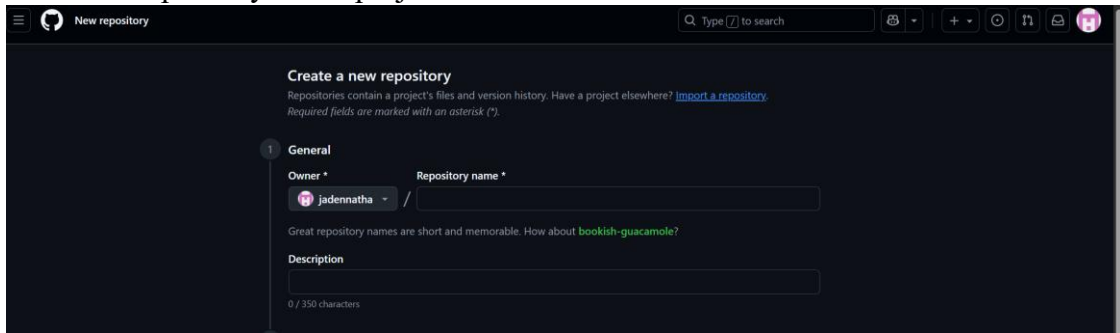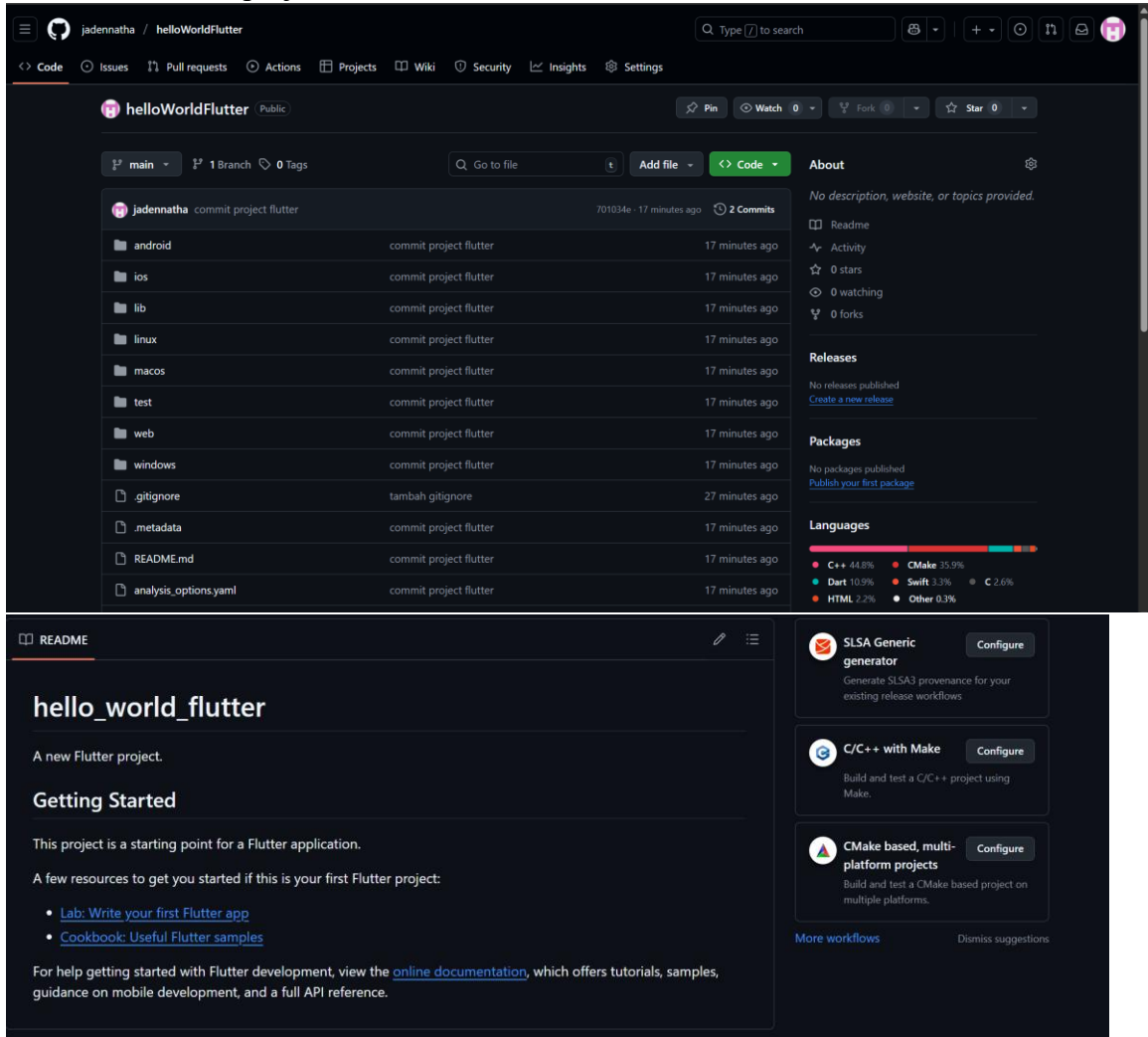# JOBSHEET 4 FLUTTER

