1. Lakukan POST request dengan menggunakan DIO untuk mengirimkan data contact! gunakan url pada slide 12.

Untuk mengakses halaman dari Post Request bisa melalui drawer screen dengan title Post Contact

* Dio.dart

import 'package:dio/dio.dart';

class DioService {

final dio = Dio();

PostContacts(String id, String name, String phone) async {

Response response;

response = await dio.post(

'https://my-json-server.typicode.com/hadihammurabi/flutter-webservice/contacts',

queryParameters: {'id': id, 'name': name, 'phone': phone},

);

if (response.statusCode == 201) {

print(response.statusCode.toString());

print(response.data.toString());

} else {

print('Gagal post contact');

}

}

* FormPost.dart

import 'package:flutter/material.dart';

import 'package:flutter\_platform\_widget/Intro-AdvanceRestAPI/Dio.dart';

class FormContactsAPI extends StatefulWidget {

const FormContactsAPI({Key? key}) : super(key: key);

@override

State<FormContactsAPI> createState() => \_FormContactsAPI();

}

class \_FormContactsAPI extends State<FormContactsAPI> {

final \_formKey = GlobalKey<FormState>();

final TextEditingController idController = TextEditingController();

final TextEditingController nameController = TextEditingController();

final TextEditingController phoneController = TextEditingController();

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Contacts'),

centerTitle: true,

),

body: Form(

key: \_formKey,

child: Column(

children: [

TextFormField(

controller: idController,

validator: (value) {

if (value!.isEmpty) {

return 'ID tidak boleh kosong';

}

},

decoration: const InputDecoration(

labelText: 'ID',

),

),

TextFormField(

controller: nameController,

validator: (value) {

if (value!.isEmpty) {

return 'Name tidak boleh kosong';

}

},

decoration: const InputDecoration(

labelText: 'Name',

),

),

TextFormField(

controller: phoneController,

validator: (value) {

if (value!.isEmpty) {

return 'Phone tidak boleh kosong';

}

},

decoration: const InputDecoration(

labelText: 'Phone',

),

),

ElevatedButton(

onPressed: () async {

if (\_formKey.currentState!.validate()) {

DioService().PostContacts(idController.text,

nameController.text, phoneController.text);

}

},

child: const Text('Submit')),

],

),

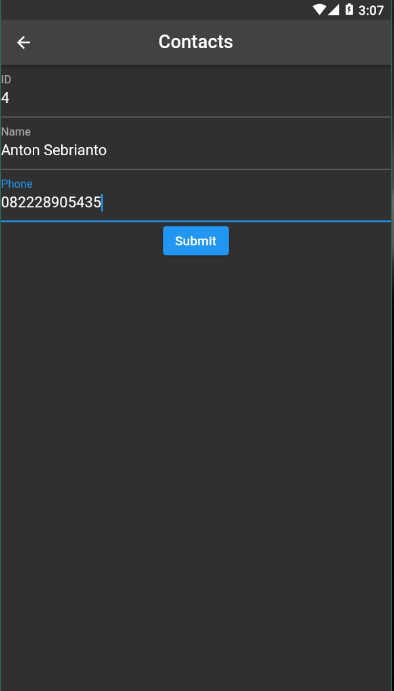
),

);

}

}

* Output :



Post data berhasil dan menampilkan log berikut



1. Ubahlah bentuk JSON berikut kedalam bentuk Object dari suatu class! Gunakan data JSON pada URL berikut:<https://my-json-server.typicode.com/hadihammurabi/flutter-webservice/contacts/2>

Untuk mengakses halaman dari JSON to Object melalui drawer screen dengan title Contact API

* ContactAPI.dart

import 'package:dio/dio.dart';

import 'package:flutter/material.dart';

import 'package:flutter/src/foundation/key.dart';

import 'package:flutter/src/widgets/framework.dart';

import 'package:flutter\_platform\_widget/Intro-AdvanceRestAPI/ContactModel.dart';

class ContactAPI extends StatefulWidget {

const ContactAPI({Key? key}) : super(key: key);

