fetch_test.dart

```dart
import 'dart:convert';

import 'dart:developer';

import 'package:flutter/material.dart';

import 'package:flutter_api_fetch_paddictyt/mytest/docs/album.dart';

import 'package:flutter_api_fetch_paddictyt/mytest/docs/backend.dart';

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


class FetchTest extends StatefulWidget {

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



  @override

  State<FetchTest> createState() => _FetchTestState();

}


class _FetchTestState extends State<FetchTest> {

  late Album album;

  bool circular = true;



  getData() async {

    var temp = await fetchAlbum();

    setState(() {

      album = temp;

      circular = false;

    });

    return album;

  }
```

```dart
@override
void initState() {
  super.initState();
  getData();
}


@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Center(
      child: circular ? CircularProgressIndicator() : Text(album.title),
    ),
  );
}
}
```

backend.dart

```dart
import 'dart:convert';

import 'package:flutter_api_fetch_paddictyt/mytest/docs/album.dart';
import 'package:http/http.dart' as http;

Future<Album> fetchAlbum() async {
  String URL = 'https://jsonplaceholder.typicode.com/albums/1';
  final response = await http.get(Uri.parse(URL));

  if (response.statusCode == 200) {
    print(response.body);
    final json = jsonDecode(response.body);
    return Album.fromJson(json);
  } else {
    print('connection error');
  }

  return Album(userId: 1, id: 1, title: 'null');
}
```

album.dart

```dart
class Album {
  final int userId;
  final int id;
  final String title;

  Album({required this.userId, required this.id, required this.title});

  factory Album.fromJson(Map<String, dynamic> json) {
    return Album(
      userId: json['userId'] as int,
      id: json['id'] as int,
      title: json['title'] as String,
    );
  }
}
```