Project : Real-Time Forex Rate Monitoring and Analysis

Instructions for Setting Up and Running the Exchange Rates Script

These instructions guide you through setting up, configuring, and automating the execution of a Python script designed to fetch and update exchange rates from an API into a MySQL database. By following these steps, you will ensure the script runs automatically at your desired intervals using Windows Task Scheduler.

Requirements

1. **Python Installation**:
   Ensure Python 3.x is installed on your system. You can download it from [Python's official website](#).
2. **MySQL Database**:
   MySQL must be installed and running on your system. Ensure the database schema is set up according to the script's requirements.
3. **Python Libraries**:
   Install the necessary Python libraries using pip. Open your command prompt and run:

```bash
Copy code
pip install mysql-connector-python requests pandas
```

Python Script Setup

1. **Script Path**:
   Save the Python script as `update_exchange_rates.py` in your desired directory (e.g., `C:\Users\allan\Videos\`).
2. **Configuration**:
   o **MySQL Connection**: Update the `db_params` dictionary in the script with your MySQL connection details:

```python
Copy code
db_params = {
    'host': 'localhost',
    'user': 'your_username',
    'password': 'your_password',
    'database': 'your_database'
}
```

   o **API Key**: Insert your Alpha Vantage API key by updating the `api_key` variable:

```python
Copy code
api_key = 'YOUR_ALPHA_VANTAGE_API_KEY'
```

- o **CSV File Path**: Ensure the path to the `currency_pairs.csv` file is correctly set:

```python
Copy code
currency_pairs_file = r'C:\Path\To\currency_pairs.csv'
```

Batch File Creation

To automate the script execution, create a batch file that runs the Python script. This file will be scheduled to run at your desired intervals.

Example Snippet for Minutely Execution:

```batch
Copy code
@echo off
REM Set the path to the Python interpreter and script
set PYTHON_PATH=C:\Path\To\Python\python.exe
set SCRIPT_PATH=C:\Path\To\Your\Directory\update_exchange_rates.py

REM Run the Python script
"%PYTHON_PATH%" "%SCRIPT_PATH%"
```

Steps to Update the Batch File:

1. **Open the Batch File**:
   - o Right-click the `run_update_exchange_rates_hourly.bat` file.
   - o Select "Open with" and choose Notepad (or another text editor).
2. **Update the Batch File**:
   - o Replace `C:\Path\To\Python\python.exe` with the path to where Python is installed on your system. You can find this path by:
     - ▪ Opening a command prompt (`cmd`).
     - ▪ Typing `where python` and pressing Enter.
   - o Replace `C:\Path\To\Your\Directory\update_exchange_rates.py` with the path to your Python script.
   - o Set log file path (included in the folder with name: "update_exchange_rates_results")

   Example:

```batch
Copy code
REM Set the path to the Python interpreter and script
```

```
set
PYTHON_PATH=C:\Users\YourUsername\AppData\Local\Programs\Python\Pytho
n39\python.exe
set
SCRIPT_PATH=C:\Users\YourUsername\Scripts\update_exchange_rates.py
```

3. **Save the Changes**:
    o Click "File" in Notepad.
    o Select "Save" to save the updated batch file.

Snippet of my bat file

```
@echo off
REM Set the path to the Python interpreter and script
set PYTHON_PATH=C:\Users\allan\AppData\Local\Microsoft\WindowsApps\python.exe
set SCRIPT_PATH=C:\Users\allan\Videos\360T\update_exchange_rates.py


REM Set the path to the log file
set LOG_FILE=C:\Users\allan\Videos\360T\update_exchange_rates_results.log

REM Run the Python script and redirect output and errors to the log file
"%PYTHON_PATH%" "%SCRIPT_PATH%" > "%LOG_FILE%" 2>&1

REM Display the contents of the log file in the Command Prompt window
type "%LOG_FILE%"

REM Pause to keep the Command Prompt window open for review
pause
```

Running the Batch File:

1. **Execute the Batch File**:
    o Double-click the `run_update_exchange_rates_hourly.bat` file to run it.
    o This will execute your Python script and display the results in a command prompt window.

