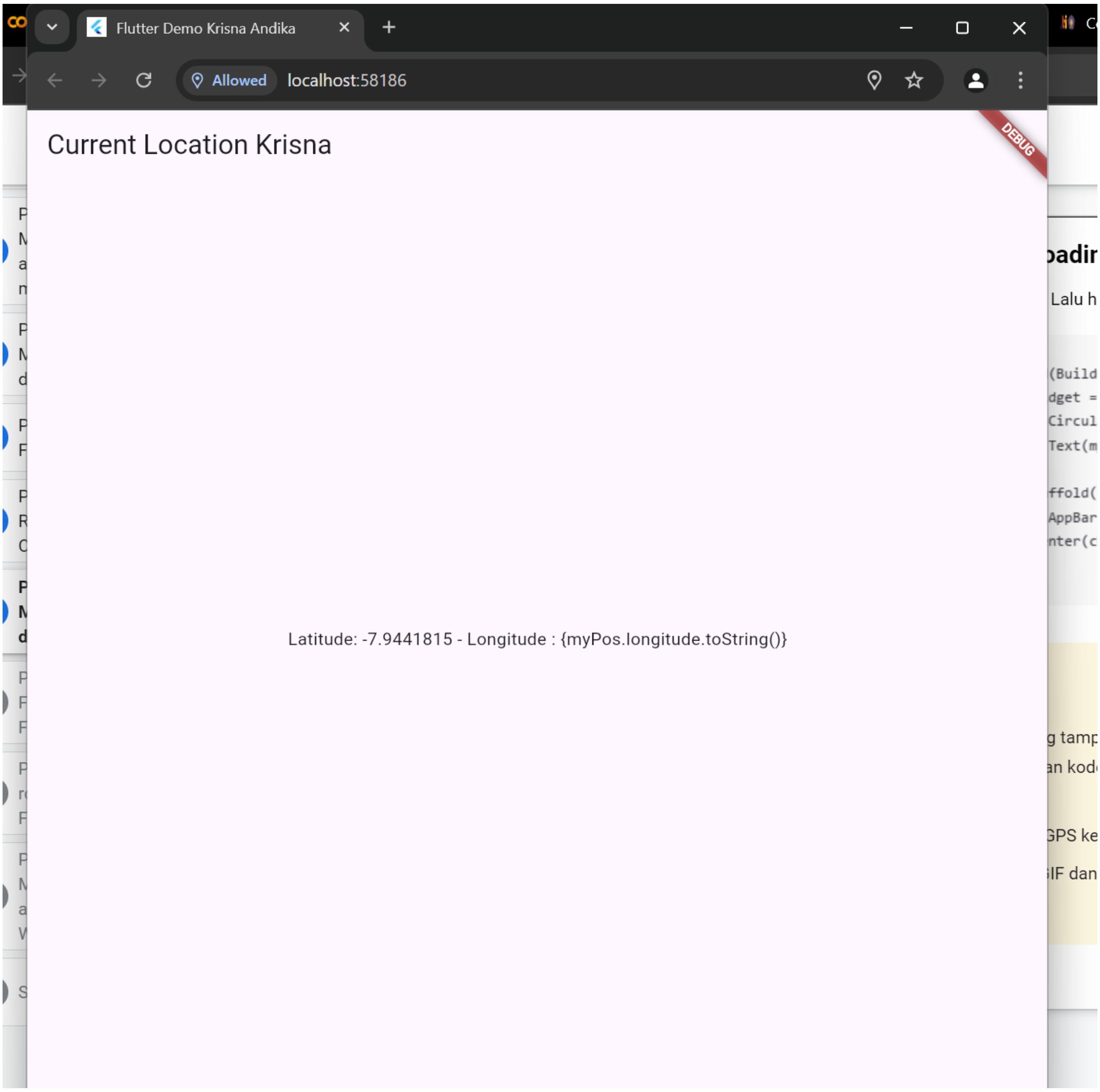
Soal 11



```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Current Location Krisna'),
      ), // AppBar
      body: Center(
        child: Text(myPosition),
      ), // Center
    ); // Scaffold
  }
}
```

