



POWER BI – CALL REST API

SQLBITS 2023

Iuliana Tuhasu



✓ Senior Data Engineer



✓ 8+ years of experience in the industry

✓ Founder at Softentity

✓ Passionate about Data 😊

Connect with me:

 LinkedIn: <https://www.linkedin.com/in/iulianatuhasu/>

 Email: tuhasu.iuliana@gmail.com

Prerequisites

Python installation

- Version: Python 3.11.2

Python Lib installation

- Python -m pip install pandas
- Python -m pip install matplotlib
- Python -m pip install requests

Power BI Desktop

- Version: 2.112.1161.0 64-bit (December 2022)

Power BI Settings

- Ensure Power BI Desktop is configured accordingly to run python scripts (File -> Options and settings -> Options -> Python scripting)



Call API using Python Script

- ▶ For the purpose of this session, I used a currency conversion endpoint (<https://apilayer.com>)
- ▶ **Goal:** To display using a Table Visualization the following information:
 - ▶ Current currency
 - ▶ Conversion date
 - ▶ Value after conversion
 - ▶ Target currency
 - ▶ Sum of money to be exchanged
- ▶ Get Data -> Other -> Python script

Get Data

All

File

Database

Power Platform

Azure

Online Services

Other

Other

- Microsoft Exchange
- Hadoop File (HDFS)
- Spark
- Hive LLAP
- R script
- Python script**
- ODBC
- OLE DB
- Acterys : Model Automation & Planning (Beta)
- Amazon OpenSearch Service (Beta)
- Anaplan
- Autodesk Construction Cloud (Beta)
- Solver
- BitSight Security Ratings
- BQE Core
- Bloomberg Data and Analytics

Certified Connectors

Template Apps

Connect

Cancel

Python script

```
import requests

import os, pandas as pd, matplotlib

url = "https://api.apilayer.com/exchangerates_data/convert?to=EUR&from=USD&amount=125"

payload = {}
headers= {
    "apikey":
    "zcl5Jw0d5fj5vosZhcQbiuyUTWoE5v51"
}

response = requests.request("GET", url,
headers=headers, data = payload)

status_code = response.status_code

result = response.text

df = pd.read_json(result)
```



Python script

Script

```
import requests
import os, pandas, matplotlib

url = "https://api.apilayer.com/exchangerates_data/convert?to=EUR&from=USD&amount=125"

payload = {}
headers= {
    "apikey": "zcl5Jw0d5fj5vosZhcQbiuyUTWoE5v51"
}

response = requests.request("GET", url, headers=headers, data = payload)
status_code = response.status_code
result = response.text
```

The script will run with the following Python installation

C:\Users\iuliana.tuhasu\AppData\Local\Programs\Python\Python311.

To configure your settings and change which Python installation you want to run, go to Options and settings.

OK

Cancel

Navigator



Display Options ▾



Python [1]

☒ df

df

success	query	info	date	result
TRUE	USD	null	2023-03-06	117.00625
TRUE	EUR	null	2023-03-06	117.00625
TRUE	125	null	2023-03-06	117.00625
TRUE		1678139943	2023-03-06	117.00625
TRUE		0.93605	2023-03-06	117.00625



Convert JSON to a Pandas DataFrame

Edit DataFrame

- ▶ Right click on “success” and “info” columns -> Remove columns
- ▶ Right click on “query” column -> Filter by unchecking the empty values
- ▶ Select all three columns query, date and result -> Unpivot Columns



Formula bar: `= Table.SelectRows("#Removed Columns", each ([query] <> ""))`

	query	date	result
1	USD		117.00625
2	EUR		117.00625
3	125		117.00625

Context menu options:

- Copy
- Remove Columns
- Remove Other Columns
- Add Column From Examples...
- Remove Duplicates
- Remove Errors
- Replace Values...
- Fill
- Change Type
- Merge Columns
- Group By...
- Unpivot Columns
- Unpivot Only Selected Columns
- Move

Unpivot other columns

Transform Add Column View Tools Help

Transpose Reverse Rows Count Rows

Table Use First Row as Headers

Any Column Data Type: Text Replace Values Unpivot Columns Detect Data Type Fill Move Rename Pivot Column Convert to List

Queries [2]

Query1

df

= Table.Distinct("#Unpivoted Columns")

Attribute	Value
1 query	USD
2 date	3/6/2023
3 result	117.00625
4 query	EUR
5 query	125

= Table.Transpose("#Removed Duplicates")

Column1	Column2	Column3	Column4	Column5
1 query	date	result	query	query
2 USD	3/6/2023	117.00625	EUR	125

Remove duplicates

File Home Transform Add Column View Tools Help

Close & Apply New Source Recent Sources Enter Data Data source settings Manage Parameters Refresh Preview Properties Advanced Editor Manage

Close New Query Data Sources Parameters Query

Choose Columns Remove Columns Keep Rows Remove Rows Split Column Group By

Remove Top Rows Remove Bottom Rows Remove Alternate Rows Remove Duplicates Remove Blank Rows Remove Errors

Queries [2]

Query1

df

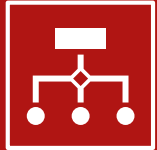
= Table.UnpivotOtherColumns("#Filtered Rows", {}, "Attribute", "Value")

Attribute	Value
1 query	USD
2 date	3/6/2023
3 result	117.00625
4 query	EUR
5 date	3/6/2023
6 result	117.00625
7 query	125
8 date	3/6/2023
9 result	117.00625

Select Transpose
(Transform tab)



1. Go to Column1 and filter out the “query” value



2. Rename each column:

- ✓ Column1 -> Current currency
- ✓ Column2 -> Conversion date
- ✓ Column3 -> Value after conversion
- ✓ Column4 -> Target currency
- ✓ Column5 -> Sum of money to be exchanged

Table.SelectRows("#Transposed Table", each ([Column1] = "USD"))

Column1	Column2	Column3	Column4	Column5
1 USD	3/6/2023	117.00625	EUR	125

Table.RenameColumns("#Filtered Rows1",{"Column1", "Current currency"}, {"Column2", "Conversion date"}, {"Column3", "Value after conversion"}, {"Column4", "Target currency"}, {"Column5", "Sum of money to be exchanged"})

Current currency	Conversion date	Value after conversion	Target currency	Sum of money to be exchanged
1 USD	2023-03-06	117.00625	EUR	125

Visualize data

- ✓ Choose a Table visualization and add all the columns

Conversion date	Current currency	Sum of money to be exchanged	Target currency	Value after conversion
2023-03-15	USD	125	EUR	118.41375

Q&A

- <https://sqlb.it/?9899>



Thank you!