PEMROGRAMAN MOBILE

JADEN NATHA KAUTSAR
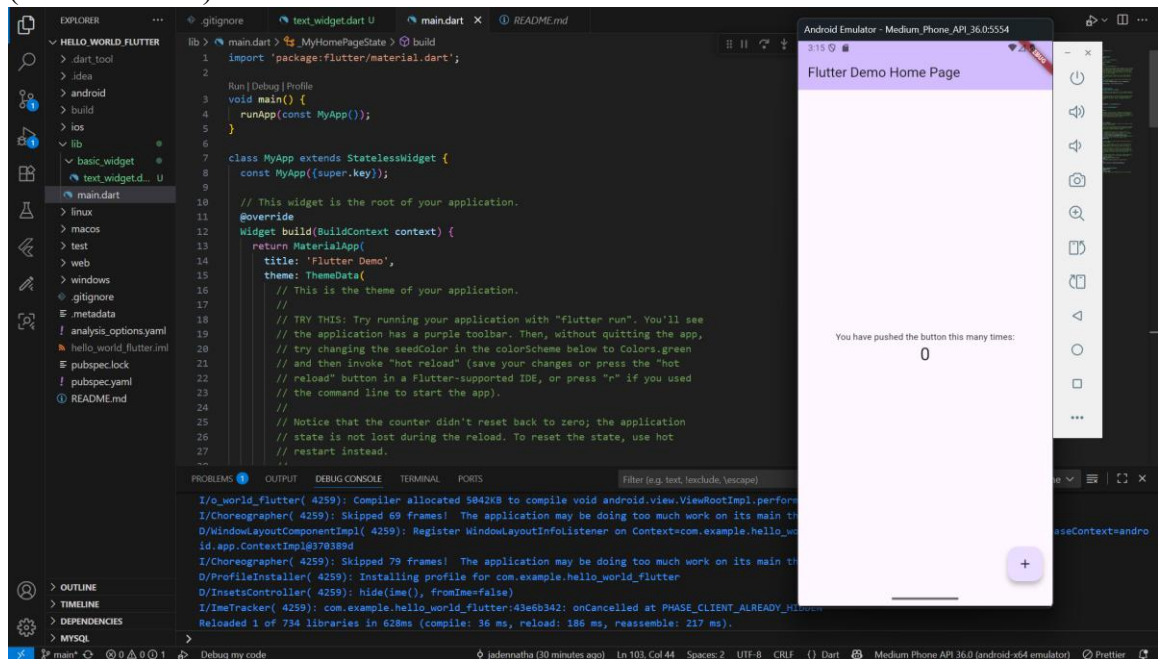
2341760068

SISTEM INFORMASI BISNIS

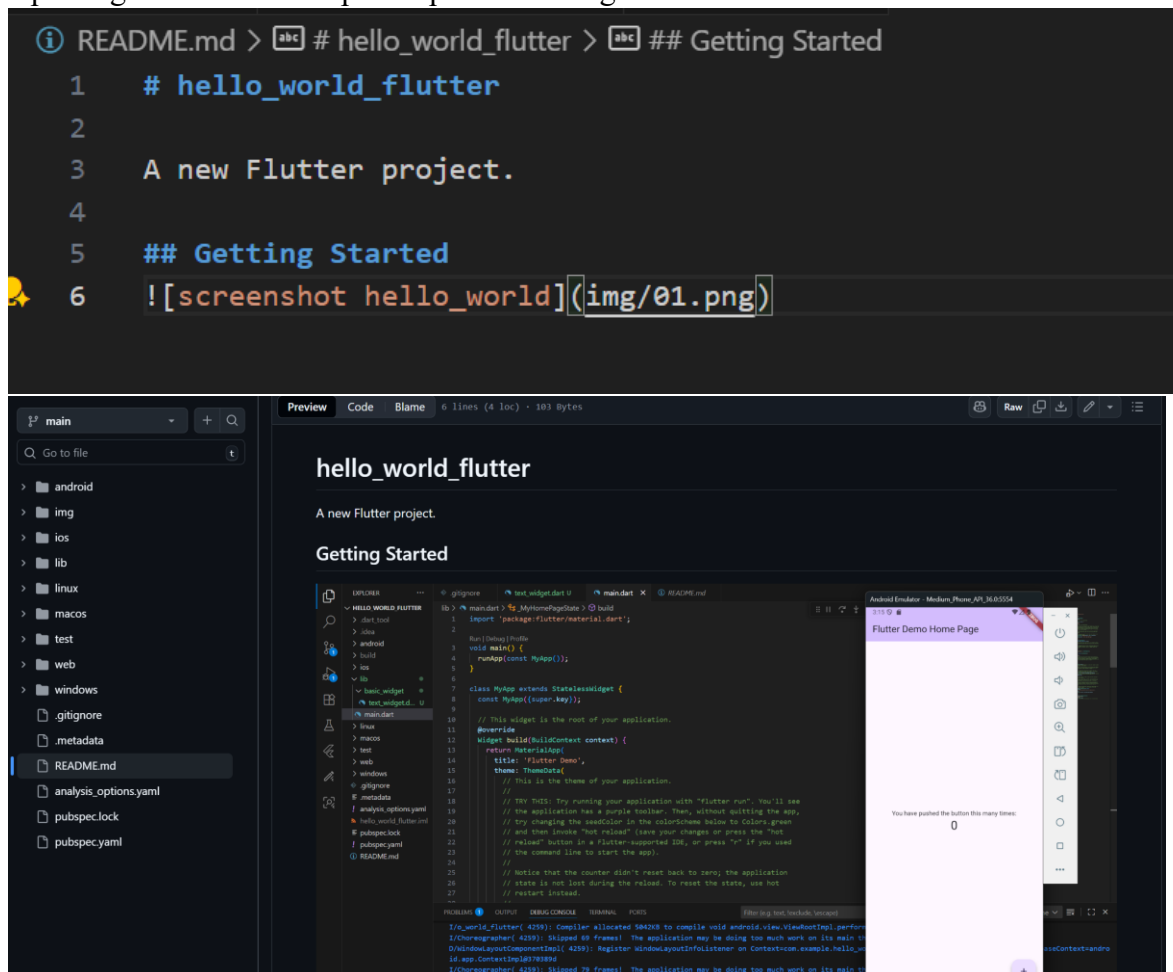1. Membuat repository untuk project baru

2. Hasil dari commit project flutter

3. Menjalankan project flutter untuk pertama kali dengan menggunakan emulator (android studio)


4. Upload gambar dan ditampilkan pada readme github

5. Membuat text widget dengan membuat folder baru dengan nama basic_widget dan membuat file dart baru dengan nama text_widget.dart

```dart
lib > basic_widget > text_widget.dart > MyTextWidget > MyTextWidget
 1    import 'package:flutter/material.dart';
 2
 3    class MyTextWidget extends StatelessWidget {
 4      const MyTextWidget({Key? key}) : super(key: key);
 5
 6      @override
 7      Widget build(BuildContext context) {
 8        return const Text (
 9          "Nama saya Jaden, sedang belajar pemrograman flutter",
10          style: TextStyle(color: Colors.blue, fontSize: 14),
11          textAlign: TextAlign.center); // Text
12      }
13    }
```

