

### Currency Converter App

Aplikasi ini bertujuan untuk memungkinkan pengguna untuk mengonversi nilai mata uang secara real-time menggunakan Frankfurter API.

#### Fitur Utama:

- Mengambil data nilai tukar mata uang secara asynchronous
- Mengonversi nilai tukar secara otomatis berdasarkan input pengguna

#### API yang Digunakan

Aplikasi ini menggunakan **Frankfurter API** untuk mengambil data nilai tukar mata uang.

- **Endpoint API:** <https://api.frankfurter.app/latest>

**Untuk menggunakan kode dalam konteks pemrograman Dart, berikut adalah langkah-langkah yang bisa diikuti:**

1. Mempersiapkan Lingkungan Pengembangan
  - Instal Dart SDK jika belum terpasang.
  - Gunakan Visual Studio Code (VS Code) dengan ekstensi Dart.
  - Atau dapat menggunakan <https://dartpad.dev/>
2. Tulis Kode Dart
  - Buat file dengan ekstensi .dart, misalnya main.dart.
  - Copy code yang ada
3. Jalankan kode Dart dengan klik Run

**Dokumentasi Code:**

The screenshot shows the DartPad interface with the following code:

```
1 import 'dart:convert';
2 import 'package:flutter/material.dart';
3 import 'package:http/http.dart' as http;
4
5 /// URL API tanpa API Key & bebas CORS
6 const String apiUrl = "https://api.frankfurter.app/latest";
7
8 void main() {
9   runApp(const CurrencyConverterApp());
10 }
11
12 /**Aplikasi utama dengan tema modern**
13 class CurrencyConverterApp extends StatelessWidget {
14   const CurrencyConverterApp({super.key});
15
16   @override
17   Widget build(BuildContext context) {
18     return MaterialApp(
19       debugShowCheckedModeBanner: false,
20       theme: ThemeData(
21         primarySwatch: Colors.indigo,
22         scaffoldBackgroundColor: Colors.grey[100],
23       ),
24       home: const CurrencyConverterScreen(),
25     );
26   }
27 }
28
29 /**Layar utama dengan state management & UI lebih menarik**
30 class CurrencyConverterScreen extends StatefulWidget {
31   const CurrencyConverterScreen({super.key});
32
33   @override
34   _CurrencyConverterScreenState createState() =>
35   _CurrencyConverterScreenState();
36 }
37
38 class _CurrencyConverterScreenState extends State<CurrencyConverterScreen> {
39   Map<String, double> exchangeRates = {};
40   String fromCurrency = "USD";
41   String toCurrency = "IDR";
42   double amount = 1.0;
43   String result = "Masukkan jumlah & pilih mata uang.";
44   bool isLoading = true;
45   String errorMessage = "";
46
47   @override
48   void initState() {
49     super.initState();
50     _fetchExchangeRates();
51   }
52 }
```

The code defines a main application structure with a Material App and a specific screen for currency conversion. It uses the `http` package to fetch data from an API and the `flutter` package for the UI.



DartPad

New Samples

```
51 }
52
53 /// **Mengambil data nilai tukar dari API dengan Error Handling**
54 Future<void> _fetchExchangeRates() async {
55   setState(() {
56     isLoading = true;
57     errorMessage = "";
58   });
59
60   try {
61     final response = await http
62       .get(Uri.parse(apiUrl))
63       .timeout(const Duration(seconds: 10));
64
65     if (response.statusCode == 200) {
66       final data = json.decode(response.body);
67       setState(() {
68         exchangeRates = Map<String, double>.from(data["rates"]);
69         isLoading = false;
70       });
71     } else {
72       throw Exception(
73         "Gagal mengambil data dari API (Status: ${response.statusCode})",
74       );
75     }
76   } catch (error) {
77     setState(() {
78       errorMessage = "⚠️ Gagal mengambil data: $error";
79       isLoading = false;
80     });
81   }
82 }
83
84 /// **Melakukan konversi mata uang dengan validasi**
85 void _convertCurrency() {
86   if (exchangeRates.isEmpty) {
87     setState(() {
88       result = "⚠️ Data nilai tukar belum tersedia.";
89     });
90     return;
91   }
92
93   if (!exchangeRates.containsKey(fromCurrency) ||
94       !exchangeRates.containsKey(toCurrency)) {
95     setState(() {
96       result = "⚠️ Mata uang tidak valid.";
97     });
98     return;
99   }
100  double fromRate = exchangeRates[fromCurrency]!;
101  double toRate = exchangeRates[toCurrency]!;
```

Privacy notice Feedback



DartPad

New Samples

```
107     " $amount $fromCurrency = ${convertedAmount.toStringAsFixed(2)} $toCurrency";
108   });
109 }
110
111 /// **Membuat Dropdown untuk memilih mata uang dengan desain lebih modern**
112 Widget _currencyDropdown(
113   String selectedCurrency,
114   ValueChanged<String?> onChanged,
115 ) {
116   return Container(
117     padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 5),
118     decoration: BoxDecoration(
119       color: Colors.white,
120       borderRadius: BorderRadius.circular(10),
121       border: Border.all(color: Colors.indigo, width: 1),
122     ),
123     child: DropdownButton<String>(
124       value: selectedCurrency,
125       isExpanded: true,
126       underline: const SizedBox(),
127       items:
128         exchangeRates.keys.map((currency) {
129           return DropdownMenuItem(value: currency, child: Text(currency));
130         }).toList(),
131       onChanged: onChanged,
132     ),
133   );
134 }
135
136 @override
137 Widget build(BuildContext context) {
138   return Scaffold(
139     appBar: AppBar(
140       title: const Text(
141         "Currency Converter",
142         style: TextStyle(fontSize: 22, fontWeight: FontWeight.bold),
143       ),
144     ),
145     body: Padding(
146       padding: const EdgeInsets.all(20),
147       child: Column(
148         mainAxisAlignment: MainAxisAlignment.center,
149         children: [
150           if (isLoading) const CircularProgressIndicator(),
151           if (errorMessage.isNotEmpty)
152             Padding(
153               padding: const EdgeInsets.all(10),
154               child: Text(
155                 errorMessage,
156                 style: const TextStyle(
157                   color: Colors.red,
158                   fontWeight: FontWeight.bold,
```

Privacy notice  Feedback 

Tampilan Aplikasi (UI Preview)

## Currency Converter

Masukkan jumlah:

20000

IDR



USD



Konversi

20000 IDR = 1.22 USD