

Tugas-14-Rest Api & State

Ikutilah langkah-langkah pengerjaan di bawah ini:

1. Menambahkan folder baru di dalam folder Tugas

buatlah folder baru dengan nama "Tugas14" di dalam folder Tugas di dalam project bootcampflutter

2. kerjakan implementasi rest api di bawah ini

implementasi kan REST API di dalamnya dengan menggunakan rest api online untuk method nya ada 3 berikut link untuk implementasinya

1. Get untuk meminta data dari server

GET: <https://achmadhilmy-sanbercode.my.id/api/v1/news>

2. Post untuk mengembalikan data dari local ke server

POST: <https://achmadhilmy-sanbercode.my.id/api/v1/news>

isi dari **post** :

```
{  
  
  "title" : "ppkm",  
  
  "value" : "ppkm jilid 2 untuk mengatasi penyebaran covid 19",  
  
}
```

3. Delete untuk menghapus data

DELETE: https://achmadhilmy-sanbercode.my.id/api/v1/news/{id_news}

example : <https://achmadhilmy-sanbercode.my.id/api/v1/news/1>

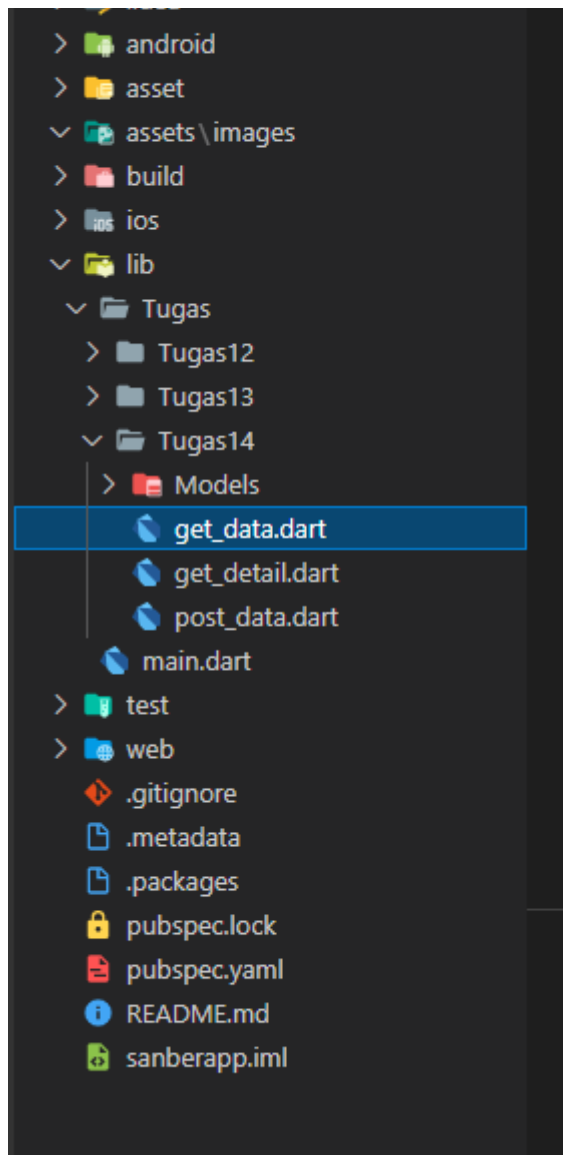
lalu silahkan ikuti codingan ini disini hanya 2 method yaitu get dan post untuk delete nya tidak perlu di implementasikan, untuk menghapus data bila sudah overload terima kasih

Video Tugas Silahkan di ikuti dan jika tidak ingin lihat video ada screenshoot tinggal di ketik ulang:

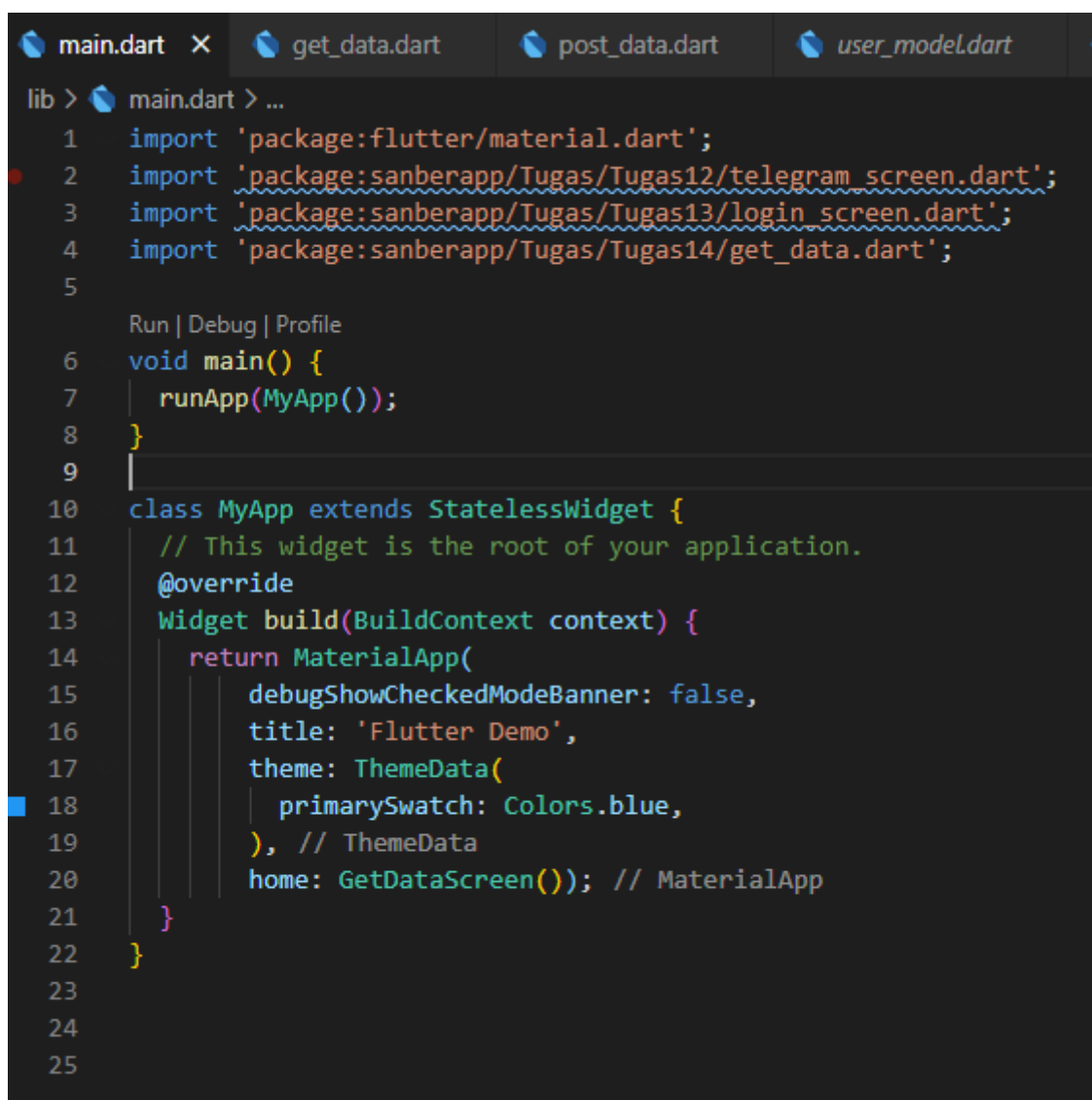
https://youtu.be/5ys9QfPcxMc?list=PLM_B8pNCcXu0wEvjig1_lzJkdU6wj2lfx