Soal 12

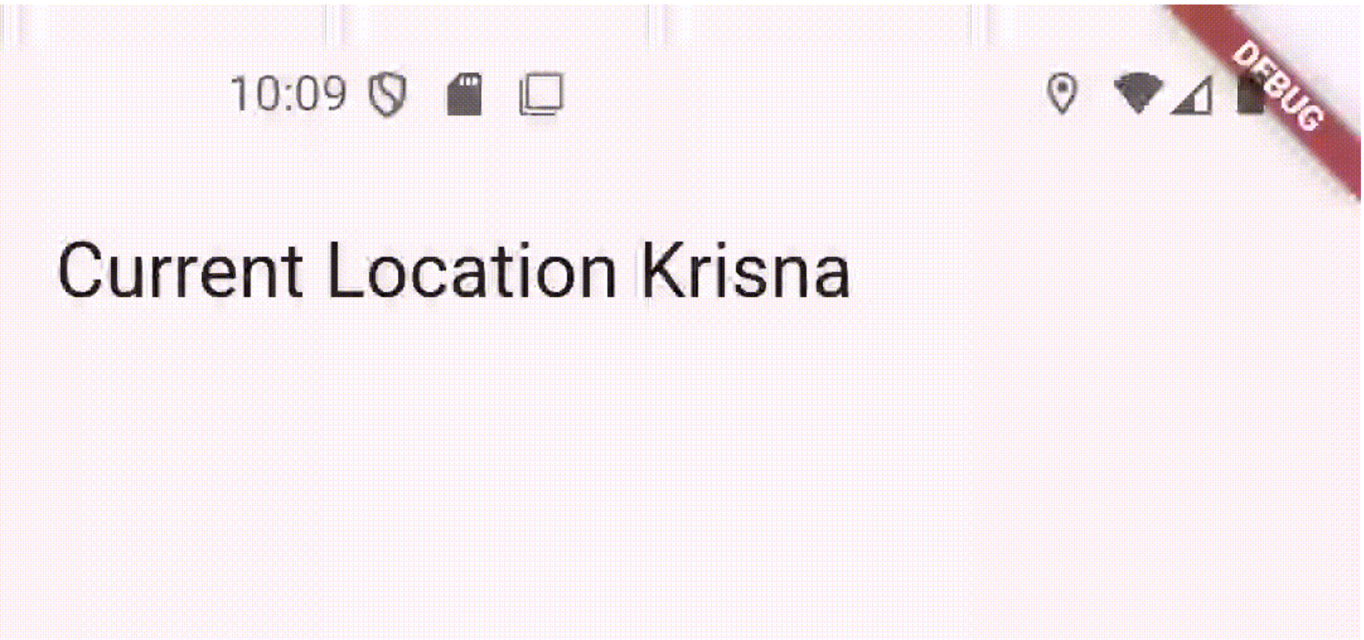
- percobaan pada browser



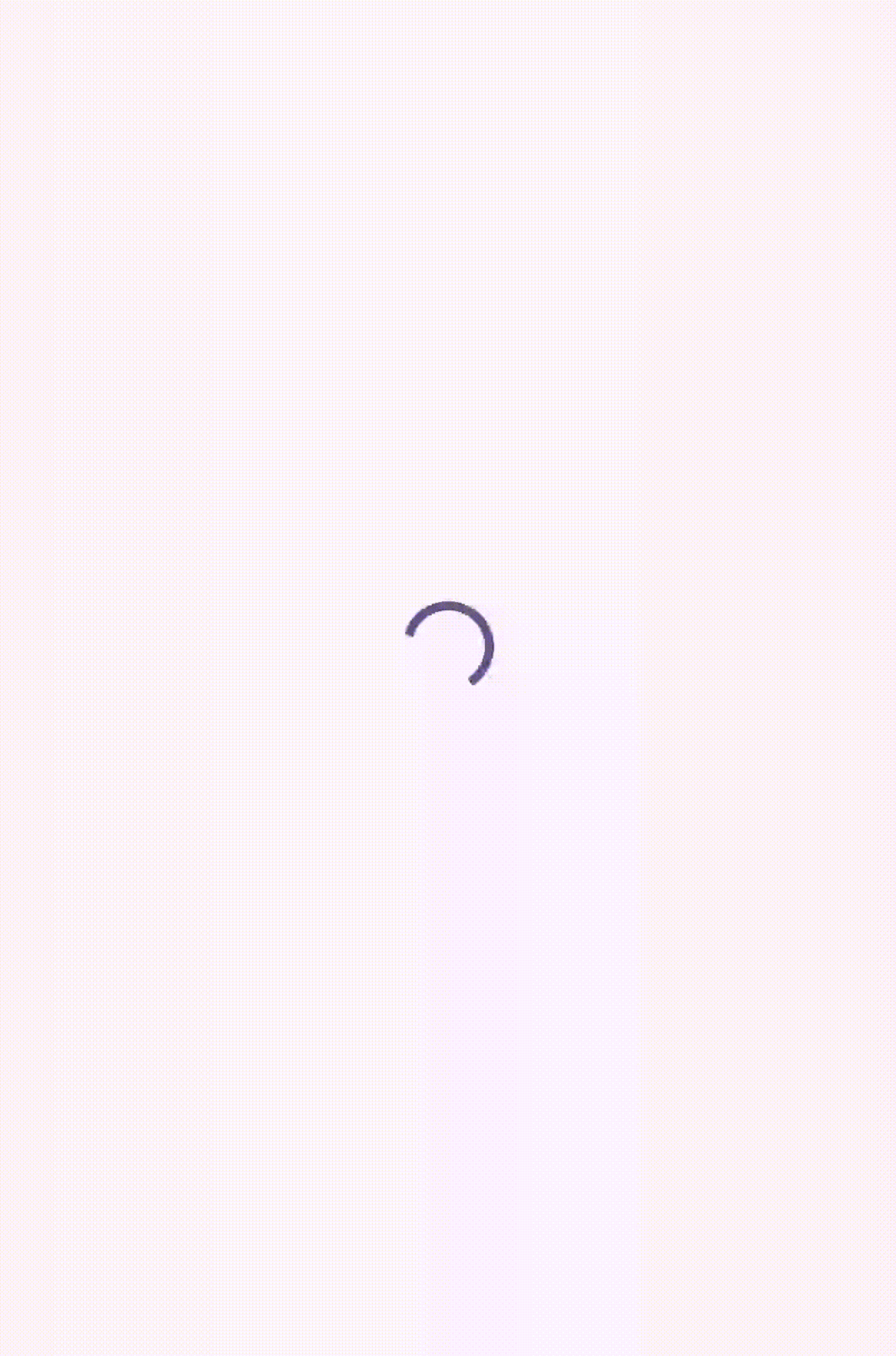
Current Location Krisna

Latitude: -7.9441815 - Longitude : {myPos.longitude.toString()}

> pada browser masih bisa mennjalankan selama perizinan lokasi di izinkan

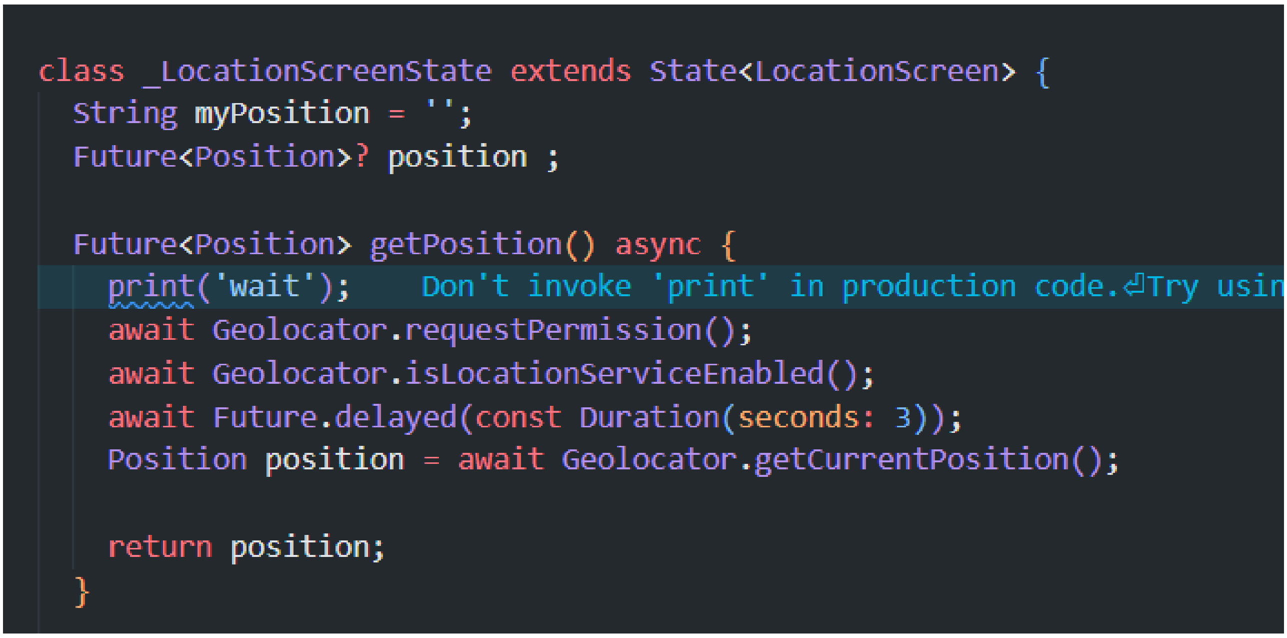


Current Location Krisna



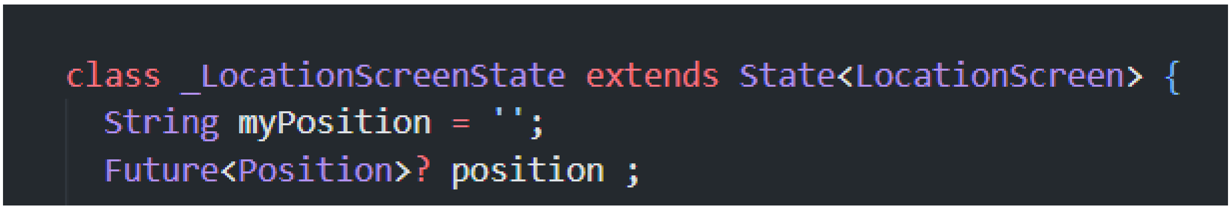
Praktikum 7

1. Modifikasi method getPosition()



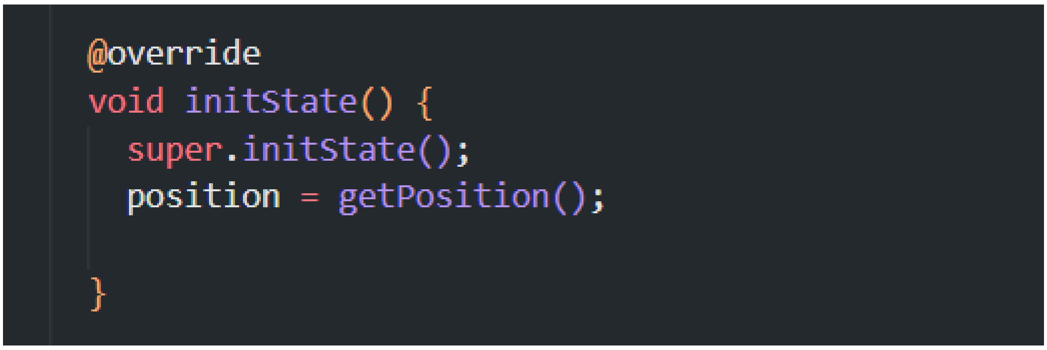
```
class _LocationScreenState extends State<LocationScreen> {
  String myPosition = '';
  Future<Position>? position ;

  Future<Position> getPosition() async {
    print('wait');    Don't invoke 'print' in production code.⏎Try usin
    await Geolocator.requestPermission();
    await Geolocator.isLocationServiceEnabled();
    await Future.delayed(const Duration(seconds: 3));
    Position position = await Geolocator.getCurrentPosition();

    return position;
  }
```

2. Tambah variabel



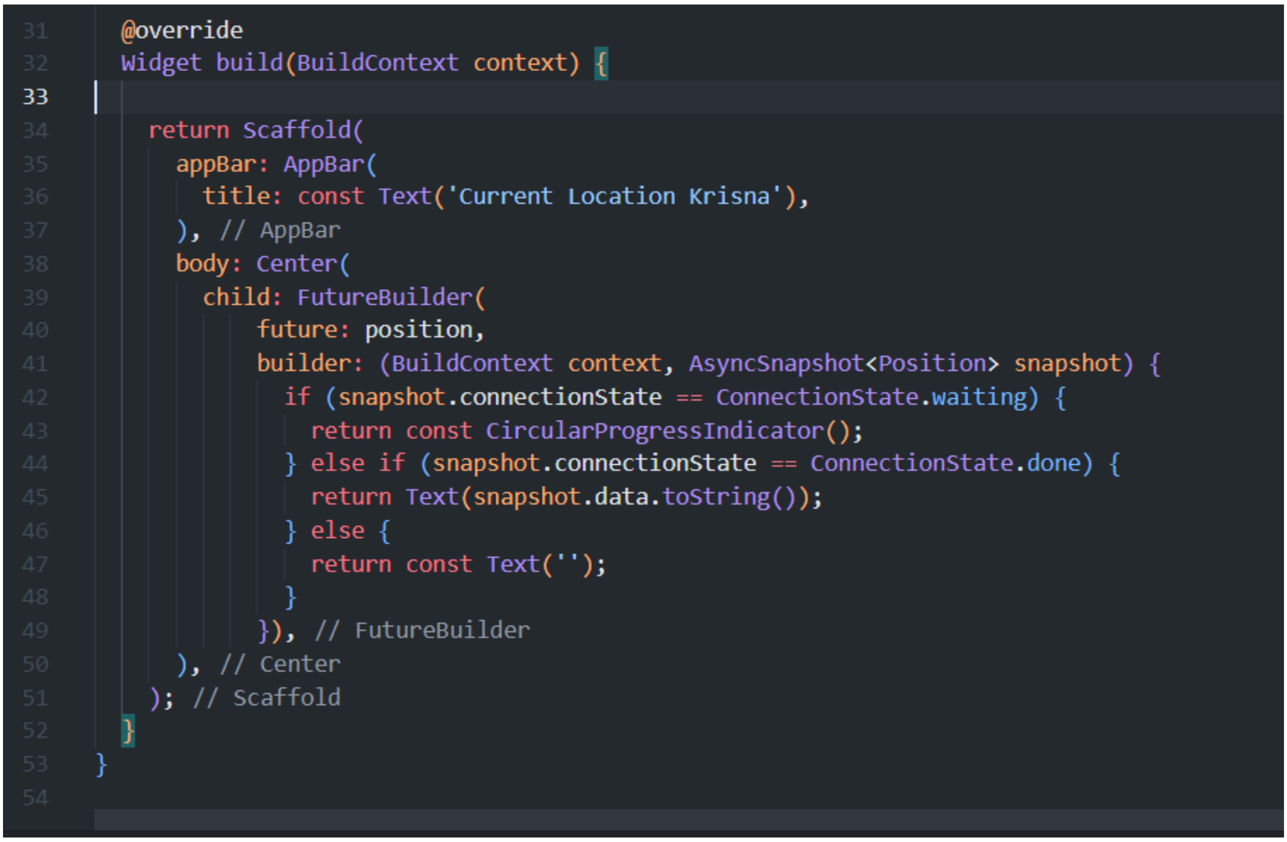
```
class _LocationScreenState extends State<LocationScreen> {
  String myPosition = '';
  Future<Position>? position ;
```

3. Tambah initState



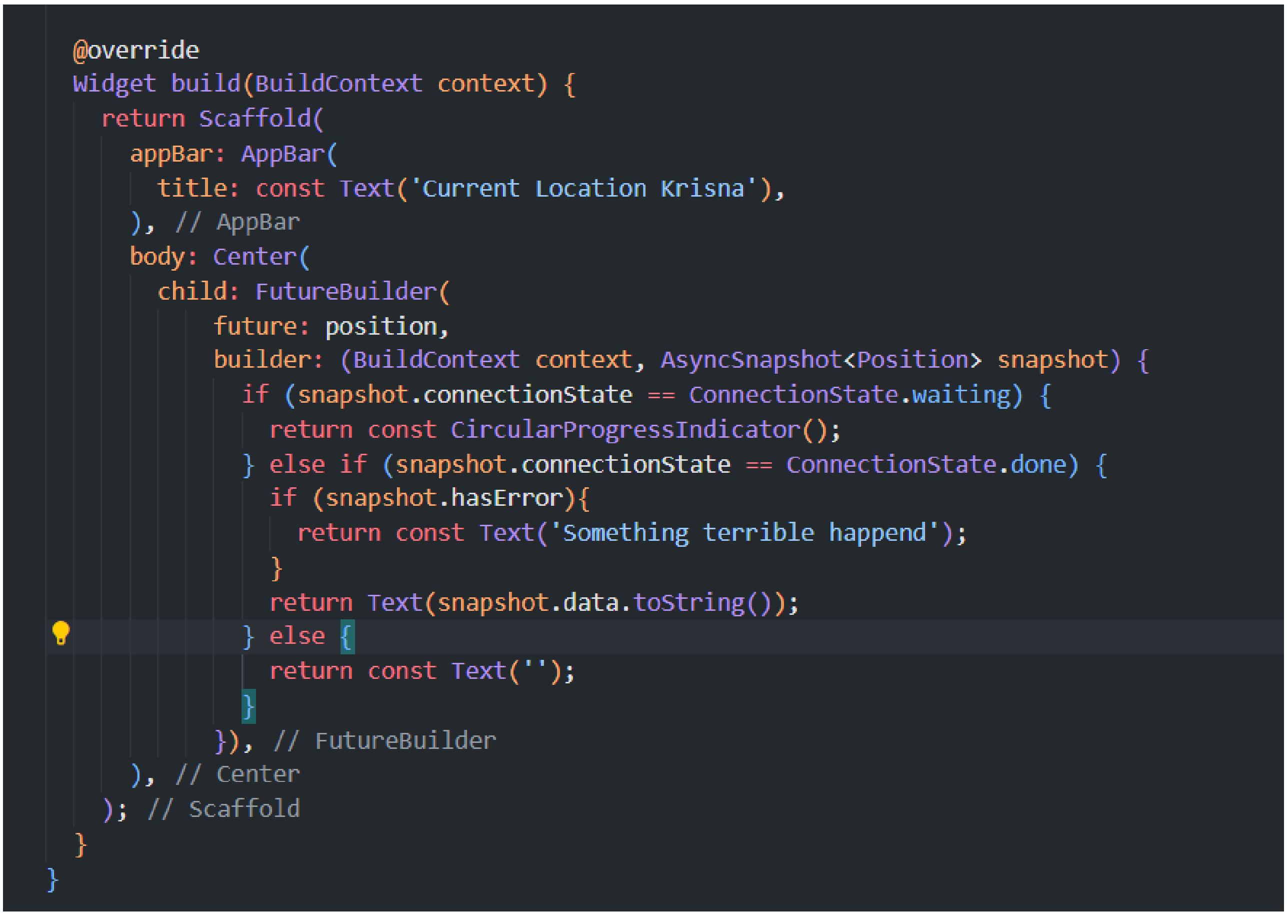
```
@override
void initState() {
  super.initState();
  position = getPosition();

}
```

