

Deklarasi class Contact yang memerlukan 2 parameter yaitu name dan phone

'toJson' merupakan sebuah metode untuk mengonversi objek 'Contact' ke bentuk Map yang berisi data 'name' dan 'phone'



```
lib > post_request.dart > ...
1 import 'dart:convert';
2 import 'package:dio/dio.dart';
3
4 //Deklarasi class Contact yang memerlukan 2 parameter yaitu name dan phone
5 class Contact{
6   final String name;
7   final String phone;
8
9   Contact({required this.name, required this.phone});
10
11 //toJson merupakan sebuah metode untuk mengonversi objek 'Contact' ke bentuk Map yang berisi data 'name' dan 'phone'
12 Map<String, dynamic> toJson(){
13   return{
14     'name': name,
15     'phone': phone,
16   };
17 }
18 }
19
20 //Membuat objek Dio
21 Dio dio= Dio();
22
23 //Endpoint URL
24 String url = 'http://localhost:3000/contacts';
25
26 Future<void> postData() async {
27   try {
28     Response response = dio.post(url, data: toJson());
29     print('Response: ${response.statusCode}');
30     print('Data: ${response.data}');
31   } catch (e) {
32     print('Error: $e');
33   }
34 }
```

PROBLEMS OUTPUT **DEBUG CONSOLE** TERMINAL

Response: 201
Data: {name: John Smith, phone: 123456789, id: 4}
Exited

Ln 20, Col 16 Spaces: 2 UTF-8 CRLF {} Dart

Metode untuk mengirim data contact melalui POST request

Dideklarasikan fungsi 'postData' dengan menggunakan tipe data 'Future<void>' yang artinya fungsi ini mengembalikan objek Future void

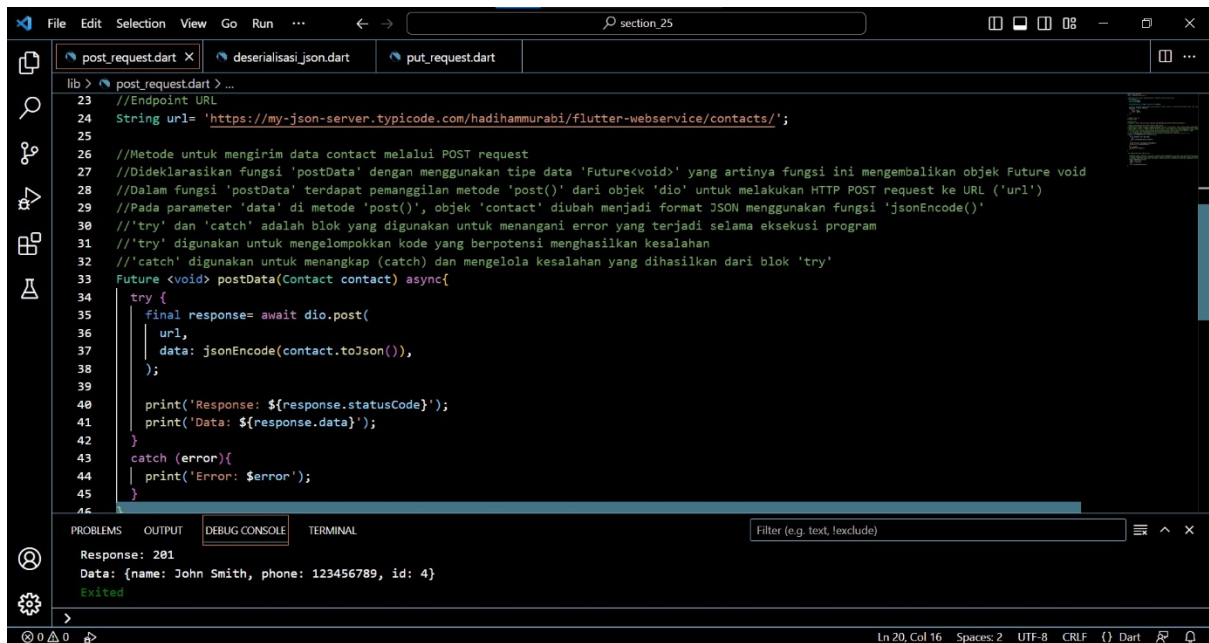
Dalam fungsi 'postData' terdapat pemanggilan metode 'post()' dari objek 'dio' untuk melakukan HTTP POST request ke URL ('url')

