IBM Developer
SKILLS NETWORK

Peer Review Assignment - Data Engineer - Extract API Data

Estimated time needed: **20** minutes

Objectives

In this part you will:

- Collect exchange rate data using an API
- Store the data as a CSV

For this lab, we are going to be using Python and several Python libraries. Some of these libraries might be installed in your lab environment or in SN Labs. Others may need to be installed by you. The cells below will install these libraries when executed.

[1]:
```
!pip install pandas
!pip install requests
```

```
Requirement already satisfied: pandas in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (1.1.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from pandas) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from pandas) (2021.1)
Requirement already satisfied: numpy>=1.15.4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from pandas) (1.19.5)
Requirement already satisfied: six>=1.5 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from python-dateutil>=2.7.3->pandas) (1.15.0)
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (2.25.1)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (1.26.6)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (2021.5.30)
Requirement already satisfied: chardet<5,>=3.0.2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (4.0.0)
```

Imports

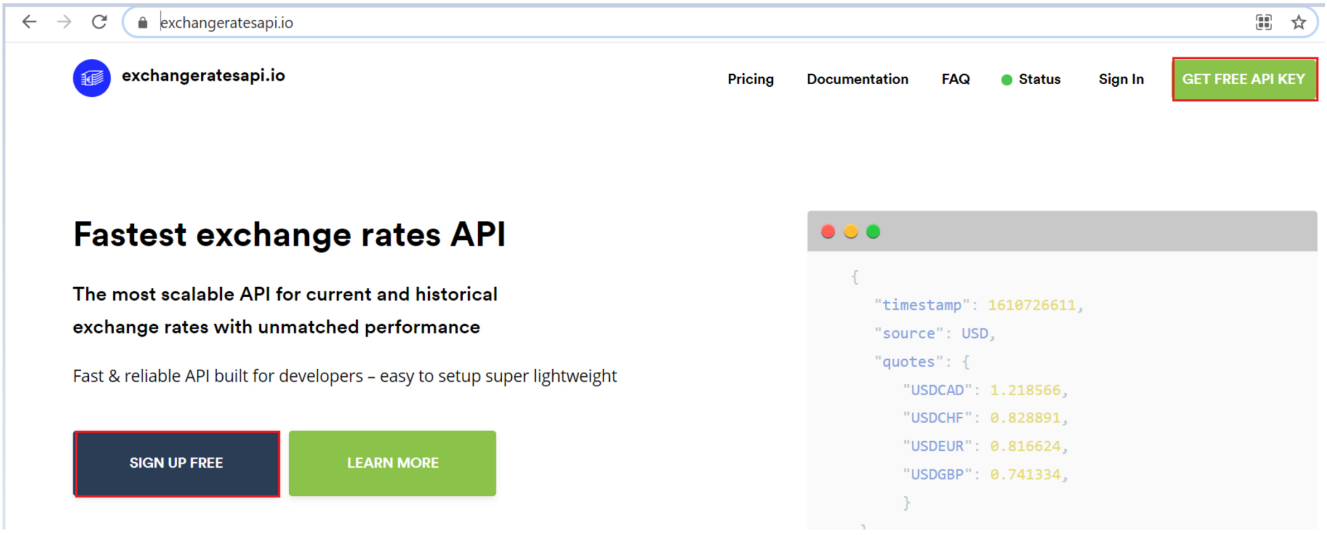
Import any additional libraries you may need here.

[2]:
```
import requests
import pandas as pd
```

Extract Data Using an API

Using ExchangeRate-API we will extract currency exchange rate data. Use the below steps to get the access key and to get the data.

1. Open the url : https://api.exchangeratesapi.io/ and create a free account.
2. Once the account is created. You will get the Get the Free API key option on the top as shown below:



3. Copy the API key and use in the url in Question 1.

Call the API

Question 1 Using the `requests` library call the endpoint given above and save the text, remember the first few characters of the output:

[5]:
```
# Write your code here
#url = "http://api.exchangeratesapi.io/v1/latest?base=EUR&access_key=97b6372f9ef89932b61f5625dcf4040d"  #Make sure to change ******* to your API key.
r = requests.get("http://api.exchangeratesapi.io/v1/latest?base=EUR&access_key=97b6372f9ef89932b61f5625dcf4040d")
r
```

[5]:
```
<Response [200]>
```

Save as DataFrame

Question 2 Using the data gathered turn it into a `pandas` dataframe. The dataframe should have the Currency as the index and `Rate` as their columns. Make sure to drop unnecessary columns.

Did you know? IBM Watson Studio lets you build and deploy an AI solution, using the best of open source and IBM software and giving your team a single environment to work in. Learn more here.

[9]:
```
# Turn the data into a dataframe
r = requests.get("http://api.exchangeratesapi.io/v1/latest?base=EUR&access_key=97b6372f9ef89932b61f5625dcf4040d")

j = r.json()

df = pd.DataFrame.from_dict(j)
```

[10]:
```
df.index.name = 'Currency'
df
```

[10]:

Currency	success	timestamp	base	date	rates
AED	True	1629067925	EUR	2021-08-15	4.332243
AFN	True	1629067925	EUR	2021-08-15	95.346491
ALL	True	1629067925	EUR	2021-08-15	121.795380
AMD	True	1629067925	EUR	2021-08-15	580.615629
ANG	True	1629067925	EUR	2021-08-15	2.116815
...	...	...	...	...	...
YER	True	1629067925	EUR	2021-08-15	294.872611
ZAR	True	1629067925	EUR	2021-08-15	17.364922
ZMK	True	1629067925	EUR	2021-08-15	10616.825731
ZMW	True	1629067925	EUR	2021-08-15	22.766221
ZWL	True	1629067925	EUR	2021-08-15	379.795297

168 rows × 5 columns

[11]:
```
# Drop unnecessary columns
df=df.drop(['base', 'date'], axis=1)
```

Load the Data

Using the dataframe save it as a CSV names `exchange_rates_1.csv`.

[12]:
```
# Save the Dataframe
df.to_csv(r'exchange_rates_1.csv', index = False)
```

Your CSV should be in this format with more currencies

	Rates
AED	4.398618
AFN	92.917693
ALL	123.099093
AMD	621.935674
ANG	2.149648

Authors

Ramesh Sannareddy, Joseph Santarcangelo and Azim Hirjani

Other Contributors

Rav Ahuja

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2021-04-15	0.2	Malika	Updated the lab from USD to EUR
2020-11-25	0.1	Ramesh Sannareddy	Created initial version of the lab


Simple
0
2
Python | Idle
Fully initialized
Mem: 443.29 / 6144.00 MB
Saving completed
Mode: Command
Ln 1, Col 1
English (American)
API_Engineer_Peer_Review_Assignment_Extract.ipynb