4. Edit method build()



```
  @override
  Widget build(BuildContext context) {

    return Scaffold(
      appBar: AppBar(
        title: const Text('Current Location Krisna'),
      ), // AppBar
      body: Center(
        child: FutureBuilder(
            future: position,
            builder: (BuildContext context, AsyncSnapshot<Position> snapshot) {
              if (snapshot.connectionState == ConnectionState.waiting) {
                return const CircularProgressIndicator();
              } else if (snapshot.connectionState == ConnectionState.done) {
                return Text(snapshot.data.toString());
              } else {
                return const Text('');
              }
            }), // FutureBuilder
      ), // Center
    ); // Scaffold
  }
}
```

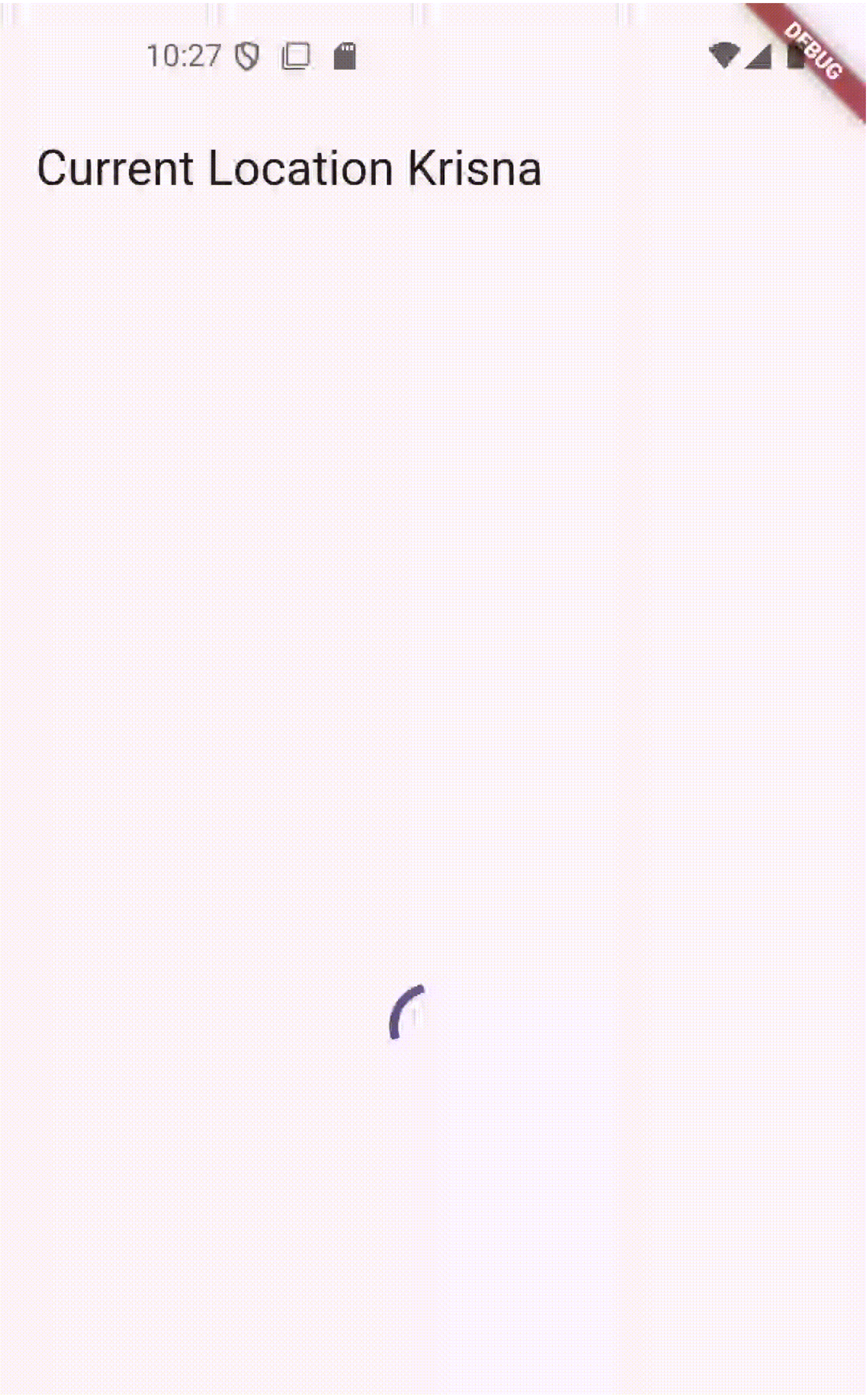
5. Tambah handling error



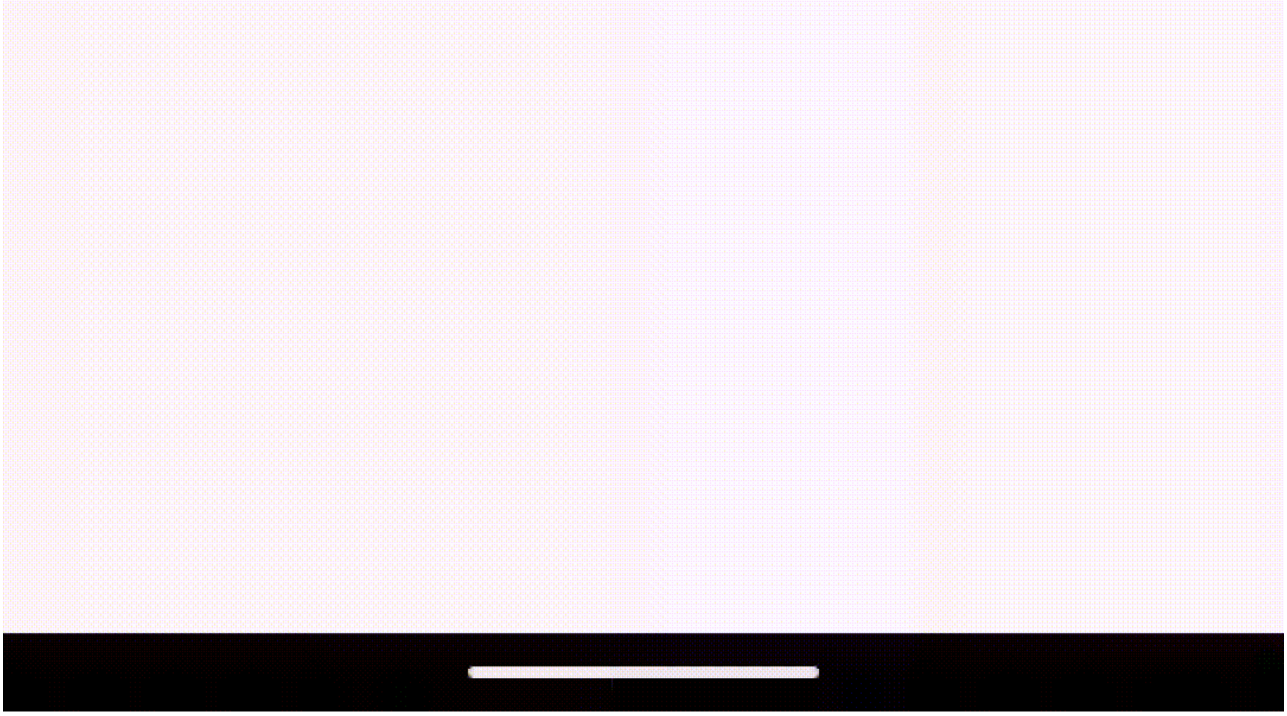
```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Current Location Krisna'),
      ), // AppBar
      body: Center(
        child: FutureBuilder(
            future: position,
            builder: (BuildContext context, AsyncSnapshot<Position> snapshot) {
              if (snapshot.connectionState == ConnectionState.waiting) {
                return const CircularProgressIndicator();
              } else if (snapshot.connectionState == ConnectionState.done) {
                if (snapshot.hasError){
                  return const Text('Something terrible happend');
                }
                return Text(snapshot.data.toString());
              } else {
                return const Text('');
              }
            }), // FutureBuilder
      ), // Center
    ); // Scaffold
  }
}
```

Soal 13

- Tidak ada perbedaan dengan UI sebelumnya, hanya saja handling pada data masih di proses dan sudah memiliki perbedaan dan juga terdapat handling bila error

