

Tugas-12-Widget & Component

Sebelum tugas mulai dikerjakan, berikut video briefing materi Component.



Pada tugas kali ini, Anda diminta untuk mulai membiasakan diri melakukan coding menggunakan sintaks-sintaks Flutter. Untuk itu, kali ini Anda bisa mengikuti sebuah video tutorial untuk dapat membangun tampilan/UI sederhana dari aplikasi Telegram. Sebelum mulai mengikuti tutorial tersebut, ada beberapa tahapan yang perlu dipersiapkan, antara lain:

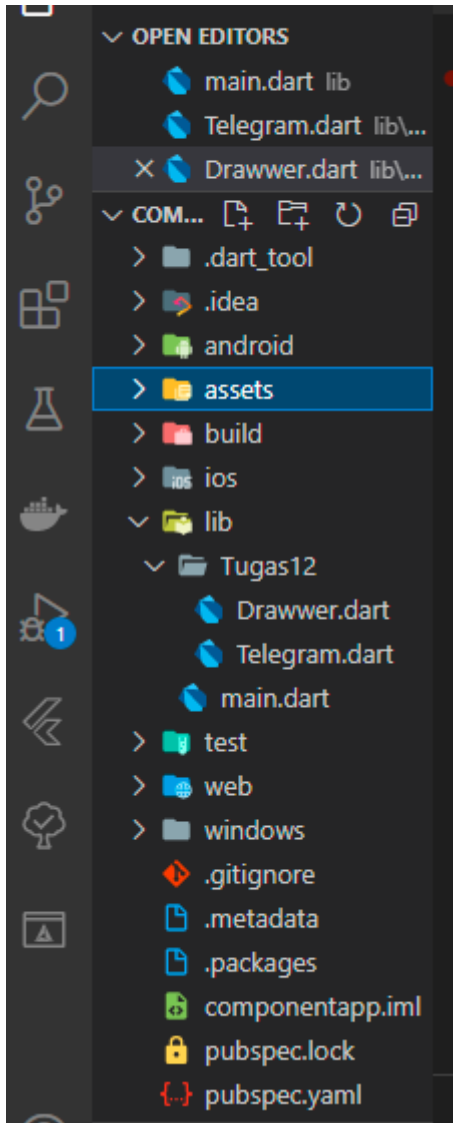
1. Mempersiapkan folder

Dalam mengerjakan tugas ini, silahkan membuat folder "Tugas" terlebih dahulu di dalam folder proyek Flutter yang telah dibuat. Dan di dalam folder "**Tugas**" buatlah folder "**Tugas12**". Dan di dalam folder "**Tugas12**", Anda bisa membuat file **Telegram.dart** dan **DrawerScreen.dart** dan Models untuk dummy datanya. untuk lebih jelas nya ikuti video diatas.

2. Mendownload data yang diperlukan

pada tutorial ini ada beberapa file yaitu models untuk dummy datanya dan juga foto profile, untuk foto profile gunakan foto masing-masing di taruh di folder assets/img, lalu untuk file models dapat di download di link berikut ini :

https://drive.google.com/drive/folders/1tf8h_AtzO6WqwNhi4yY0xNOPhNDaEVXd?usp=sharing.



Tutorial Membuat UI Aplikasi Telegram

Berikut tutorial membangun UI aplikasi mobile Telegram. Ikuti langkah-langkah yang digunakan di video tersebut dan pastikan code yang diikuti dapat dijalankan pada perangkat Android Anda.



Dan Bagi Teman-teman yang tidak bisa mengikuti video tutorial, sobat sanber dapat mengikuti coding berikut ini :

main.dart

```

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: Telegram(),
    );
  }
}

```

Telegram.dart

```

Telegram.dart U X

lib > Tugas > Tugas12 > Telegram.dart > _TelegramState > build
1  import 'package:flutter/material.dart';
2  import 'package:sanberappflutter/Tugas/Tugas12/DrawerScreen.dart';
3  import 'package:sanberappflutter/Tugas/Tugas12/Models/Chart_model.dart';
4
5  class Telegram extends StatefulWidget {
6    @override
7    _TelegramState createState() => _TelegramState();
8  }
9
10 class _TelegramState extends State<Telegram> {
11   @override
12   Widget build(BuildContext context) {
13     return Scaffold(
14       appBar: AppBar(title: Text("Telegram"),
15         actions: <Widget>[
16           Padding(padding: const EdgeInsets.all(8.0),
17             child: Icon(Icons.search),
18             ) // Padding
19         ], // <Widget>[]
20       ), // AppBar
21       drawer: DrawerScreen(),
22       body: ListView.separated(
23         itemBuilder: (ctx, i){
24           return ListTile(
25             leading: CircleAvatar(
26               radius: 28,