@override

State<ContactAPI> createState() => \_ContactAPIState();

}

class \_ContactAPIState extends State<ContactAPI> {

ContactModel? contact;

@override

void initState() {

JSONtoObject();

super.initState();

}

JSONtoObject() async {

Response response;

response = await Dio().get(

'https://my-json-server.typicode.com/hadihammurabi/flutter-webservice/contacts/2');

if (response.statusCode == 200) {

print(response.data);

contact = ContactModel.fromJson(response.data);

setState(() {});

print(contact!.id);

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Contact'),

),

body: Column(

children: [

Text('id : ${contact!.id.toString()}'),

Text('name : ${contact!.name.toString()}'),

Text('phone : ${contact!.phone.toString()}'),

],

),

);

}

}

* Output :



Log dari JSON to Object



1. Lakukan PUT request dengan menggunakan DIO. Berikut adalah kriteria request yang harus dipenuhi:
2. endpoint yang digunakan: <https://jsonplaceholder.typicode.com/posts/1>
3. request body yang digunakan:

{

id: 1,

title: 'foo',

body: 'bar',

userId: 1,

}

Untuk mengakses halaman dari Put Request melalui drawer screen dengan titile Put Request

* Dio.dart

PutRequestDio(int userId, String title, String body) async {

Response response;

response = await dio.put(

'https://jsonplaceholder.typicode.com/posts/1',

data: {

'title': title,

'body': body,

'userId': userId,

},

);

if (response.statusCode == 200) {

print(response.data.toString());

}

}

* FormPut.dart

import 'package:flutter/material.dart';

import 'package:flutter\_platform\_widget/Intro-AdvanceRestAPI/Dio.dart';

class FormPut extends StatefulWidget {

const FormPut({Key? key}) : super(key: key);

@override

State<FormPut> createState() => \_FormPut();

}

class \_FormPut extends State<FormPut> {

final \_formKey = GlobalKey<FormState>();

final TextEditingController titleController = TextEditingController();

final TextEditingController bodyController = TextEditingController();

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Contacts'),

centerTitle: true,

),

body: Form(

key: \_formKey,

child: Column(

children: [

TextFormField(

controller: titleController,

validator: (value) {

if (value!.isEmpty) {

return 'Title tidak boleh kosong';

}

},

decoration: const InputDecoration(

labelText: 'Title',

),

),

TextFormField(

controller: bodyController,

validator: (value) {

if (value!.isEmpty) {

return 'Body tidak boleh kosong';

}

},

decoration: const InputDecoration(

labelText: 'Body',

),

),

ElevatedButton(

onPressed: () async {

if (\_formKey.currentState!.validate()) {

DioService().PutRequestDio(

1, titleController.text, bodyController.text);

}

},

child: const Text('Submit')),

],

),

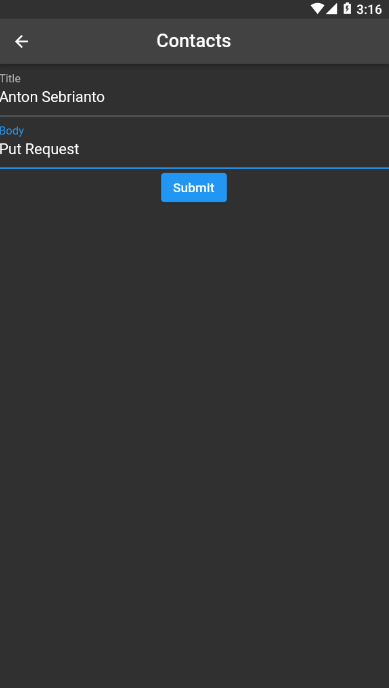
),

);

}

}

* Output



Log dari put request