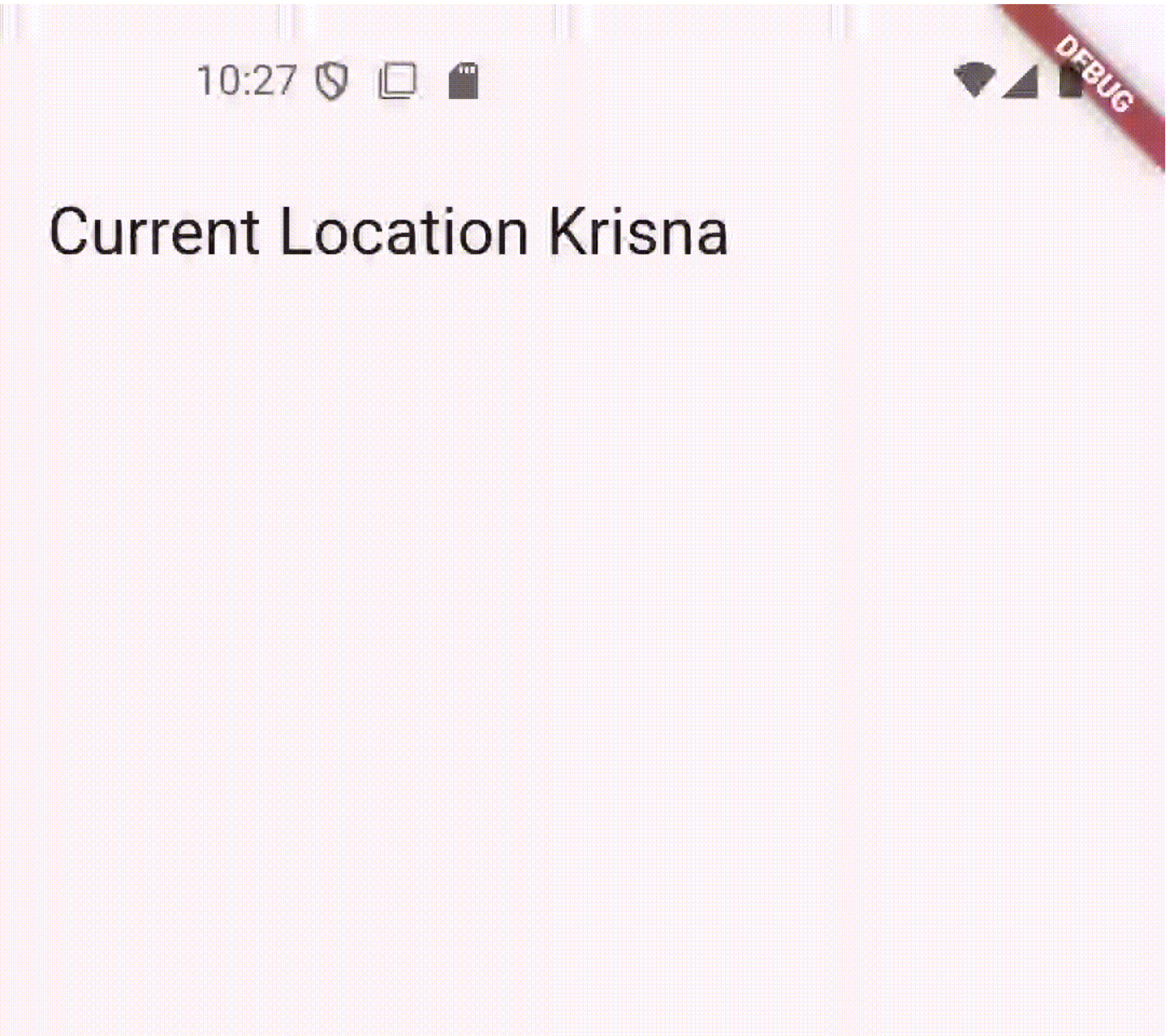


10:27
DEBUG
Current Location Krisna

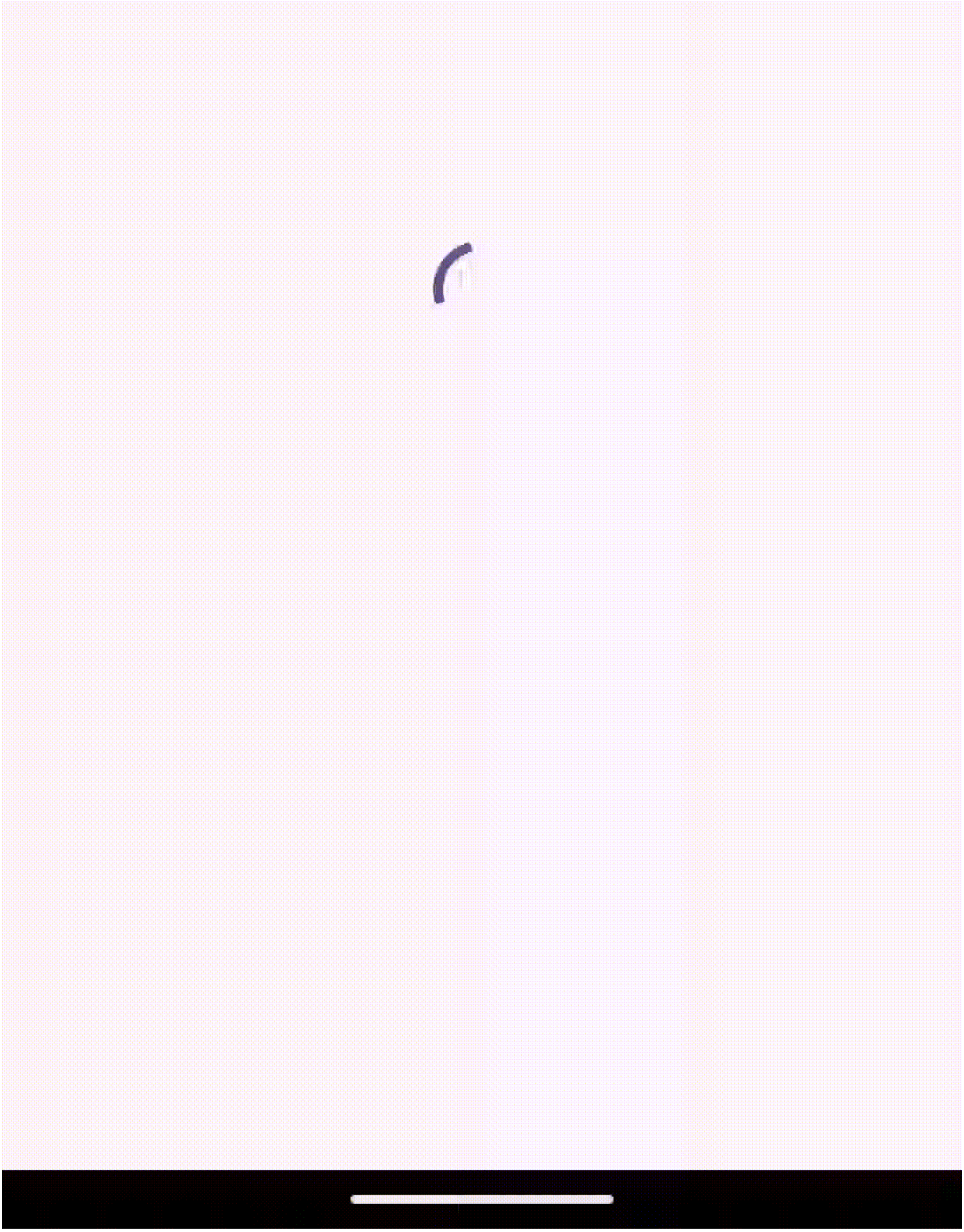


Soal 14

- Tidak ada perbedaan dengan UI sebelumnya,namun jika terjadi error saat process data maka akan menghasilkan text berbeda

