

通貨換算アプリケーション 簡易操作マニュアル

1. アプリケーションの概要

本アプリケーションは、日本円 (JPY)、韓国ウォン (KRW)、シンガポールドル (SGD) 間の通貨換算を行うための Web アプリケーションです。外部 API を利用して最新の為替レートを取得し、迅速かつ正確な換算結果を提供します。

2. 実行環境の準備と起動手順

本アプリケーションを実行するには、API キーの設定と、必要な Python ライブラリのインストールが必要です。

2.1. 必要なライブラリのインストール

アプリケーションの実行に必要なライブラリをインストールします。

`pip install -r requirements.txt`

```
[(.venv)] PS C:\Users\... \Documents\TravelRateConverter-main> pip install -r requirements.txt
Using cached python_dotenv-1.2.1-py3-none-any.whl (21 kB)
Using cached flask-3.1.2-py3-none-any.whl (103 kB)
Using cached blinker-1.9.0-py3-none-any.whl (8.5 kB)
Using cached certifi-2025.10.5-py3-none-any.whl (163 kB)
Using cached charset_normalizer-3.4.4-cp311-cp311-win_amd64.whl (106 kB)
Using cached click-8.3.0-py3-none-any.whl (109 kB)
Using cached idna-3.11-py3-none-any.whl (71 kB)
Using cached itsdangerous-2.2.0-py3-none-any.whl (16 kB)
Using cached jinja2-3.1.6-py3-none-any.whl (134 kB)
Using cached markupsafe-3.0.3-cp311-cp311-win_amd64.whl (15 kB)
Using cached urllib3-2.5.0-py3-none-any.whl (129 kB)
Using cached werkzeug-3.1.3-py3-none-any.whl (224 kB)
Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Installing collected packages: urllib3, python-dotenv, markupsafe, itsdangerous, idna, colorama, charset_normalizer, certifi, blinker, werkzeug, requests, jinja2, click, Flask
Successfully installed Flask-3.1.2 blinker-1.9.0 certifi-2025.10.5 charset_normalizer-3.4.4 click-8.3.0 colorama-0.4.6 idna-3.11 itsdangerous-2.2.0 jinja2-3.1.6 markupsafe-3.0.3 python-dotenv-1.2.1 requests-2.32.5 urllib3-2.5.0 werkzeug-3.1.3
[notice] A new release of pip is available: 24.0 -> 25.3
[notice] To update, run: python.exe -m pip install --upgrade pip
[(.venv)] PS C:\Users\... \Documents\TravelRateConverter-main>
```

2.2. API キーの取得

本アプリケーションは、為替レートの取得に [ExchangeRate-API](#) を利用しています。利用には API キーが必要です。アカウントを登録し、API キーを取得してください。

2.2.1. サイトにアクセス

上記のリンクをクリックしサイトにアクセスし、メールアドレスを入力します。

The Accurate & Reliable Exchange Rate API

🌐 Currency conversion rates for 161 currencies.

👤 Over 15 years of exceptional uptime & support.

✅ Perfect for SaaS, Dashboards & E-Commerce.

Hundreds of thousands of developers relax while our exchange rate API delivers forex rates, day-in, day-out!

[Get Free Key!](#)

2.2.2. アカウント作成

アカウント作成画面が表示されるので、メールアドレスとパスワードを入力して作成します。

Create Account

You're signing up to our free plan!


Our Pro Plan has a free 2 week trial.
No credit card required, just click here to try!

Email

☐ Accept Marketing Emails

We send one discount offer a year. Address never shared.

Password

☐ 私がボットではないことを確認します。  クラウドフラール
プライバシー・と契約条件

Verification still required....

2.2.3. メールの確認

入力したメールアドレス宛にメールが送られてくるので開きます。
そこにダッシュボードのリンクが添付されているので、それをクリックすれば API キーが取得できます。

Welcome to ExchangeRate-API, thanks for signing up.