pubspec.yaml

```
environment:  
  sdk: ">=2.12.0 <3.0.0"  
  
dependencies:  
  cupertino_icons: ^1.0.2  
  flutter:  
    sdk: flutter  
  http: ^0.13.3  
  
dev_dependencies:  
  flutter_test:  
    sdk: flutter
```



main.dart



get_data.dart


```

78
79         // data![index]['nama_kegiatan'],
80         style: TextStyle(
81           fontSize: 14,
82           fontWeight: FontWeight.w700), // TextStyle
83       ), // Text
84       Icon(Icons.chevron_right),
85     ],
86   ), // Row
87   // Text(filterBantuanList
88   //       .data![index].deskripsi!),
89   SizedBox(
90     height: 3,
91   ), // SizedBox
92   Divider(
93     height: 20,
94     thickness: 1,
95     endIndent: 25,
96     indent: 15,
97   ) // Divider
98 ], // <Widget>[]
99 ), // Column
100 ), // Padding
101 ) // GestureDetector
102 ],
103 ), // Column
104 ); // Container
105 })), // ListView.builder
106 ), // RefreshIndicator

```

```

107     floatingActionButton: FloatingActionButton(
108       onPressed: () {
109         Navigator.push(
110           context, MaterialPageRoute(builder: (context) => PostDataApi()));
111       },
+ 112       child: Icon(Icons.add),
113     ), // FloatingActionButton
114   ); // Scaffold
115 }
116 }
117

```

get_detail.dart

```

main.dart  get_data.dart  get_detail.dart ×  post_data.dart  int.dart
lib > Tugas > Tugas14 > get_detail.dart > _getDetailScreenState > build
1  import 'package:flutter/material.dart';
2
3  class getDetailScreen extends StatefulWidget {
4    List? value;
5
6    getDetailScreen({Key? key, required this.value}) : super(key: key);
7
8    @override
9    _getDetailScreenState createState() => _getDetailScreenState(value);
10 }
11
12 class _getDetailScreenState extends State<getDetailScreen> {
13   List? value;
14   _getDetailScreenState(this.value);
15   @override
16   Widget build(BuildContext context) {
17     return Scaffold(
18       appBar: AppBar(
19         title: Text("detail screen news"),
20       ), // AppBar

```

```

24     crossAxisAlignment: CrossAxisAlignment.start,
25     children: <Widget>[
26       Text(
27         value![0],
28         style: TextStyle(fontSize: 16, fontWeight: FontWeight.w700),
29       ), // Text
30       SizedBox(
31         height: 12,
32       ), // SizedBox
33       Text(
34         value![1],
35         style: TextStyle(fontSize: 14, fontWeight: FontWeight.w400),
36       ), // Text
37     ], // <Widget>[]
38   ), // Column
39 ); // Container // Scaffold
40 }
41 }
42

```

user_model.dart

```

main.dart  get_data.dart  get_detail.dart  user_model.dart X  post_data.dart
lib > Tugas > Tugas14 > Models > user_model.dart > UserModel > UserModel
1  import 'dart:convert';
2
3  UserModel userModelFromJson(String str) => UserModel.fromJson(json.decode(str));
4  String userModelToJson(UserModel data) => json.encode(data.toJson());
5
6  class UserModel {
7    UserModel({
8      required this.title,
9      required this.value,
10     required this.id,
11     required this.createdAt,
12   });
13   String title;
14   String value;
15
16   String id;
17   DateTime createdAt;
18
19   factory UserModel.fromJson(Map<String, dynamic> json) => UserModel(
20     title: json["title"],
21     value: json["value"],
22     id: json["id"],
23     createdAt: DateTime.parse(json["createdAt"]),
24   );
25   Map<String, dynamic> toJson() => {
26     "title": title,
27     "value": value,
28     "id": id,
29     "createdAt": createdAt.toIso8601String(),
30   };
31 }
32