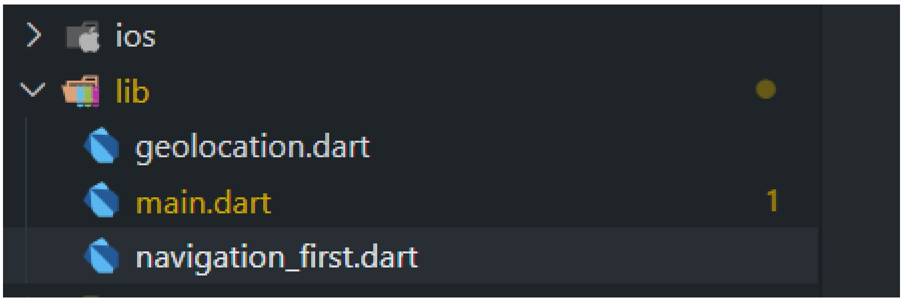


Current Location Krisna

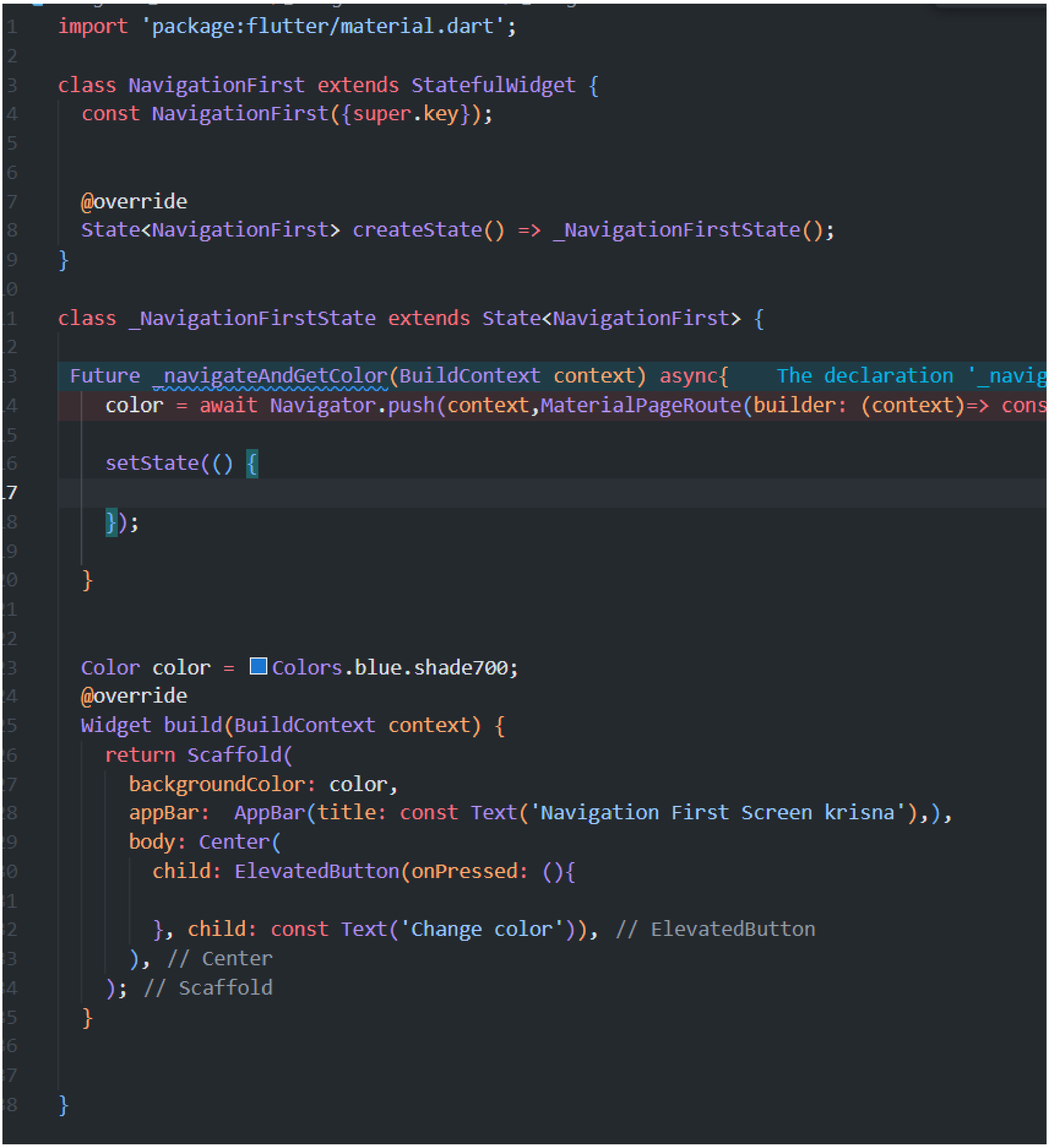


Praktikum 8

1. Buat file baru navigation_first.dart

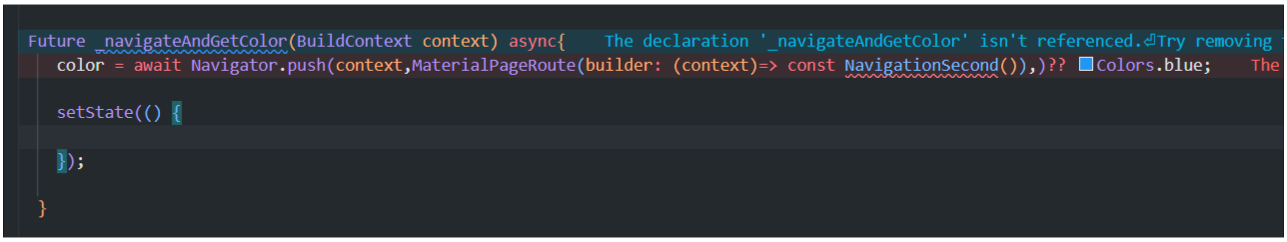


2. Isi kode navigation_first.dart



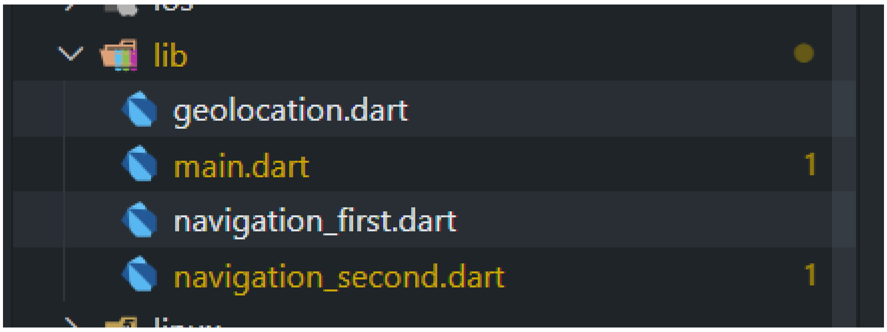
```
import 'package:flutter/material.dart';

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


  @override
  State<NavigationFirst> createState() => _NavigationFirstState();
}

class _NavigationFirstState extends State<NavigationFirst> {

 Future _navigateAndGetColor(BuildContext context) async{    The declaration '_navig
    color = await Navigator.push(context,MaterialPageRoute(builder: (context)=> cons

    setState(() {

    });

  }


  Color color = Colors.blue.shade700;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: color,
      appBar:  AppBar(title: const Text('Navigation First Screen krisna'),),
      body: Center(
        child: ElevatedButton(onPressed: (){

        }, child: const Text('Change color')), // ElevatedButton
      ), // Center
    ); // Scaffold
  }


}
```

3. Tambahkan method di class _navigationFirstState



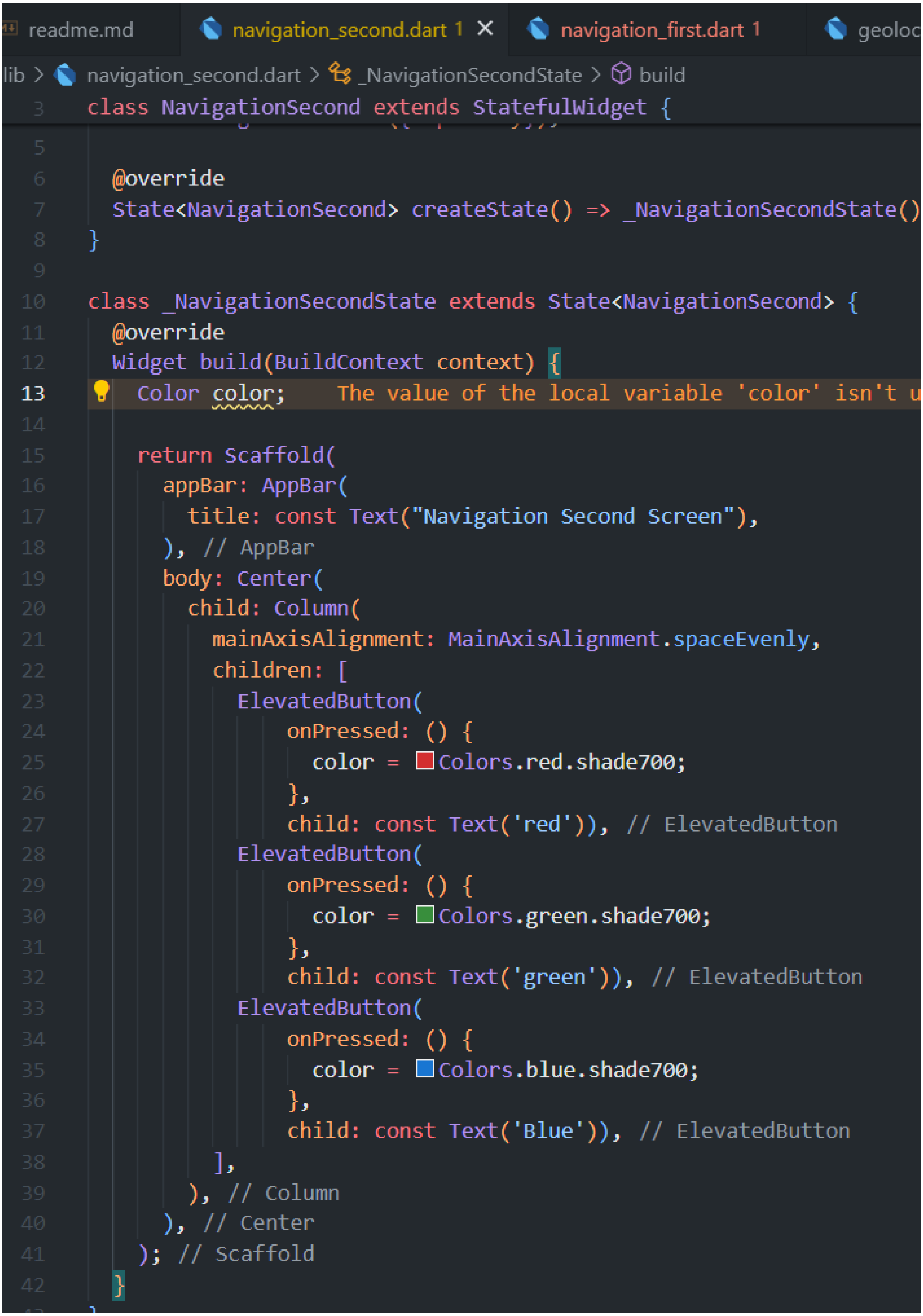
```
Future _navigateAndGetColor(BuildContext context) async{    The declaration '_navigateAndGetColor' isn't referenced.⏎Try removing
  color = await Navigator.push(context,MaterialPageRoute(builder: (context)=> const NavigationSecond()),)?? ■Colors.blue;    The

  setState(() {

  });

}
```

4. Buat file baru navigation_second.dart



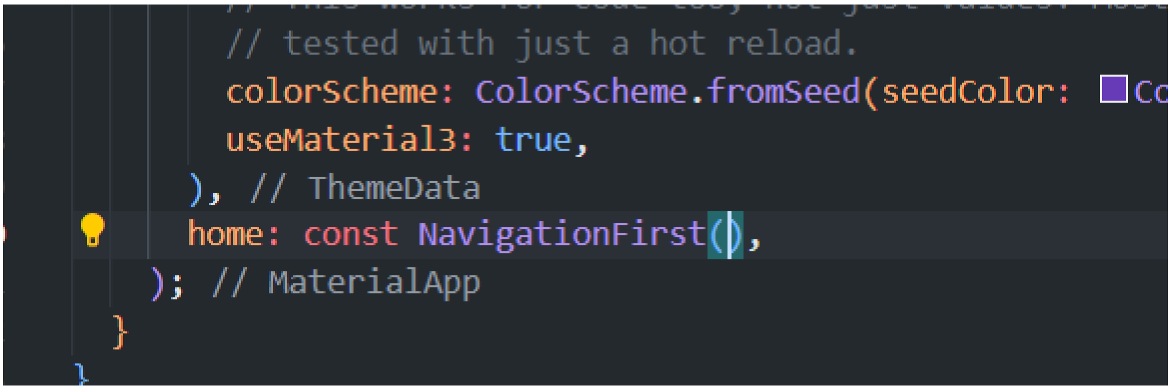
```
⌄ lib
    geolocation.dart
    main.dart                 1
    navigation_first.dart
    navigation_second.dart    1
```

5. Buat class NavigationSecond dengan statefullwidget



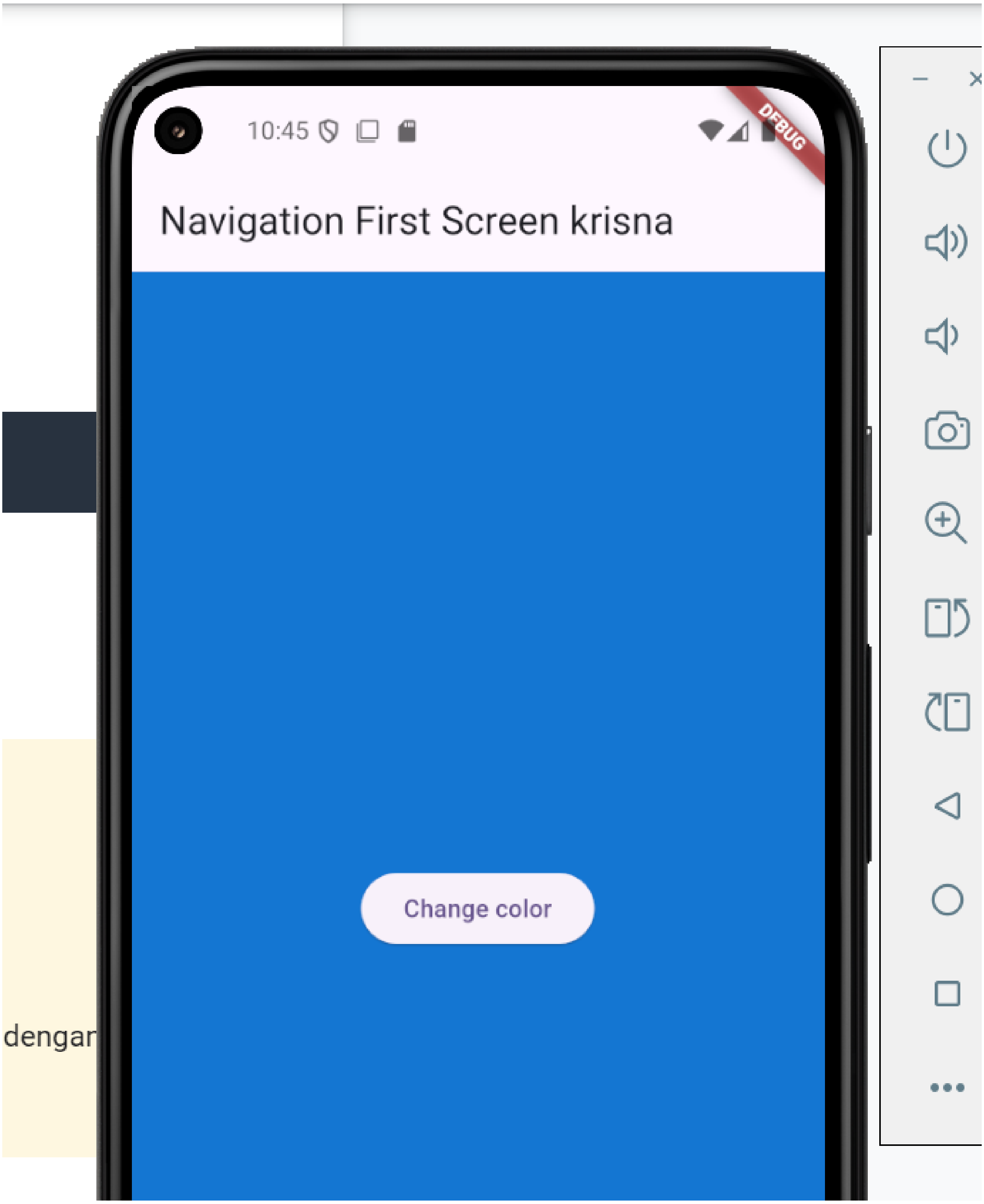
```dart
class NavigationSecond extends StatefulWidget {

  @override
  State<NavigationSecond> createState() => _NavigationSecondState()
}

class _NavigationSecondState extends State<NavigationSecond> {
  @override
  Widget build(BuildContext context) {
    Color color;    The value of the local variable 'color' isn't u

    return Scaffold(
      appBar: AppBar(
        title: const Text("Navigation Second Screen"),
      ), // AppBar
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
          children: [
            ElevatedButton(
                onPressed: () {
                  color = 🟥Colors.red.shade700;
                },
                child: const Text('red')), // ElevatedButton
            ElevatedButton(
                onPressed: () {
                  color = 🟩Colors.green.shade700;
                },
                child: const Text('green')), // ElevatedButton
            ElevatedButton(
                onPressed: () {
                  color = 🟦Colors.blue.shade700;
                },
                child: const Text('Blue')), // ElevatedButton
          ],
        ), // Column
      ), // Center
    ); // Scaffold
  }
```

