

Muhammad Ikhsan Al Hakim

2311104064

Unguided 15

Main.dart

```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'screens/home_screen.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return GetMaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Data Storage API - GetX',
      home: HomeScreen(),
    );
  }
}
```

Api_service.dart

```
import 'dart:convert';
import 'package:http/http.dart' as http;

class ApiService {
  final String baseUrl = "https://jsonplaceholder.typicode.com";

  Future<List<dynamic>> fetchPosts() async {
    final response = await http.get(Uri.parse('$baseUrl/posts'));
    if (response.statusCode == 200) {
      return json.decode(response.body);
    } else {
      throw Exception('Gagal mengambil data');
    }
  }

  Future<void> createPost() async {
    final response = await http.post(
      Uri.parse('$baseUrl/posts'),
      headers: {'Content-Type': 'application/json'},
      body: json.encode({
```

```

        'title': 'Flutter GetX',
        'body': 'Contoh POST dengan GetX',
        'userId': 1,
      )),
    );

    if (response.statusCode != 201) {
      throw Exception('Gagal menambah data');
    }
  }

  Future<void> updatePost() async {
    final response = await http.put(
      Uri.parse('$baseUrl/posts/1'),
      headers: {'Content-Type': 'application/json'},
      body: json.encode({
        'title': 'Updated Title',
        'body': 'Updated Body',
        'userId': 1,
      })),
    );

    if (response.statusCode != 200) {
      throw Exception('Gagal update data');
    }
  }

  Future<void> deletePost() async {
    final response = await http.delete(Uri.parse('$baseUrl/posts/1'));

    if (response.statusCode != 200) {
      throw Exception('Gagal hapus data');
    }
  }
}

```

Post_controller.dart

```

import 'package:get/get.dart';
import '../services/api_service.dart';

class PostController extends GetxController {
  final ApiService apiService = ApiService();

  var posts = <dynamic>[].obs;
  var isLoading = false.obs;

```

```

Future<void> getPosts() async {
  isLoading.value = true;
  try {
    posts.value = await apiService.fetchPosts();
    Get.snackbar('Sukses', 'Data berhasil diambil');
  } catch (e) {
    Get.snackbar('Error', e.toString());
  } finally {
    isLoading.value = false;
  }
}

Future<void> addPost() async {
  try {
    await apiService.createPost();
    Get.snackbar('Sukses', 'Data berhasil ditambahkan');
  } catch (e) {
    Get.snackbar('Error', e.toString());
  }
}

Future<void> updatePost() async {
  try {
    await apiService.updatePost();
    Get.snackbar('Sukses', 'Data berhasil diperbarui');
  } catch (e) {
    Get.snackbar('Error', e.toString());
  }
}

Future<void> deletePost() async {
  try {
    await apiService.deletePost();
    posts.removeWhere((post) => post['id'] == 1);
    Get.snackbar('Sukses', 'Data berhasil dihapus');
  } catch (e) {
    Get.snackbar('Error', e.toString());
  }
}

```

Home_screen.dart

```

import 'package:flutter/material.dart';
import 'package:get/get.dart';
import '../controllers/post_controller.dart';

class HomeScreen extends StatelessWidget {
  final PostController controller = Get.put(PostController());

```

```
HomeScreen({super.key});
```

```
@override
```

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: const Text('Data Storage API - GetX'),  
      backgroundColor: Colors.orange,  
    ),  
    body: Padding(  
      padding: const EdgeInsets.all(12.0),  
      child: Column(  
        children: [  
          Obx() {  
            if (controller.isLoading.value) {  
              return const CircularProgressIndicator();  
            }  
  
            return Expanded(  
              child: ListView.builder(  
                itemCount: controller.posts.length,  
                itemBuilder: (context, index) {  
                  return Card(  
                    child: ListTile(  
                      title: Text(  
                        controller.posts[index]['title'],  
                        style: const TextStyle(fontSize: 12),  
                      ),  
                      subtitle: Text(  
                        controller.posts[index]['body'],  
                        style: const TextStyle(fontSize: 11),  
                      ),  
                    ),  
                  ),  
                ],  
              ),  
            ),  
          ),  
          const SizedBox(height: 10),  
          Wrap(  
            spacing: 8,  
            children: [  
              ElevatedButton(  
                onPressed: controller.getPosts,  
                child: const Text('GET'),  
              ),  
              ElevatedButton(  
                onPressed: controller.addPost,  
                style: ElevatedButton.styleFrom(  

```

```
        backgroundColor: Colors.green,
      ),
      child: const Text('POST'),
    ),
    ElevatedButton(
      onPressed: controller.updatePost,
      style: ElevatedButton.styleFrom(backgroundColor: Colors.blue),
      child: const Text('UPDATE'),
    ),
    ElevatedButton(
      onPressed: controller.deletePost,
      style: ElevatedButton.styleFrom(backgroundColor: Colors.red),
      child: const Text('DELETE'),
    ),
  ],
),
],
),
),
);
}
```

Hasil Run

