demonstrate the use of RESTAPI:

Code:

Main.dart:

```dart
import 'package:flutter/material.dart';
import 'package:restkedar/user.dart';
import 'package:restkedar/remote_services.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ),
      home: const MyHomePage(title: 'Flutter Demo Home Page'),
    );
  }
}

class MyHomePage extends StatefulWidget {
  const MyHomePage({super.key, required this.title});

  final String title;

  @override
  State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  List<User>? users;
  var isLoaded = false;

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

  getData() async {
    users = await remoteservice().getUsers();
    if (users != null) {
      setState(() {
        isLoaded = true;
      });
```

```
  }
 }

 @override
 Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
     backgroundColor: Theme.of(context).colorScheme.inversePrimary,
     title: Text(widget.title),
    ),
    body: Visibility(
     visible: isLoaded,
     child: ListView.builder(
      itemCount: users?.length,
      itemBuilder: (context, index) {
       return Container(
         padding: const EdgeInsets.all(8.0),
         child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text(users![index].name),
            Text(users![index].emailid),
          ],
         ),
       );
      },
     ),
     replacement: const Center(
      child: CircularProgressIndicator(),
     ),
    ),
   );
  }
}
```

user.dart:

```
import 'dart:convert';

List<User> userFromJson(String str) => List<User>.from(json.decode(str).map((x) =>
User.fromJson(x)));

String userToJson(List<User> data) => json.encode(List<dynamic>.from(data.map((x) =>
x.toJson())));

class User {
 String name;
 String emailid;

 User({
  required this.name,
  required this.emailid,
 });

 factory User.fromJson(Map<String, dynamic> json) => User(
  name: json["name"],
```

```
    emailid: json["emailid"],
  );

  Map<String, dynamic> toJson() => {
    "name": name,
    "emailid": emailid,
  };
}
```

Remote_service.dart:

```
import 'package:http/http.dart' as http;
import 'package:restkedar/user.dart';

class remoteservice{
  Future<List<User>?> getUsers() async
  {
   var client = http.Client();
   var uri = Uri.parse("https://mocki.io/v1/e4adeb48-dcdb-4287-a72b-c15f52a98e90");
   var response = await client.get(uri);
   if (response.statusCode == 200)
   { var data = response.body;
     return userFromJson(data);


   }

  }
}
```

OUTPUT:

JSON GENERATOR

Generate | Reset

Help Ukraine!

```
{
  "records": [
    {
      "name": "Sandeep Yadav",
      "emailid": "sandyyadav578@gmail.com"
    },
    {
      "name": "John Doe",
      "emailid": "john.doe@example.com"
    },
    {
      "name": "Jane Smith",
      "emailid": "jane.smith@example.com"
    },
    {
      "name": "Robert Johnson",
      "emailid": "robert.johnson@example.com"
    },
    {
      "name": "Emily Davis",
      "emailid": "emily.davis@example.com"
    },
    {
      "name": "Michael Brown",
      "emailid": "michael.brown@example.com"
    }
  ]
}
```

```
{
  "records": [
    {
      "name": "Sandeep Yadav",
      "emailid": "xandyyadav578@gmail.com"
    },
    {
      "name": "John Doe",
      "emailid": "john.doe@example.com"
    },
    {
      "name": "Jane Smith",
      "emailid": "jane.smith@example.com"
    },
    {
      "name": "Robert Johnson",
      "emailid": "robert.johnson@example.com"
    },
    {
      "name": "Emily Davis",
      "emailid": "emily.davis@example.com"
    },
    {
      "name": "Michael Brown",
      "emailid": "michael.brown@example.com"
    }
  ]
}
```

```
{
  "records": [
    {
      "name": "Sandeep Yadav",
      "emailid":
```

Success! Your API is available at
https://mocki.io/v1/e4219579-382b-489b-bf7d-1c0a5e894475

Create API

quicktype

Sponsor us | Share on Twitter | Open on GitHub | Options

Name: User | Source type: JSON

Language: Other — Dart

- Null Safety
- Types only
- Put encoder & decoder in Class
- Make all properties required
- Make all properties final
- Generate CopyWith method
- Make all properties optional