6. Edit main.dart



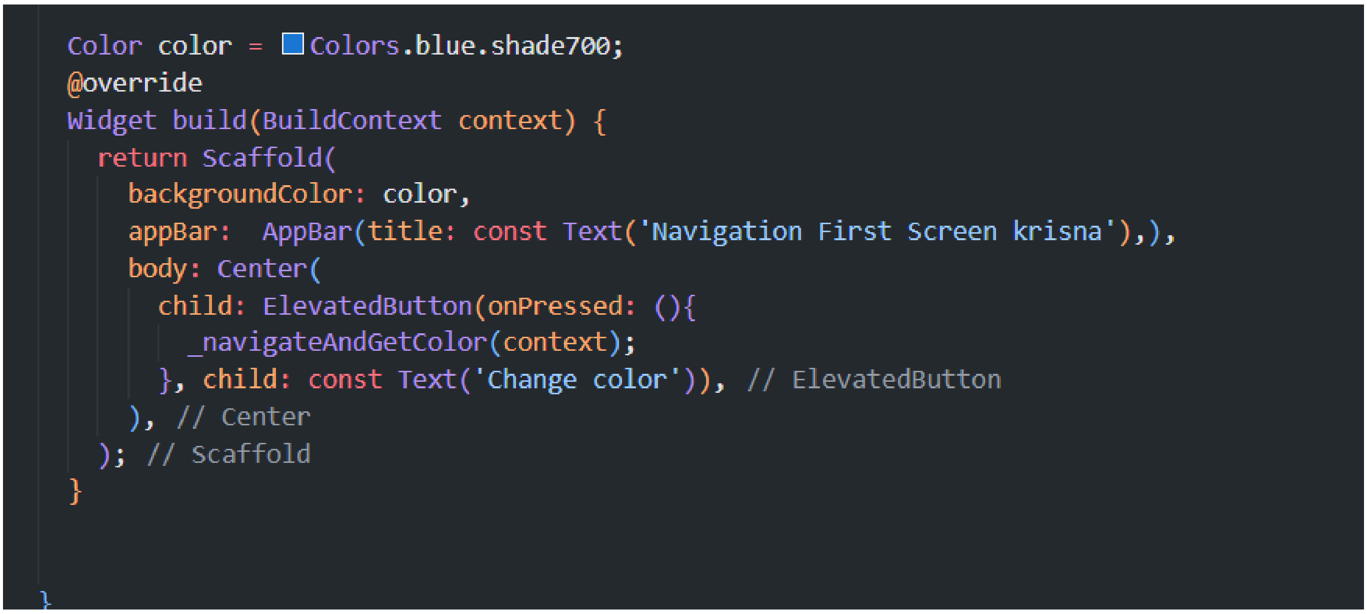
```
        // This works for code too, not just values: Most
        // tested with just a hot reload.
        colorScheme: ColorScheme.fromSeed(seedColor: ■Co
        useMaterial3: true,
      ), // ThemeData
      home: const NavigationFirst(),
    ); // MaterialApp
  }
}
```

7. run



Navigation First Screen krisna

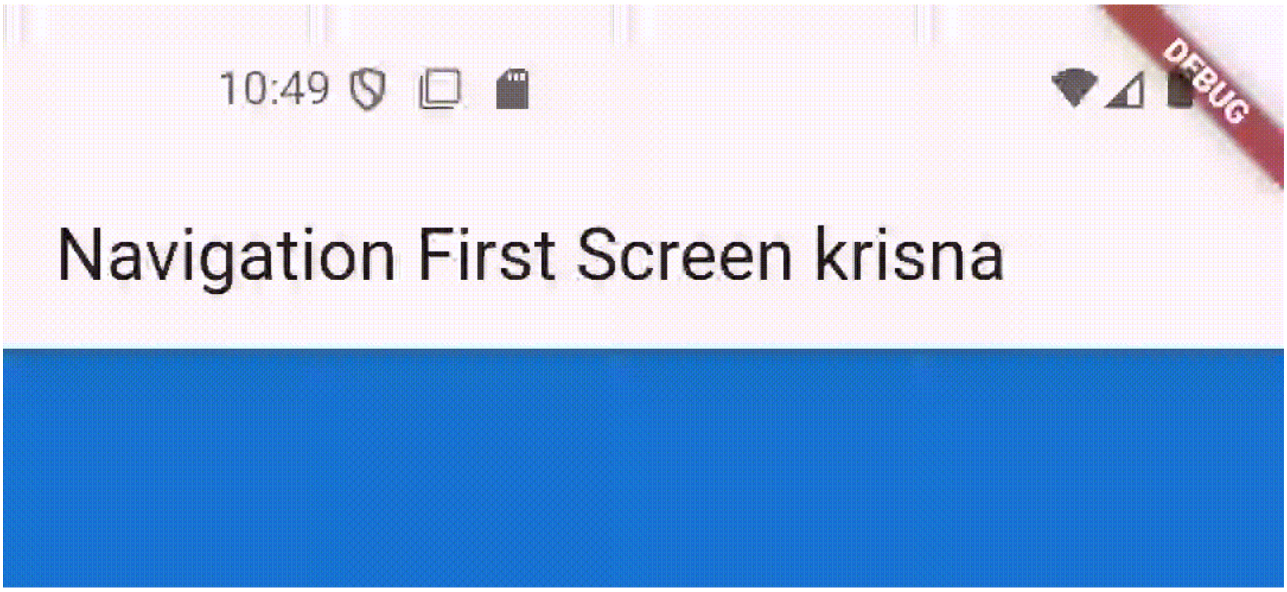
Soal 15

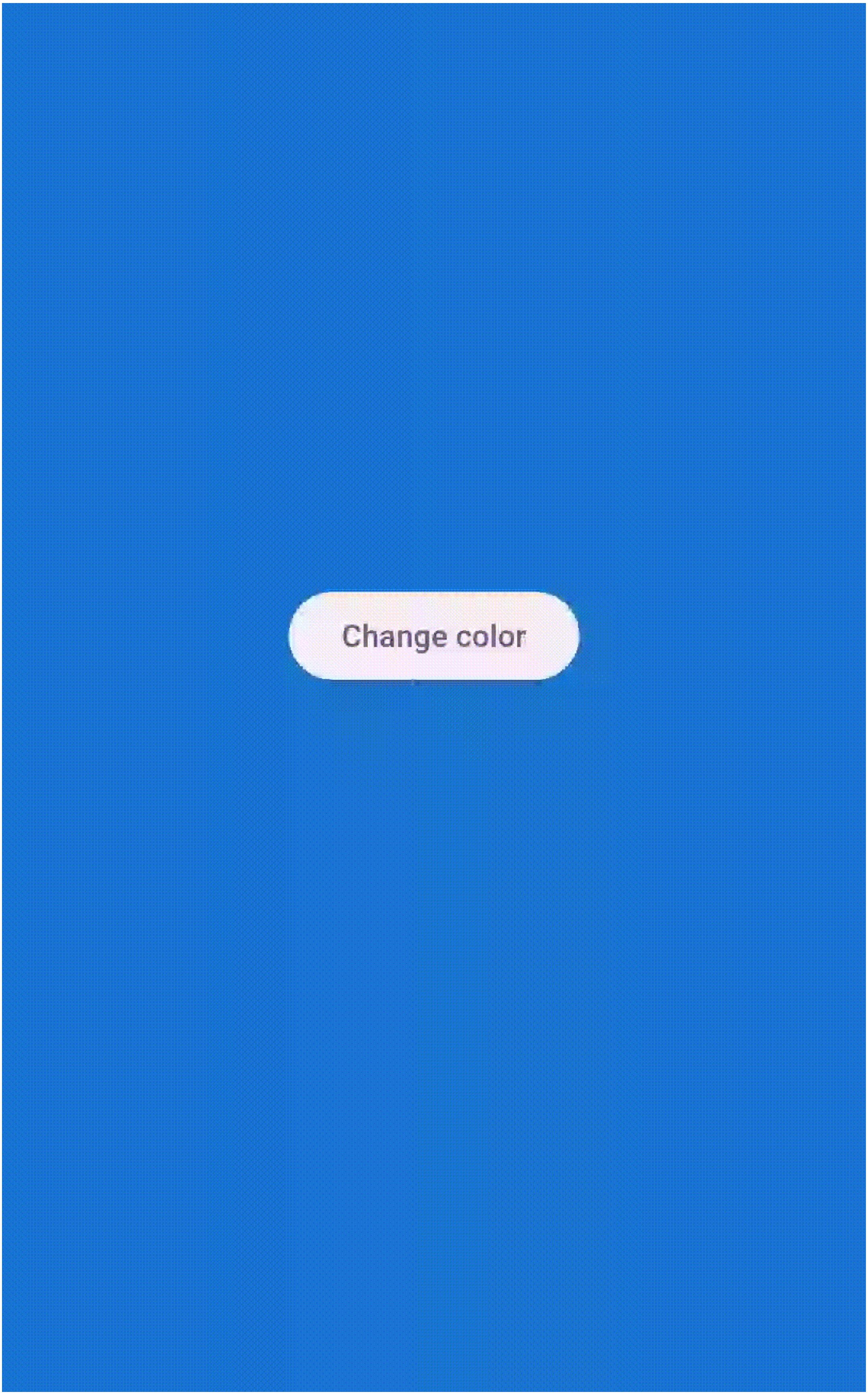


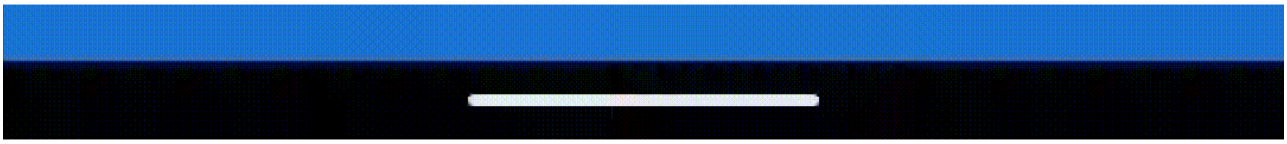
```
  Color color = Colors.blue.shade700;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: color,
      appBar:  AppBar(title: const Text('Navigation First Screen krisna'),),
      body: Center(
        child: ElevatedButton(onPressed: (){
          _navigateAndGetColor(context);
        }, child: const Text('Change color')), // ElevatedButton
      ), // Center
    ); // Scaffold
  }


}
```