Please click this link to activate your account: <https://app.exchangerate-api.com/activate/>

Your new API key is already being preloaded into our cache system for a fast start once activated.

2.2.4. API キーの設定

① .env ファイル方式

この方法は、API キーをリポジトリから隔離するための一般的な方法です。

- リポジトリ内の .env.template ファイルをコピーし、ファイル名を .env に変更します。
 - copy .env.template .env (コマンドプロンプト・PowerShell どちらでも)
- .env ファイルを開き、以下の形式で取得した API キーを記載します。

EXCHANGERATE_API_KEY=ここに取得した API キーを貼り付け

```
.env X
.env
1 # ExchangeRate-APIのAPIキーを設定してください
2 # https://www.exchangerate-api.com/
3 EXCHANGERATE_API_KEY=akdfjaeur3wnf@49hnfeh@i4hfn@a4
```

② 環境変数を直接設定する方式

.env ファイル方式を利用せず、システムの環境変数として直接設定することも可能です。

- Windows (コマンドプロンプト):
 - set EXCHANGERATE_API_KEY=YOUR_API_KEY
- Windows (PowerShell)
 - \$env:EXCHANGERATE_API_KEY="YOUR_API_KEY"
- macOS / Linux:
 - export EXCHANGERATE_API_KEY="YOUR_API_KEY"

2.3. アプリケーションの起動

API キーの設定が完了した後、アプリケーションをローカルサーバーで起動します。

flask --app main run

```
(.venv) PS C:\Users\... \Documents\TravelRateConverter-main> flask --app main run
* Serving Flask app 'main'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

2.4. ブラウザでのアクセス

サーバーが起動したら、Web ブラウザを開き、以下の URL にアクセスしてください。

http://127.0.0.1:5000

3. 通貨換算の操作手順

ブラウザでアプリケーションにアクセスすると、換算を行うための GUI 画面が表示されます。

The screenshot shows a web application titled "Travel Rate Converter". It features a form with the following elements: a text input field labeled "金額:" (Amount); a dropdown menu labeled "換算元:" (Convert from) with "日本円 (JPY)" selected; another dropdown menu labeled "換算先:" (Convert to) with "日本円 (JPY)" selected; a blue button labeled "換算" (Convert); and a light gray rectangular area at the bottom, likely for the result.

3.1. 換算する金額の入力

1. 画面上部の金額入力欄に、換算したい数値を入力します。



The screenshot shows the 'Travel Rate Converter' form. The '金額' (Amount) input field contains the value '1000' and is highlighted with a red rectangular box. Below it, the '換算元' (From) dropdown is set to '日本円 (JPY)' and the '換算先' (To) dropdown is also set to '日本円 (JPY)'. A blue '換算' (Convert) button is at the bottom.

3.2. 換算元の通貨を選択

2. 換算元の通貨（例：JPY）をドロップダウンリストから選択します。



The screenshot shows the 'Travel Rate Converter' form. The '換算元' (From) dropdown menu is open, showing '日本円 (JPY)' as the selected option, and is highlighted with a red rectangular box. The '金額' field is '1000' and the '換算先' dropdown is also set to '日本円 (JPY)'. A blue '換算' button is at the bottom.

3.3. 換算先の通貨を選択

3. 換算先の通貨（例：KRW）をドロップダウンリストから選択します。



The screenshot shows the 'Travel Rate Converter' form. The '換算先' (To) dropdown menu is open, showing '韓国ウォン (KRW)' as the selected option, and is highlighted with a red rectangular box. The '金額' field is '1000' and the '換算元' dropdown is set to '日本円 (JPY)'. A blue '換算' button is at the bottom.

3.4. 換算の実行

4. 入力と選択が完了したら、「換算」ボタンを押して計算を実行します。

3.5. 換算結果の確認

5. 画面下部に換算結果が表示されます。

Travel Rate Converter

金額:

換算元:

日本円 (JPY) ▼

換算先:

日本円 (JPY) ▼

換算

1,000.0 JPY = 9,379.20 KRW