Simple Python Currency Converter

📝 Overview

This is a straightforward Python program designed to convert money from one currency to another (like converting Rupees to Dollars).

Key things to know:

- It's a simple command-line tool—you run it directly from your terminal.

- The exchange rates are fixed (hardcoded) inside the program.

- The **US Dollar (USD)** is the main reference currency, meaning all calculations are done relative to the USD.

⚙️ How to Set Up and Run

What You Need (Prerequisites)

Just one thing: **Python 3** installed on your computer.

How to Execute

1. **Save the file:** Take the provided code (the `Currency_Converter.py` file) and save it on your machine.

2. **Open your terminal:** Go to your command prompt (Terminal, PowerShell, or Command Prompt) and navigate to where you saved the file.

3. **Run the script:** Type the following command and hit Enter:

   ```
   python Currency_Converter.py
   ```

4. **Answer the questions:** The program will ask you for three pieces of information:

   - The **amount** (the number you want to convert, e.g., 500).

   - The **FROM** currency code (e.g., `INR` for Indian Rupee, `USD` for US Dollar).

   - The **TO** currency code (e.g., `EUR` for Euro).

   Pro Tip: If you accidentally type letters when it asks for the amount, it will keep asking until you enter a valid number.

🖥️ Behind the Scenes (Code Details)

The `exchange_rates` Dictionary

This is essentially a lookup table that holds all the exchange values.

- **Key**: The three-letter code for the currency (like `GBP` or `JPY`).

- **Value**: How much of that currency equals **1 US Dollar (USD)**.

- **Coverage**: This table includes many currencies from around the world (Asia, Europe, Americas, Africa).

The `convert_currency` Function

This function (the `convert_currency(amount, from_currency, to_currency)` part) is where the actual math happens.

1. **Safety Check (Validation)**: It first checks to make sure the currency codes you entered (`FROM` and `TO`) actually exist in the `exchange_rates` list. If they don't, it stops and prints an error.

2. **Step 1: Convert to USD**: It takes your starting amount and converts it into its equivalent value in USD. This is the crucial middle step, using the formula:

$$\text{Amount in USD} = \frac{\text{Starting Amount}}{\text{Rate of FROM Currency}}$$

3. **Step 2: Convert to Target Currency**: Finally, it takes the USD value and converts it into the currency you want (the `TO` currency) using the formula:

$$\text{Final Converted Amount} = \text{Amount in USD} \times \text{Rate of TO Currency}$$

4. **Result**: The final converted amount is displayed right there in your terminal.

Input Handling

The program automatically converts the currency codes you type to **CAPITAL LETTERS** (uppercase) so they always match the keys in the `exchange_rates` dictionary.