soal 16

- Sudah sesuai dengan warna favorit saya

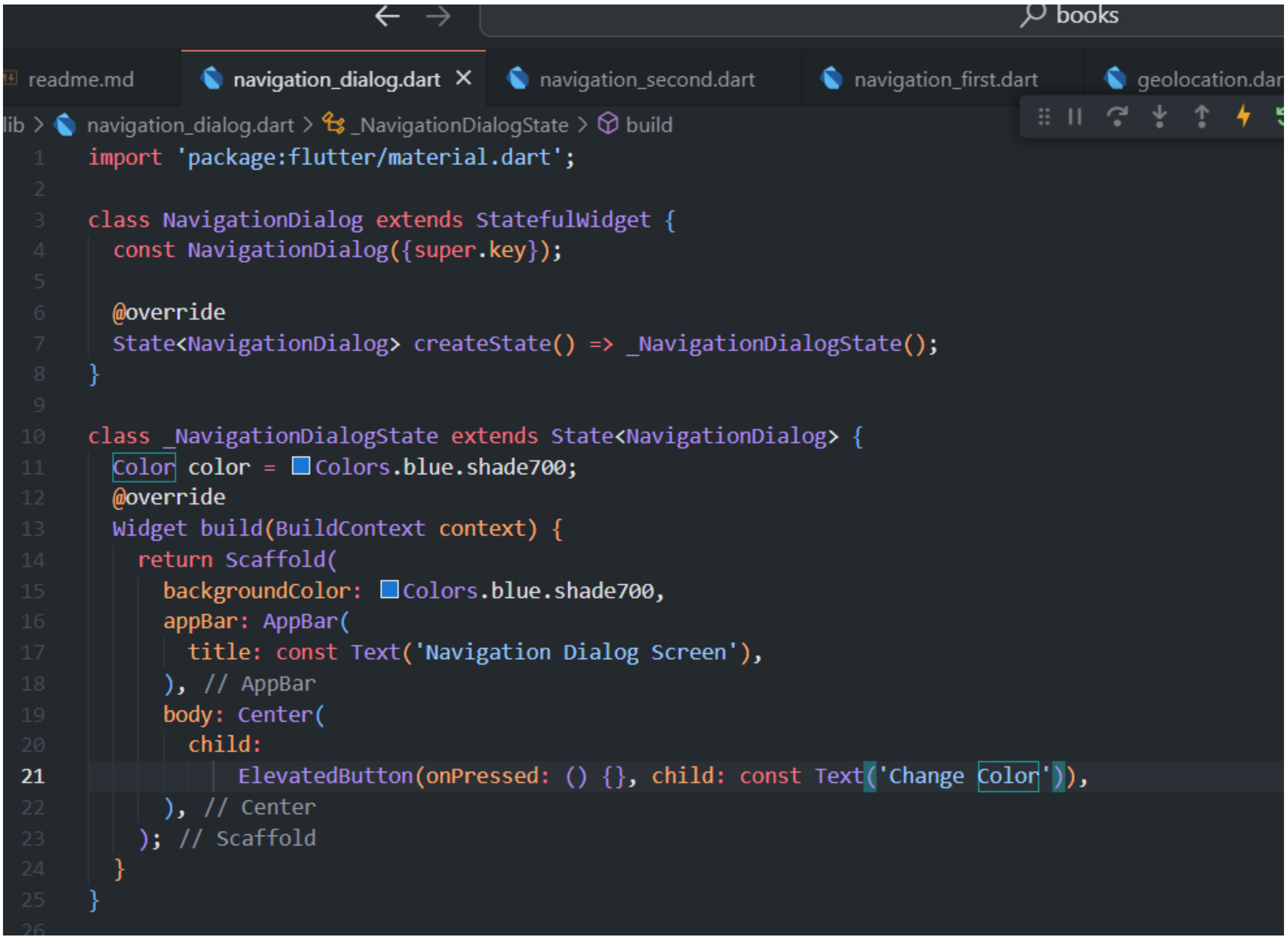



Change color



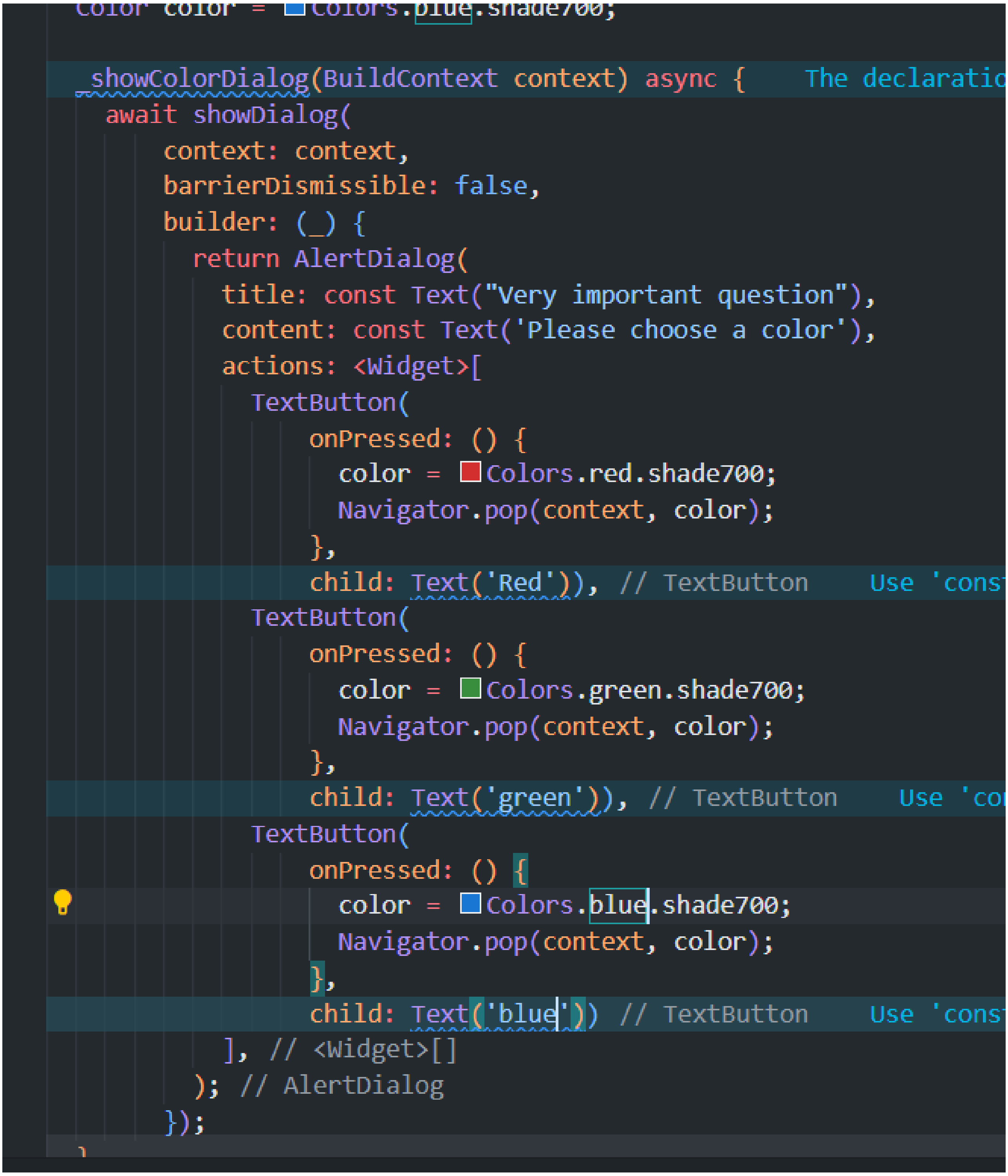
Praktikum 9

1. Buat file baru navigation_dialog.dart
2. Isi kode navigation_dialog.dart



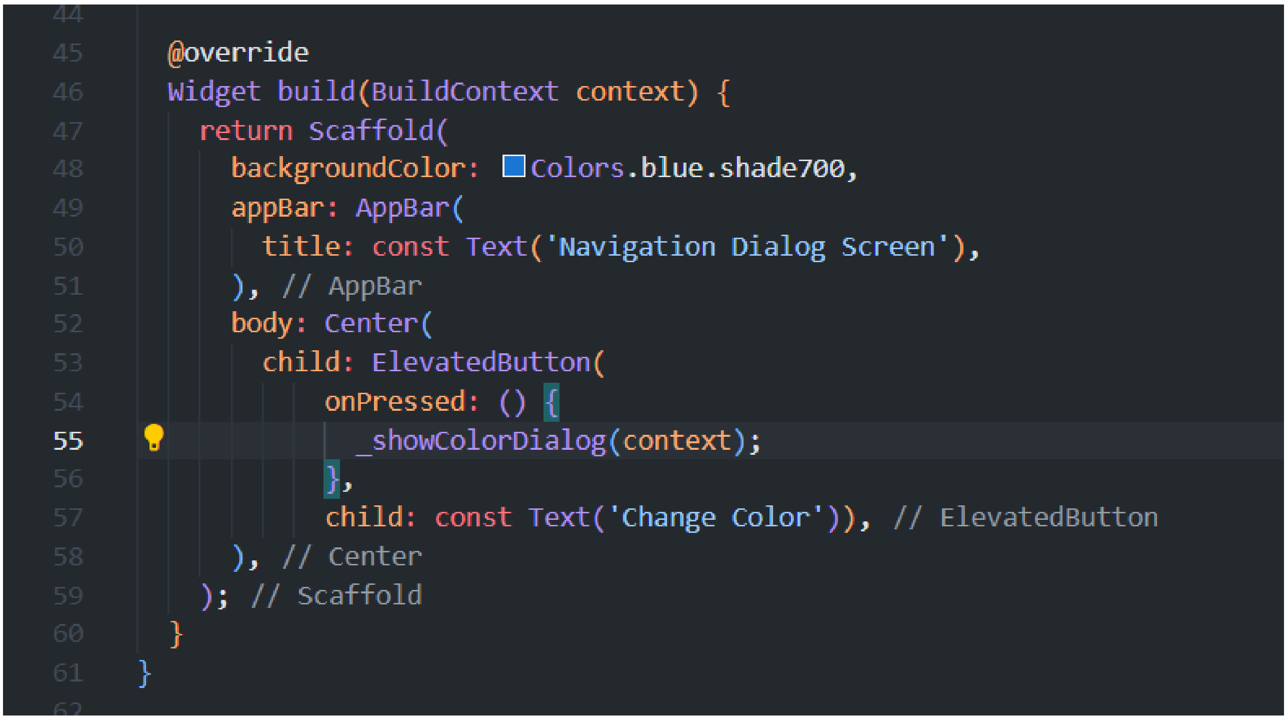
```dart
import 'package:flutter/material.dart';

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

  @override
  State<NavigationDialog> createState() => _NavigationDialogState();
}

class _NavigationDialogState extends State<NavigationDialog> {
  Color color = Colors.blue.shade700;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.blue.shade700,
      appBar: AppBar(
        title: const Text('Navigation Dialog Screen'),
      ), // AppBar
      body: Center(
        child:
          ElevatedButton(onPressed: () {}, child: const Text('Change Color')),
      ), // Center
    ); // Scaffold
  }
}
```