Pada parameter 'data' di metode 'post()', objek 'contact' diubah menjadi format JSON menggunakan fungsi 'jsonEncode()'

'try' dan 'catch' adalah blok yang digunakan untuk menangani error yang terjadi selama eksekusi program

'try' digunakan untuk mengelompokkan kode yang berpotensi menghasilkan kesalahan

'catch' digunakan untuk menangkap (catch) dan mengelola kesalahan yang dihasilkan dari blok 'try'



```
lib > post_request.dart > ...
23 //Endpoint URL
24 String url= 'https://my-json-server.typicode.com/hadihamurabi/flutter-webservice/contacts/';
25
26 //Metode untuk mengirim data contact melalui POST request
27 //Dideklarasikan fungsi 'postData' dengan menggunakan tipe data 'Future<void>' yang artinya fungsi ini mengembalikan objek Future void
28 //Dalam fungsi 'postData' terdapat pemanggilan metode 'post()' dari objek 'dio' untuk melakukan HTTP POST request ke URL ('url')
29 //Pada parameter 'data' di metode 'post()', objek 'contact' diubah menjadi format JSON menggunakan fungsi 'jsonEncode()'
30 // 'try' dan 'catch' adalah blok yang digunakan untuk menangani error yang terjadi selama eksekusi program
31 // 'try' digunakan untuk mengelompokkan kode yang berpotensi menghasilkan kesalahan
32 // 'catch' digunakan untuk menangkap (catch) dan mengelola kesalahan yang dihasilkan dari blok 'try'
33 Future<void> postData(Contact contact) async{
34   try {
35     final response= await dio.post(
36       url,
37       data: jsonEncode(contact.toJson()),
38     );
39
40     print('Response: ${response.statusCode}');
41     print('Data: ${response.data}');
42   }
43   catch (error){
44     print('Error: $error');
45   }
46 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Response: 201
Data: {name: John Smith, phone: 123456789, id: 4}
Exited

Ln 20, Col 16 Spaces: 2 UTF-8 CRLF {} Dart

Pembuatan objek 'newContact' dari kelas 'Contact' dengan menggunakan constructor yang telah didefinisikan sebelumnya

Terdapat pemanggilan fungsi 'postData' dengan argumen 'newContact' sebagai data yang akan dikirim dalam POST request

'postData' akan mengirimkan objek 'newContact' dalam bentuk data JSON melalui metode POST menggunakan DIO ke URL yang telah didefinisikan



```
lib > post_request.dart > main
43 catch (error){
44   print('Error: $error');
45 }
46 }
47
48 Run | Debug
49 void main(List<String> args) async{
50   //Pembuatan objek 'newContact' dari kelas 'Contact' dengan menggunakan constructor yang telah didefinisikan sebelumnya
51   //Terdapat pemanggilan fungsi 'postData' dengan argumen 'newContact' sebagai data yang akan dikirim dalam POST request
52   // 'postData' akan mengirimkan objek 'newContact' dalam bentuk data JSON melalui metode POST menggunakan DIO ke URL yang telah didefinisik
53   Contact newContact= Contact(
54     name: 'John Smith',
55     phone: '123456789'
56   );
57   await postData(newContact);
58 }
59
60
61
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Response: 201
Data: {name: John Smith, phone: 123456789, id: 4}
Exited

Ln 57, Col 32 Spaces: 2 UTF-8 CRLF {} Dart