```

post_data.dart

```

1  import package:flutter/material.dart;
2  import 'package:sanberapp/Tugas/Tugas14/get_data.dart';
3  import 'Get_data.dart';
4  import 'Models/user_model.dart';
5  import 'package:http/http.dart' as http;
6
7  class PostDataApi extends StatefulWidget {
8    PostDataApi({
9      Key? key,
10     }) : super(key: key);
11
12     @override
13     _PostDataApiState createState() => _PostDataApiState();
14   }
15

```

```

15
16  class Dialogs {
17    static Future<void> showLoadingDialog(
18      BuildContext context, GlobalKey key) async {
19      return showDialog<void>(
20        context: context,
21        barrierDismissible: false,
22        builder: (BuildContext context) {
23          return new WillPopScope(
24            onWillPop: () async => false,
25            child: SimpleDialog(
26              // key: key,
27              backgroundColor: Colors.black54,
28              children: <Widget>[
29                Center(
30                  child: Column(children: [
31                    CircularProgressIndicator(),
32                    SizedBox(
33                      height: 10,
34                    ), // SizedBox
35                    Text(
36                      "Please Wait....",
37                      style: TextStyle(color: Colors.blueAccent),
38                    ) // Text
39                  ]), // Column
40                ) // Center
41              ])); // <Widget>[] // SimpleDialog // WillPopScope
42        });
43    }
44  }
45

```



```

49 final TextEditingController valueController = TextEditingController();
50 void _submitted(BuildContext context) async {
51   final String title = titleController.text;
52   final String value = valueController.text;
53   final UserModel? user = await createUser(title, value);
54   setState(() {
55     _user = user;
56   });
57   Navigator.pop(context);
58 }
59
60 Future<UserModel?> createUser(
61   String title,
62   String value,
63 ) async {
64   try {
65     Dialogs.showLoadingDialog(context, _keyLoader); //invoking login
66     var apiUrl =
67       Uri.parse("https://achmadhilmy-sanbercode.my.id/api/v1/news");
68     final response =
69       await http.post(apiUrl, body: {"title": title, "value": value});
70     if (response.statusCode == 201) {
71       final String responseString = response.body;
72       return userModelFromJson(responseString);
73     }
74     Navigator.of(context).pop(context);
75   } catch (e) {
76     print(e.toString());
77   }
78 }
79

```

```

79
80 final GlobalKey<State> _keyloader = new GlobalKey<State>();
81 @override
82 Widget build(BuildContext context) {
83   return Scaffold(
84     appBar: AppBar(
85       title: Text("Post Data"),
86     ), // AppBar
87     body: Container(
88       padding: EdgeInsets.all(32),
89       child: Column(
90         children: <Widget>[
91           TextField(
92             controller: titleController,
93             decoration: InputDecoration(
94               border: OutlineInputBorder(), labelText: "title"), // InputDecoration
95           ), // TextField
96           SizedBox(height: 10),
97           TextField(
98             controller: valueController,
99             decoration: InputDecoration(
100               border: OutlineInputBorder(), labelText: "Value"), // InputDecoration
101             maxLines: 5,
102           ), // TextField
103           SizedBox(height: 10),
104           _user == null
105             ? Container()
106             : Text(
107               "the user ${_user!.title} is created, and id ${_user!.id}"), // Text
108         ], // <Widget>[]
109
110         //Text("the user ${_user.name} is created, and id ${_user.id}"),
111       )), // Column // Container
112     floatingActionButton: FloatingActionButton(
113       onPressed: () => _submitted(context),
114       tooltip: 'Increment',
115       child: Icon(Icons.add),
116     ), // This trailing comma makes auto-formatting nicer for build methods. // FloatingActionButton
117   ); // Scaffold

```

```
+ 115         child: Icon(Icons.add),
116     ), // This trailing comma makes auto-formatting nicer for build methods. // FloatingActionButton
117 ); // Scaffold
118 }
119 }
120
```

Rating - Feedback

Berikan Rating pada posting ini:



Berikan kritik dan saran..

Submit