Copy Code

```
import 'dart:convert';

User userFromJson(String str) => User.fromJson(json.decode(str));

String userToJson(User data) => json.encode(data.toJson());

class User {
    List<Record> records;

    User({
        required this.records,
    });

    factory User.fromJson(Map<String, dynamic> json) => User(
        records: List<Record>.from(json["records"].map((x) => Record.fromJson(x))),
    );

    Map<String, dynamic> toJson() => {
        "records": List<dynamic>.from(records.map((x) => x.toJson())),
    };
}

class Record {
    String name;
    String emailid;

    Record({
        required this.name,
        required this.emailid,
    });

    factory Record.fromJson(Map<String, dynamic> json) => Record(
        name: json["name"],
        emailid: json["emailid"],
    );

    Map<String, dynamic> toJson() => {
        "name": name,
        "emailid": emailid,
    };
}
```
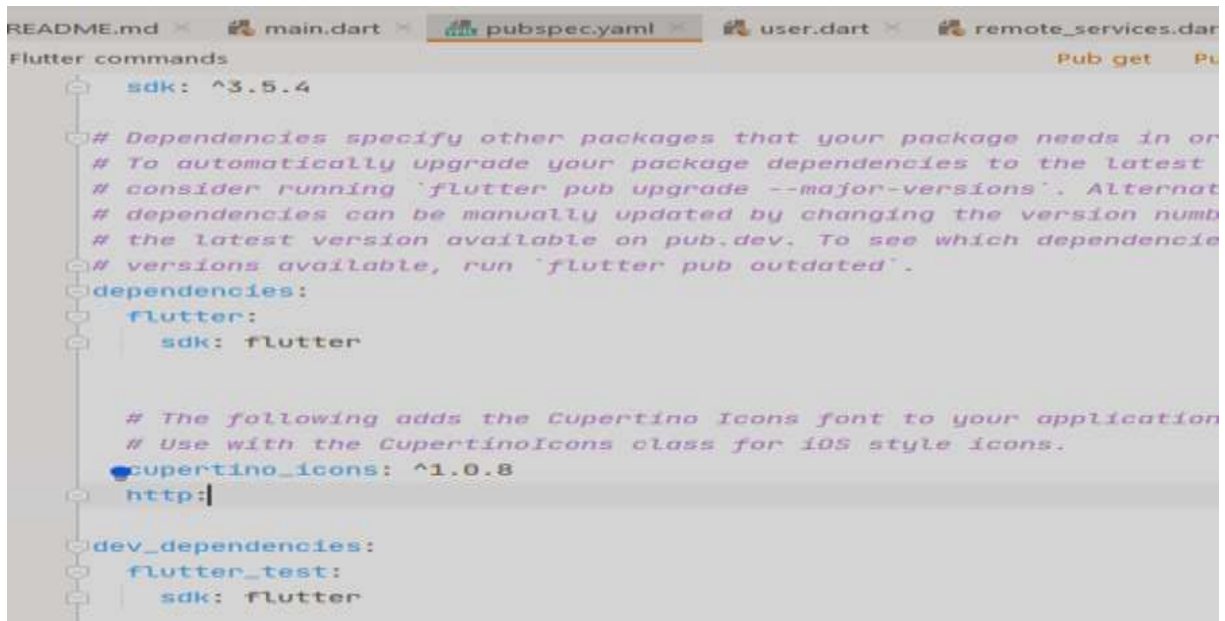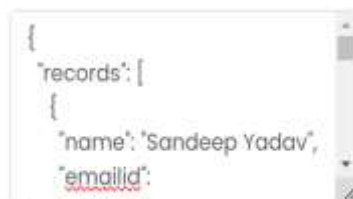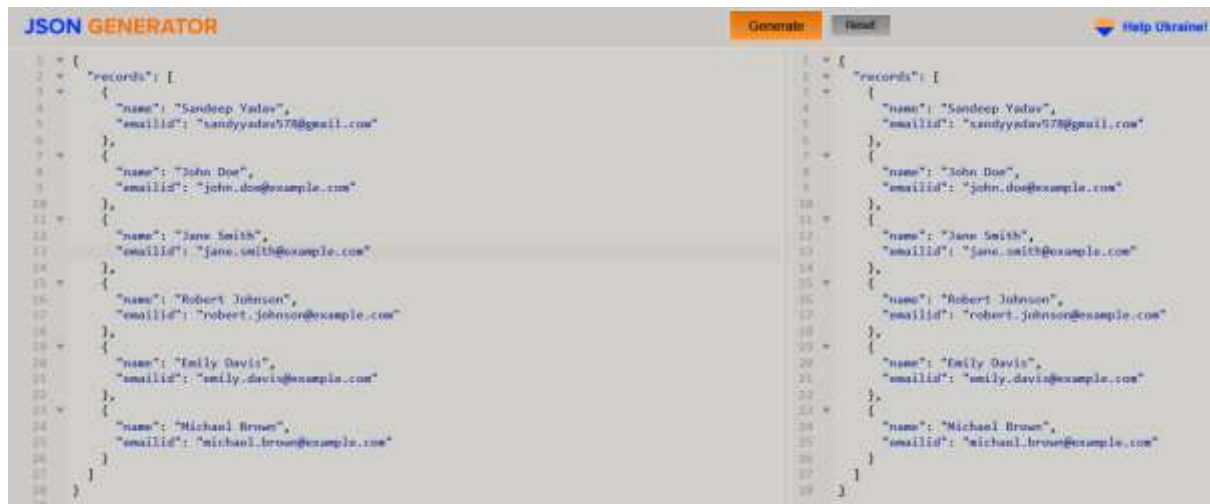
Flutter Demo Home Page

Name: Sandeep Yadav
Email ID: sandyyadav578@gmail.com

Name: Jane Smith
Email ID: jane.smith@example.com

Name: Jane Smith
Email ID: jane.smith@example.com

Name: Robert Johnson
Email ID: robert.johnson@example.com

Name: Emily Davis
Email ID: emily.davis@example.com

Name: Michael Brown
Email ID: michael.brown@example.com