6. Mengubah kode pada file main dan berikut tampilannya
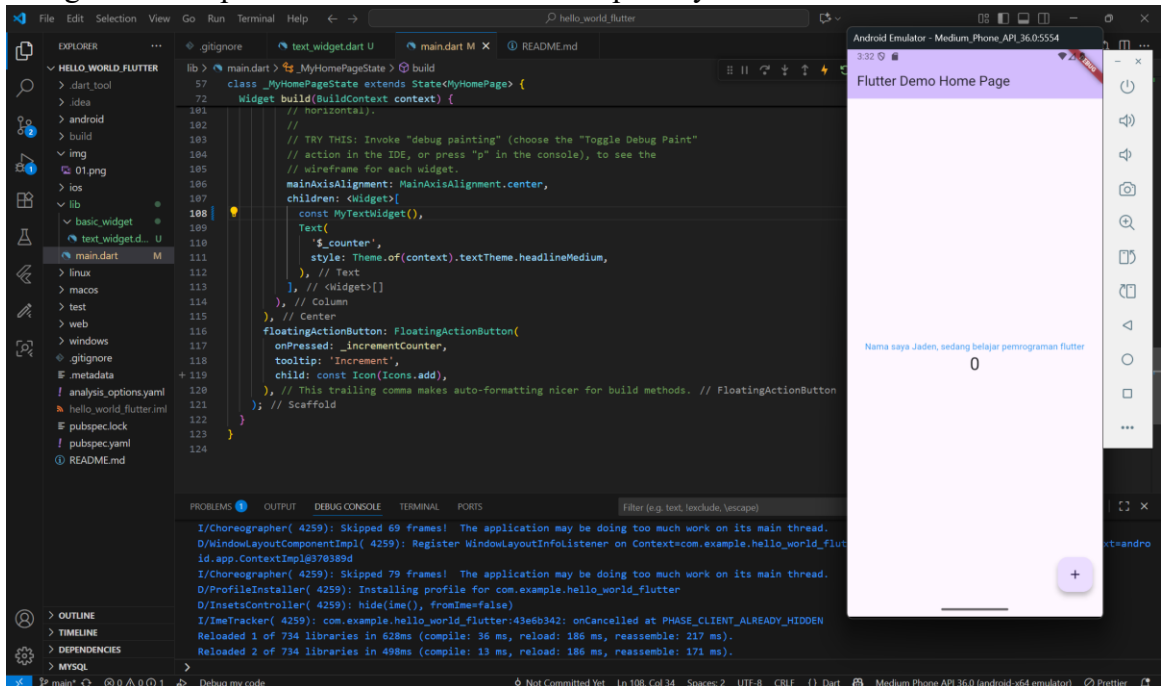
7. Membuat widget image dengan membuat file baru dengan nama image_widget.dart

```dart
lib > basic_widget > image_widget.dart > MyImageWidget
1    import 'package:flutter/material.dart';
2
3    class MyImageWidget extends StatelessWidget {
4      const MyImageWidget({Key? key}) : super(key: key);
5
6      @override
7      Widget build(BuildContext context) {
8        return const Image(
9          image: AssetImage("logo_polinema.jpg")
10       ); // Image
11     }
12  }
```

penyesuaian asset pada file pubspec.yaml dan tambahkan file logo Anda di folder assets project hello_world.

```yaml
89    # see https://flutter.dev/to/font-from-package
90    assets:
91      - assets/Logo_Politeknik_Negeri_Malang.png
```

