

PROJECT ON : CURRENCY CONVERTER

Introduction

The Currency Converter project is a simple web application developed using **Python's Flask** web framework. It allows users to convert between various currencies in real-time, using an external currency exchange rate API. The primary objective of this project is to provide a practical example of building a web application using Flask and integrating an external API to fetch live data.

This web-based currency converter is designed to take an amount in one currency and convert it into another currency by utilizing the latest exchange rates.

Features

- **User-friendly Web Interface:** Developed with Flask, this web app allows users to input the amount and select the currencies to convert.
 - **Real-time Exchange Rates:** The app fetches live exchange rates using an external API (like **ExchangeRate-API** or any similar service).
 - **Support for Multiple Currencies:** The application supports multiple currencies, such as USD, EUR, GBP, INR, etc.
 - **Error Handling:** Displays appropriate error messages if the API fails to fetch data or if the conversion process encounters an issue.
 - **Responsive Design:** The web interface is simple and responsive, adjusting well to different screen sizes.
-

Main Components

- **Flask Application:** The main framework that handles HTTP requests and routes.
- **HTML Form:** Provides an interface where the user can input the amount, select the source and target currencies.
- **External API:** Used to fetch real-time exchange rates (e.g., **ExchangeRate-API**).
- **JavaScript (Optional):** For handling certain frontend tasks, like form validation or dynamic updates.
- **Error Handling:** If the API call fails or an invalid response is received, the application displays a user-friendly error message.

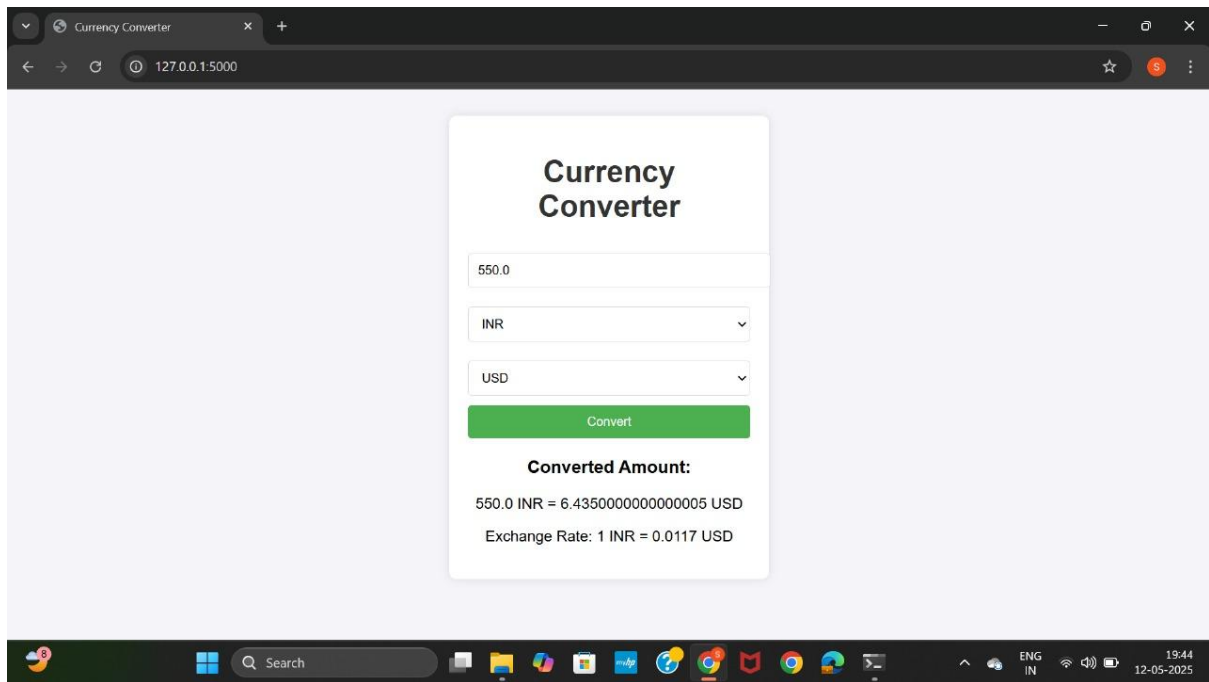
Output and Usage

Output:

When the user accesses the application, a form is displayed where they can:

- Enter the amount they want to convert.
- Select the source currency.
- Select the target currency.
- Press "Convert" to get the result.

If the conversion is successful, the converted amount and the exchange rate are shown. If there is any error (e.g., API failure), an error message is displayed instead.



Usage

1. Run the application by executing `python app.py` in the terminal.
 2. Open a web browser and go to `http://127.0.0.1:5000/`.
 3. Enter the amount and select the currencies to convert.
 4. Press the "Convert" button.
 5. View the converted amount and exchange rate on the page.
-

Conclusion

This Currency Converter project is an excellent example of how to integrate external APIs into a Flask web application. It demonstrates key Flask concepts such as routing, templates, handling form data, and integrating third-party services. This project can be extended to support more advanced features like converting to multiple currencies at once, saving conversion history, and even supporting other APIs for additional functionality.