3. Tambah method async



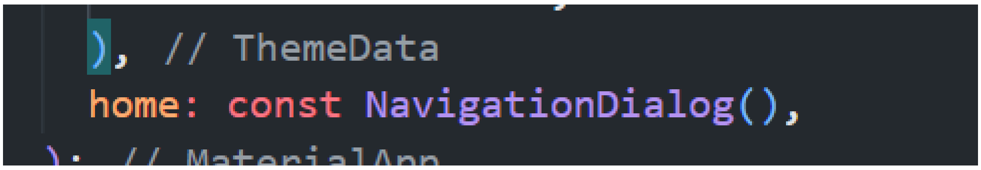
```
  Color color = 🟦Colors.blue.shade700;

  _showColorDialog(BuildContext context) async {    The declaratio
    await showDialog(
        context: context,
        barrierDismissible: false,
        builder: (_) {
          return AlertDialog(
            title: const Text("Very important question"),
            content: const Text('Please choose a color'),
            actions: <Widget>[
              TextButton(
                  onPressed: () {
                    color = 🟥Colors.red.shade700;
                    Navigator.pop(context, color);
                  },
                  child: Text('Red')), // TextButton    Use 'cons
              TextButton(
                  onPressed: () {
                    color = 🟩Colors.green.shade700;
                    Navigator.pop(context, color);
                  },
                  child: Text('green')), // TextButton    Use 'co
              TextButton(
                  onPressed: () {
                    color = 🟦Colors.blue.shade700;
                    Navigator.pop(context, color);
                  },
                  child: Text('blue')) // TextButton    Use 'cons
            ], // <Widget>[]
          ); // AlertDialog
        });
  }
```

4. panggil method di elevated button



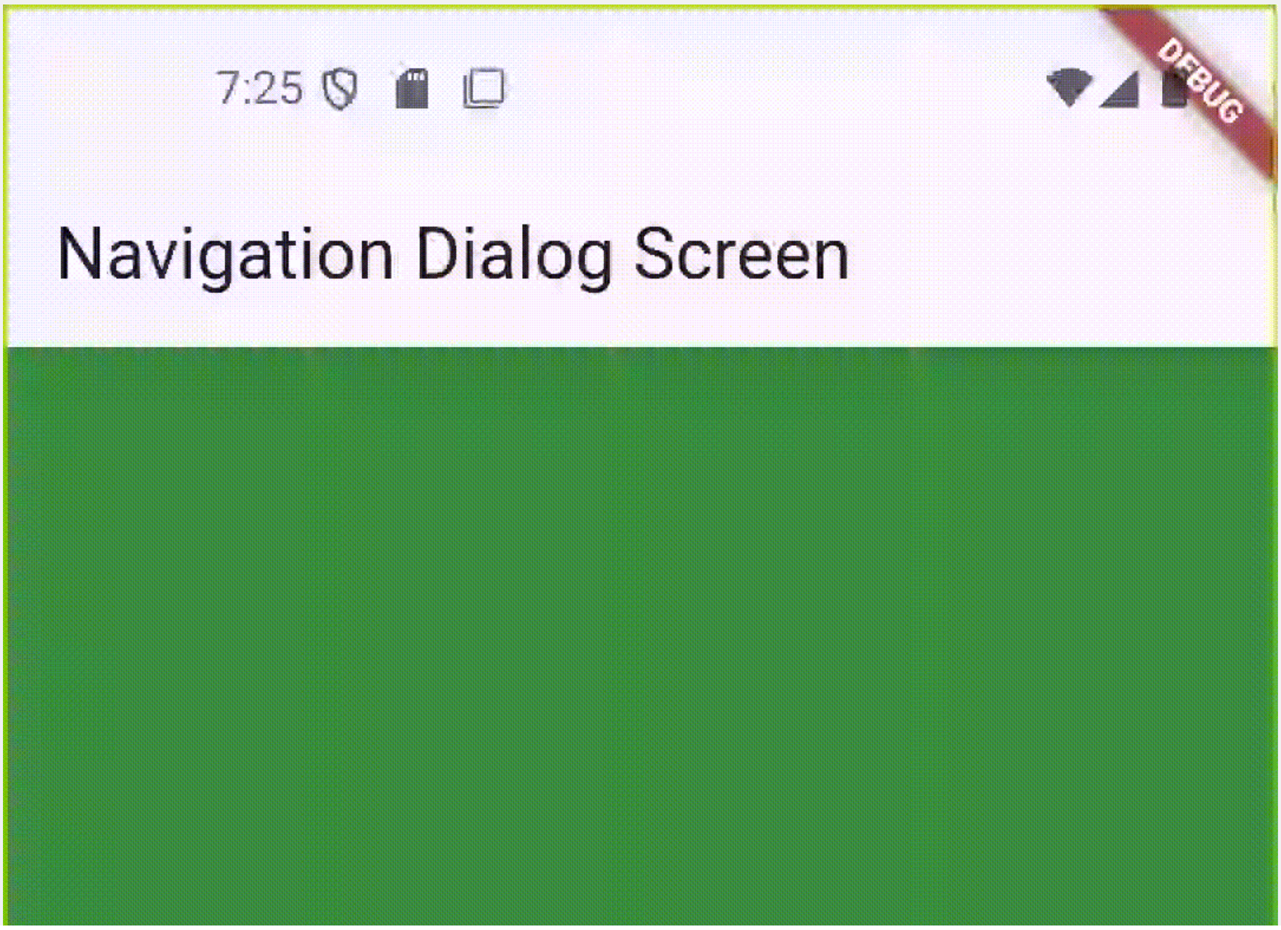
```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.blue.shade700,
      appBar: AppBar(
        title: const Text('Navigation Dialog Screen'),
      ), // AppBar
      body: Center(
        child: ElevatedButton(
            onPressed: () {
              _showColorDialog(context);
            },
            child: const Text('Change Color')), // ElevatedButton
      ), // Center
    ); // Scaffold
  }
}
```

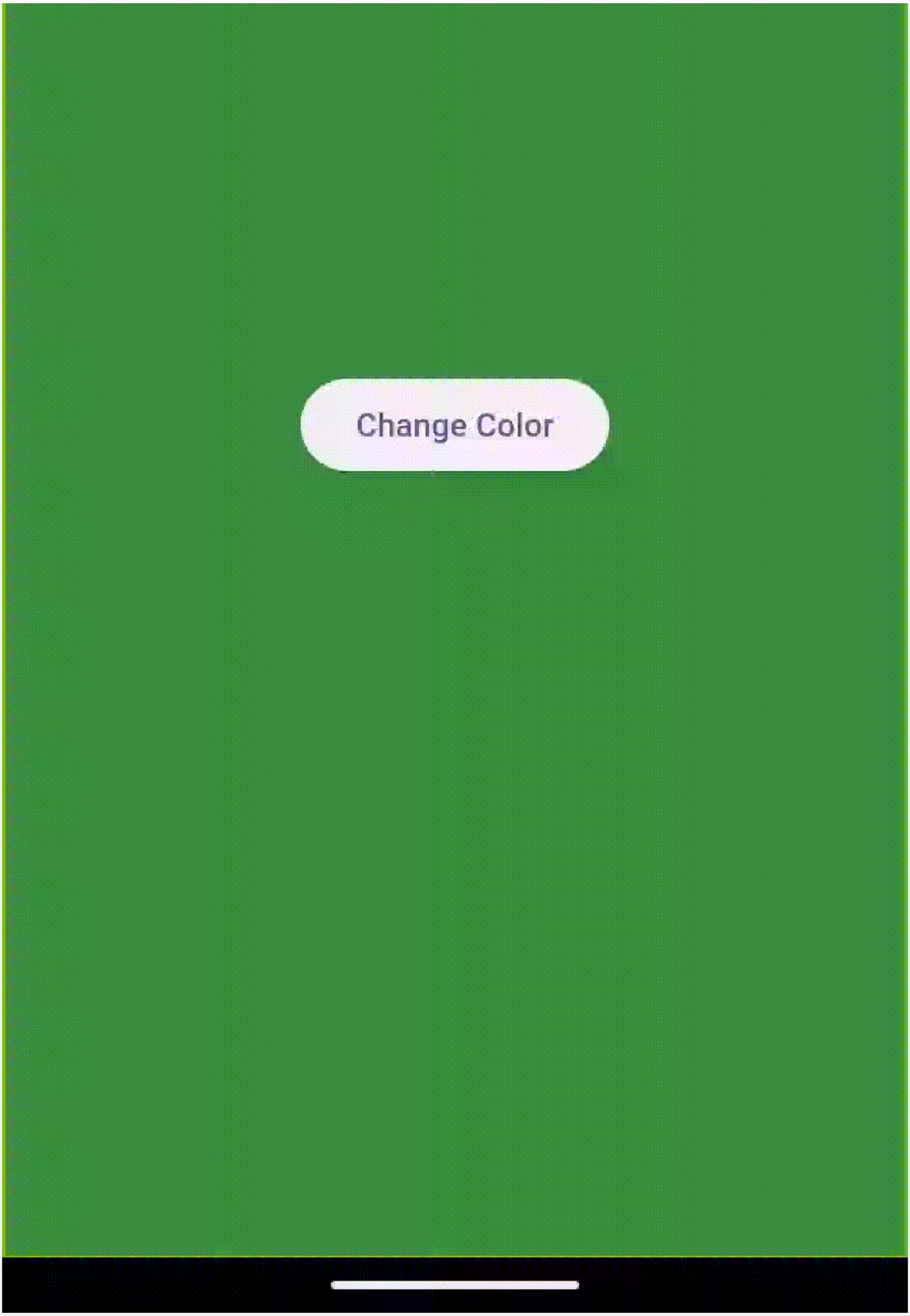
5. Edit main.dart



```
      ), // ThemeData
      home: const NavigationDialog(),
    ); // MaterialApp
```

Soal 17

> terjadi karena ketika kita memilih salah satu warna, maka program akan menjalankan navigator.pop dengan warna yang berbeda
>
> 
>



Change Color