Hasilnya
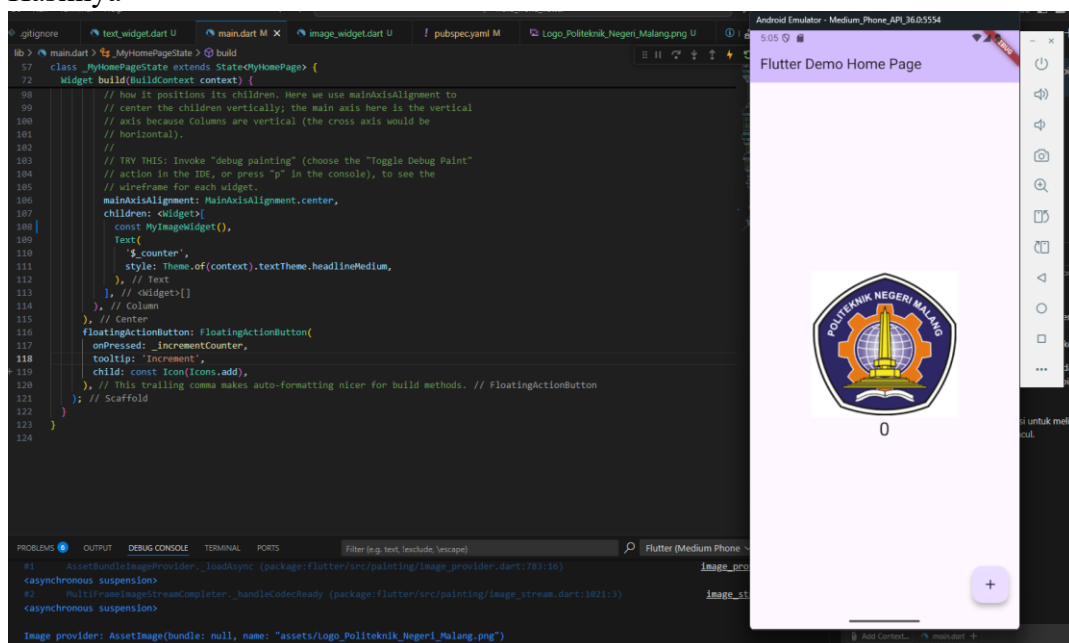
8. Membuat widget Cupertino Button dan Loading Bar

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
        home: Container(
          margin: const EdgeInsets.only(top: 30),
          color: Colors.white,
          child: Column(
            children: <Widget>[
              CupertinoButton(
                child: const Text("Contoh button"),
                onPressed: () {},
              ), // CupertinoButton
              const CupertinoActivityIndicator(),
            ], // <Widget>[]
          ), // Column
        ), // Container
    ); // MaterialApp
  }
}
```

9. Membuat Floating Action Button (FAB)

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        floatingActionButton: FloatingActionButton(
          onPressed: () {
            // Add your onPressed code here!
          },
          child: const Icon(Icons.thumb_up),
          backgroundColor: Colors.pink,
        ), // FloatingActionButton
      ), // Scaffold
    ); // MaterialApp
  }
}
```

10. Mengatur Scaffold widget digunakan untuk mengatur tata letak sesuai dengan material design.

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

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

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        primarySwatch: Colors.red,
      ), // ThemeData
      home: const MyHomePage(title: 'My Increment App'),
    ); // MaterialApp
  }
}

class MyHomePage extends StatefulWidget {
  const MyHomePage({super.key, required this.title});

  final String title;