Automating with Windows Task Scheduler

To ensure the script runs automatically, set up a scheduled task in Windows Task Scheduler.

Steps to Create a Scheduled Task:

1. **Open Windows Task Scheduler**:
    - Press `Windows Key + S` to open the search bar.
    - Type "Task Scheduler" and press Enter to open the Task Scheduler application.
2. **Create a New Task**:
    - In the Task Scheduler window, click on "Create Basic Task" on the right-hand side under the Actions pane.
3. **Name Your Task**:
    - Enter a name for the task (e.g., "Update Exchange Rates Hourly").
    - Optionally, add a description (e.g., "Runs a Python script every hour to update exchange rates").
    - Click "Next".
4. **Set the Trigger**:
    - Select "Daily" to run the task every day.
    - Click "Next".
5. **Specify the Start Time**:
    - Set the start date and time. This will be the first time the task runs.
    - Set "Recur every" to 1 day to run the task every day.
    - Click "Next".
6. **Configure the Repetition**:
    - On the next screen, select the "Repeat task every" option.
    - Choose "1 hour" from the drop-down menu.
    - Set "for a duration of" to 1 day to repeat the task every hour within a 24-hour period.
    - Click "Next".
7. **Define the Action**:
    - Select "Start a program" and click "Next".
8. **Specify the Program/Script**:
    - In the "Program/script" field, click "Browse" and select the batch file (`run_update_exchange_rates_hourly.bat`).
    - Click "Next".
9. **Review and Finish**:
    - Review the summary of your task settings.
    - If everything looks correct, click "Finish" to create the task.
10. **Verify Task Settings**:

- In the Task Scheduler window, find your task under the Task Scheduler Library.
- Right-click the task and select "Properties" to review or modify settings.

Note: Repeat the same steps to create a task for minutely execution if needed.

Output Expectations

- **Hourly Job**: This job will run the Python script every hour, updating the exchange rates in the database.
- **Minutely Job**: If configured, this job will run every minute, useful for high-frequency updates involving many currency pairs.

By following these instructions, you can ensure your exchange rate updates are handled automatically, reliably, and with minimal manual intervention.

Output: TASK A

```
Currency Pair    Current Rate    % Change
=================================================
USDAUD           1.470960        0.04%
USDCAD           1.345100        0.06%
USDEUR           0.896100        0.20%
USDGBP           0.755080        0.11%
USDJPY           144.442000      0.21%
Results displayed successfully.
```

Output: TASK B

```
Currency Pair    Current Rate    % Change
==========================================
AUD/CAD          0.914510         0.17%
AUD/CHF          0.573050        -0.08%
AUD/CNY          4.838000         0.34%
AUD/EUR          0.609720         0.52%
AUD/GBP          0.513650         0.06%
AUD/HKD          5.296000         0.33%
AUD/JPY          98.059000        0.20%
AUD/NZD          1.088250        -0.29%
AUD/SGD          0.885500         0.36%
AUD/USD          0.679400         0.31%
CAD/AUD          1.092900        -0.20%
CAD/CHF          0.626610        -0.26%
CAD/CNY          5.290500         0.20%
CAD/EUR          0.666500         0.36%
CAD/GBP          0.561200        -0.18%
CAD/HKD          5.793310         0.19%
CAD/JPY          107.223000       0.03%
CAD/NZD          1.188700        -0.54%
CAD/SGD          0.967900         0.10%
CAD/USD          0.742300         0.08%
CHF/AUD          1.743790         0.06%
CHF/CAD          1.594900         0.23%
CHF/CNY          8.442300         0.47%
CHF/EUR          1.063400         0.58%
CHF/GBP          0.895900         0.13%
CHF/HKD          9.245600         0.46%
CHF/JPY          171.107000       0.31%
CHF/NZD          1.898400        -0.20%
```

Note: Please find included video file for a demonstration