```

```

29         title: Text(items[i].name, style: TextStyle(fontWeight: FontWeight.bold)),
30         subtitle: Text(items[i].message),
31         trailing: Text(items[i].time),
32     ); // ListTile
33 }
34 separatorBuilder: (ctx, i) {
35     return Divider();
36 }, itemCount: items.length), // ListView.separated
37 floatingActionButton: FloatingActionButton(
38     child: Icon(Icons.create, color: Colors.white),
39     backgroundColor: Color(0xFF65a9e0),
40     onPressed: () {},
41 ), // FloatingActionButton
42 ); // Scaffold
43 }
44 }

```

DrawerScreen.dart

```

1  import 'package:flutter/cupertino.dart';
2  import 'package:flutter/material.dart';
3
4  class DrawerScreen extends StatefulWidget {
5      @override
6      _DrawerScreenState createState() => _DrawerScreenState();
7  }
8
9  class _DrawerScreenState extends State<DrawerScreen> {
10     @override
11     Widget build(BuildContext context) {
12         return Drawer(
13             child: ListView(
14                 children: <Widget>[
15                     UserAccountsDrawerHeader(
16                         accountName: Text("Achmad Hilmy Firdaus"),
17                         currentAccountPicture: CircleAvatar(
18                             backgroundImage: AssetImage("assets/img/hilmy.jpg")
19                         ), // CircleAvatar
20                         accountEmail: Text("achmad.hilmy@sanbercode.com"),
21                     ), // UserAccountsDrawerHeader
22                     DrawerListTile(
23                         iconData : Icons.group,
24                         title: "NewGroup",
25                         onTapPressed: () {},
26                     ), // DrawerListTile

```

```

30         onPressed: () {},
31     ), // DrawerListTile
32     DrawerListTile(
33       iconData : Icons.notifications,
34       title: "New Channel Chat",
35       onPressed: () {},
36     ), // DrawerListTile
37     DrawerListTile(
38       iconData : Icons.contacts,
39       title: "contacts",
40       onPressed: () {},
41     ), // DrawerListTile
42     DrawerListTile(
43       iconData : Icons.bookmark_border,
44       title: "Saved Message",
45       onPressed: () {},
46     ), // DrawerListTile
47     DrawerListTile(
48       iconData : Icons.phone,
49       title: "Calls",
50       onPressed: () {},
51   ) // DrawerListTile

```

```

52   ], // <Widget>[]
53 ) // ListView
54 ); // Drawer
55 }
56 }
57
58 class DrawerListTile extends StatelessWidget {
59   final IconData iconData;
60   final String title;
61   final VoidCallback onPressed;
62
63   const DrawerListTile({Key key, this.iconData, this.title, this.onPressed})
64     :super(key:key);
65   @override
66   Widget build(BuildContext context) {
67     return ListTile(
68       onTap: onPressed,
69       dense: true,
70       leading: Icon(iconData),
71       title: Text(title, style: TextStyle(fontSize: 16),),
72     ); // ListTile
73   }
74 }