  @override
  State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  void _incrementCounter() {
    setState(() {
      // This call to setState tells the Flutter framework that something has
      // changed in this State, which causes it to rerun the build method below
      // so that the display can reflect the updated values. If we changed
      // _counter without calling setState(), then the build method would not be
      // called again, and so nothing would appear to happen.
      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: Theme.of(context).colorScheme.inversePrimary,
        title: Text(widget.title),
      ), // AppBar
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            const Text('You have pushed the button this many times:'),
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headlineMedium,
            ), // Text
          ], // <Widget>[]
        ), // Column
      ), // Center
      bottomNavigationBar: BottomAppBar(
        child: Container(
          height: 50.0,
        ), // Container
      ), // BottomAppBar
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: 'Increment',
        child: const Icon(Icons.add),
      ), // FloatingActionButton
      floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,
    ); // Scaffold
  }
}
```

11. Mengubah main untuk Dialog widget pada flutter yang memiliki dua jenis dialog yaitu AlertDialog dan SimpleDialog.

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

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      home: Scaffold(
        body: MyLayout(),
      ), // Scaffold
    ); // MaterialApp
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(8.0),
      child: ElevatedButton(
        child: const Text('Show alert'),
        onPressed: () {
          showAlertDialog(context);
        },
      ), // ElevatedButton
    ); // Padding
  }
}

showAlertDialog(BuildContext context) {
  // set up the button
  Widget okButton = TextButton(
    child: const Text("OK"),
    onPressed: () {
      Navigator.pop(context);
    },
  ); // TextButton

  // set up the AlertDialog
  AlertDialog alert = AlertDialog(
    title: const Text("My title"),
    content: const Text("This is my message."),
    actions: [
      okButton,
    ],
  ); // AlertDialog

  // show the dialog
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return alert;
    },
  );
}
```

12. Membuat input dan Selection Widget

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

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: const Text("Contoh TextField")),
        body: const TextField(
          obscureText: false,
          decoration: InputDecoration(
            border: OutlineInputBorder(),
            labelText: 'Nama',
          ), // InputDecoration
        ) // TextField
      ), // Scaffold
    ); // MaterialApp
  }
}
```

Android Emulator - Medium_Phone_API_36.0:5554

5:38

Contoh TextField

Nama

13. Membuat date and time pickers

```dart
Run | Debug | Profile
4  void main() {
5    runApp(const MyApp());
6  }
7
8  class MyApp extends StatelessWidget {
9    const MyApp({Key? key}) : super(key: key);
10
11   // This widget is the root of your application.
12   @override
13   Widget build(BuildContext context) {
14     return MaterialApp(
15       title: 'Contoh Date Picker',
16       home: MyHomePage(title: 'Contoh Date Picker')
17     ); // MaterialApp
18   }
19 }
```

```dart
65  class MyHomePage extends StatefulWidget {
66    const MyHomePage({Key? key, required this.title}) : super(key: key);
67
68    final String title;
69
70    @override
71    _MyHomePageState createState() => _MyHomePageState();
72  }
73
74  class _MyHomePageState extends State<MyHomePage> {
75    // Variable/State untuk mengambil tanggal
76    DateTime selectedDate = DateTime.now();
77
78    //  Initial SelectDate FLutter
79    Future<void> _selectDate(BuildContext context) async {
80      // Initial DateTime FIinal Picked
81      final DateTime? picked = await showDatePicker(
82        context: context,
83        initialDate: selectedDate,
84        firstDate: DateTime(2015, 8),
85        lastDate: DateTime(2101));
86      if (picked != null && picked != selectedDate) {
87        setState(() {
88          selectedDate = picked;
89        });
90      }
91    }
92
93    @override
94    Widget build(BuildContext context) {
95      return Scaffold(
```

```
 96          appBar: AppBar(
 97            title: Text(widget.title),
 98          ), // AppBar
 99          body: Center(
100            child: Column(
101              mainAxisSize: MainAxisSize.min,
102              children: <Widget>[
103                Text("${selectedDate.toLocal()}".split(' ')[0]),
104                const SizedBox(
105                  height: 20.0,
106                ), // SizedBox
107                ElevatedButton(
108                  onPressed: () => {
109                    _selectDate(context),
110                    // ignore: avoid_print
111                    print(selectedDate.day + selectedDate.month + selectedDate.year)
112                  },
113                  child: const Text('Pilih Tanggal'),
114                ), // ElevatedButton
115              ], // <Widget>[]
116            ), // Column
117          ), // Center
118        ); // Scaffold
119      }
120    }
```