```

Model/Chart_Models.dart (Sudah ada tinggal download)

```

3   final String message;
4   final String time;
5   final String profileUrl;
6
7   ChartModel({this.name, this.message, this.time, this.profileUrl});
8 }
9 final List<ChartModel>items = [
10  ChartModel(name: 'hilmy', message: 'Hello Hilmy', time: '12.00', profileUrl: 'https://asset.kompas.com/crops/IjmmRwybxe-4
11  ChartModel(name: 'riska', message: 'hello riska', time: '9 march', profileUrl: 'https://img.okezone.com/content/2019/04/3
12  ChartModel(name: 'vita', message: 'hello vita', time: '10 march', profileUrl: 'https://asset.kompas.com/crops/ployX7cQ0qs
13  ChartModel(name: 'niva', message: 'hello niva', time: '12.21', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/23/
14  ChartModel(name: 'citra', message: 'hello citra', time: '13.41', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/2
15  ChartModel(name: 'adi', message: 'hello adi', time: '25 january', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/
16  ChartModel(name: 'ashraf', message: 'hello ashraf', time: '19.00', profileUrl: 'https://img-o.okeinfo.net/content/2020/04
17  ChartModel(name: 'niva', message: 'hello niva', time: '12.21', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/23/
18  ChartModel(name: 'citra', message: 'hello citra', time: '13.41', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/2
19  ChartModel(name: 'adi', message: 'hello adi', time: '25 january', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/
20  ChartModel(name: 'ashraf', message: 'hello ashraf', time: '19.00', profileUrl: 'https://img-o.okeinfo.net/content/2020/04
21  ChartModel(name: 'niva', message: 'hello niva', time: '12.21', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/23/
22  ChartModel(name: 'citra', message: 'hello citra', time: '13.41', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/2
23  ChartModel(name: 'adi', message: 'hello adi', time: '25 january', profileUrl: 'https://img-o.okeinfo.net/content/2020/04/
24  ChartModel(name: 'ashraf', message: 'hello ashraf', time: '19.00', profileUrl: 'https://img-o.okeinfo.net/content/2020/04
25 ];

```

pubspec.yaml

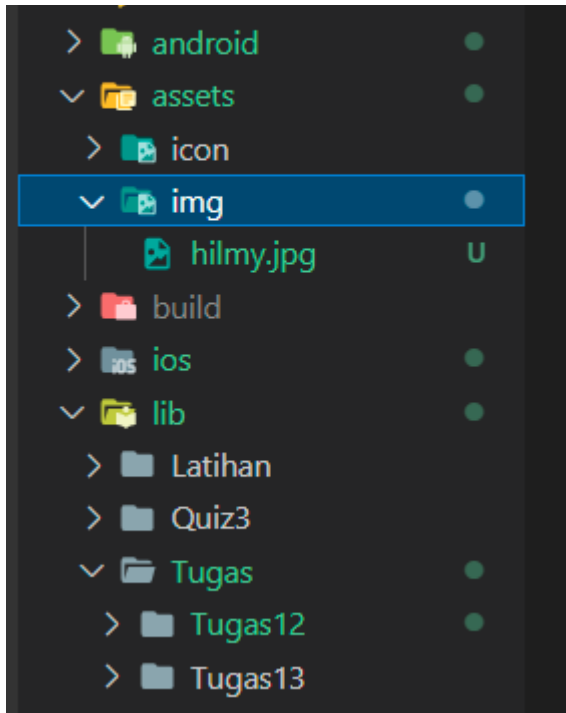
```

Telegram.dart U  Chart_model.dart U  pubspec.yaml U x
pubspec.yaml
33  flutter_test:
34  |   sdk: flutter
35
36  # For information on the generic Dart part of this file, see the
37  # following page: https://dart.dev/tools/pub/pubspec
38
39  # The following section is specific to Flutter.
40  flutter:
41
42  # The following line ensures that the Material Icons font is
43  # included with your application, so that you can use the icons in
44  # the material Icons class.
45  uses-material-design: true
46
47  # To add assets to your application, add an assets section, like this:
48  assets:
49  |   - assets/img/
50  #   - images/a_dot_ham.jpeg
51
52  # An image asset can refer to one or more resolution-specific "variants", see
53  # https://flutter.dev/assets-and-images/#resolution-aware.
54
55  # For details regarding adding assets from package dependencies, see
56  # https://flutter.dev/assets-and-images/#from-packages
57

```

uncoment assets ubah jadi asstes/img

assets/img



untuk foto silahkan menggunakan foto masing-masing peserta

Jika tutorial tersebut sudah diikuti sampai selesai, selanjutnya silahkan push hasil pekerjaan Anda ke Gitlab. Dan jangan lupa untuk mengumpulkan link commit hasil pekerjaan Anda ke member.jabarcodingcamp.id.

Rating - Feedback

Berikan Rating pada posting ini:



Berikan kritik dan saran..

Submit