

Aula 12 - APIS

A sigla API deriva da expressão inglesa *Application Programming Interface* que, traduzida para o português, pode ser compreendida como uma interface de programação de aplicação. Ou seja, API é um conjunto de normas que possibilita a comunicação entre plataformas através de uma série de padrões e protocolos.

forma para a API e ela nos retorna algo (podemos também mandar requisições para realizar alterações, etc...porém não é o escopo desta aula.)

informações geográficas por meio de sua API. Por meio da seguinte URL:

Conseguimos acessar as distâncias entre as 3 localizações (longitude latitude) contidas na URL. A resposta vem em um formato chamado

550N (isso depende da API e dos parâmetros passados). Com esse formato, conseguimos extrair os dados necessários.

insira a mesma no seguinte site (<http://jsonviewer.stack.hu/>), e veja como a informações esta estruturada.

12.1.5. Example: $\tilde{\mathcal{L}} = \mathcal{L}$, $\mathcal{A} = \mathcal{B}$, $\mathcal{C} = \mathcal{D}$

Para coeltarmos a mesma informação acima direto do nosso programa em Pyt

```
import requests as rq
```

```

respostas = rq.get("https://economia.awesomeapi.com.br/last/USD-BRL,EUR-BRL,BTC-BRL")

# As respostas estão em um json que está dentro de respostas
respostas.json()[ 'USDBRL' ]

Out[2]: {'code': 'USD',
          'codein': 'BRL',
          'type': 'BRL',
          'last': 5.27,
          'high': 5.27,
          'low': 5.27,
          'open': 5.27,
          'close': 5.27,
          'preclose': 5.27,
          'bid': 5.27,
          'ask': 5.27,
          'timestamp': 1544256000,
          'timestamp_utc': 1544256000,
          'is_early': False,
          'is_late': False}

```

```
'high': '5.3096',
'low': '5.2459',
'varBid': '0.0077',
'pctChange': '0.15',
'bid': '5.2541',
'ask': '5.2553',
'timestamp': '1663361997',
'create_date': '2022-09-16 17:59:57'})
```

12.2 - Exemplos de APIs em Python

API que fornece a hora atualizada (<http://worldtimeapi.org>):

Horário mundial

```
respostas = rq.get("http://worldtimeapi.org/api/timezone/America/Sao_Paulo")
respostas.json()[ "datetime" ]
```

```
Out[10]:      2022-09-11T17:43:13.123733-05:00
```

Geoprocessamento/ matriz de distancias

[979-0-12816-574-9](#)

```
respostas = rq.get('http://router.project-osrm.org/table/v1/driving/-49.26742667248318,-25.43800114521021;-49.26742667248318,-25.43800114521021')
M = respostas.json()["distances"]
M
```

Mapas do gogole maps

```
chave = "XXXXXXXXXXXXXXXXXXXXX"
respostas = rq.get("https://maps.googleapis.com/maps/api/staticmap?center=" + local + "&zoom=14&size=400x400&key=" + chave)

# Abre um arquivo para salvar o conteúdo
f = open(r'G:\Meu Drive\Arquivos\UFPR\Disciplinas\2 - Intro Mineração de Dados\file.jpeg', 'wb')
f.write(respostas.content)
f.close()

# Abre a imagem
```

The image is a composite of two parts. The top part is a map of a section of Curitiba, Brazil, showing streets like Rua 24 Horas, Av. Iguaçu, and Av. Chile, and landmarks such as the Museu Paranaense, Praça Tiradentes, Mercado Municipal de Curitiba, and the Estádio Joaquim Américo Guimarães. The bottom part is a screenshot of a Jupyter Notebook with two code cells. The first cell uses the requests library to fetch data from 'https://api.siorg.economia.gov.br/respostas' and prints the response. The second cell fetches data from 'https://dadosabertos.camara.leg.br/api/v2/eventos' and prints the response for a specific event ID (66366). The output of the second cell shows a detailed JSON object for an event titled 'Vacinação das pessoas com diabetes e com obesidade'.

Mercado de ações

Para coletar dados de ações podemos usar o próprio pacote do pandas `datareader`

```
In [4]: import pandas as pd
from pandas_datareader import data as web

# Bovespa
df = web.DataReader('F^BVSP', data_source='yahoo', start='02-20-2020', end='02-20-2021')
print(df)

# Facebook
df2 = web.DataReader("FB", data_source='yahoo', start='08-20-2020', end='12-20-2020')
print(df2)

# Alcoa corporation
df3 = web.DataReader("AA", data_source='yahoo', start='08-20-2020', end='12-20-2020')
print(df3)
```

| Date | High | Low | Open | Close | Volume | Adj Close |
|------------|--------|--------|--------|--------|----------|-----------|
| 2020-02-20 | 116552 | 114379 | 116518 | 114586 | 6793000 | 114586 |
| 2020-02-21 | 114585 | 112661 | 114585 | 113681 | 5896200 | 113681 |
| 2020-02-26 | 113647 | 105053 | 113647 | 105718 | 9369800 | 105718 |
| 2020-02-27 | 106656 | 102984 | 105711 | 102984 | 11237700 | 102984 |
| 2020-02-28 | 104172 | 99951 | 102984 | 104172 | 11228400 | 104172 |
| ... | ... | ... | ... | ... | ... | ... |
| 2021-02-11 | 120283 | 118440 | 118440 | 119235 | 7567400 | 119235 |
| 2021-02-12 | 119763 | 118163 | 119300 | 119116 | 7655900 | 119116 |
| 2021-02-17 | 120573 | 118880 | 119421 | 120391 | 8326100 | 120391 |
| 2021-02-18 | 120845 | 118515 | 120361 | 119140 | 10720600 | 119140 |
| 2021-02-19 | 119250 | 117867 | 119199 | 118748 | 10402800 | 118748 |

High

| Date | | | | | |
|------------|------------|------------|------------|------------|----------|
| 2021-12-07 | 326.540009 | 321.000000 | 321.570007 | 322.809998 | 18794000 |
| 2021-12-08 | 332.750000 | 323.070007 | 325.000000 | 330.559998 | 19937700 |
| 2021-12-09 | 336.130005 | 328.000000 | 329.540009 | 329.820007 | 16879200 |
| 2021-12-10 | 335.029999 | 326.369995 | 325.559998 | 329.750000 | 14528000 |
| 2021-12-13 | 341.089996 | 329.589996 | 330.950012 | 334.489990 | 22948700 |
| ... | ... | ... | ... | ... | ... |
| 2022-12-01 | 121.199997 | 118.400002 | 119.199997 | 120.440002 | 36551400 |
| 2022-12-02 | 124.040001 | 117.610001 | 117.830002 | 123.489998 | 39875100 |
| 2022-12-05 | 124.669998 | 121.249999 | 121.750000 | 122.430000 | 35474900 |

2022-12-06 120 550003

| Date | Adj | Close |
|------------|------------|-------|
| 2021-12-07 | 322.809998 | |
| 2021-12-08 | 330.559998 | |
| 2021-12-09 | 329.820007 | |
| 2021-12-10 | 329.750000 | |
| 2021-12-13 | 334.489990 | |

• • • • •

| | | | | | | |
|------------------------|------------|-----------|-----------|-------|---------|-----------|
| 2022-12-01 | 120.440002 | | | | | |
| 2022-12-02 | 123.489998 | | | | | |
| 2022-12-05 | 122.430000 | | | | | |
| 2022-12-06 | 114.120003 | | | | | |
| 2022-12-07 | 113.430000 | | | | | |
| [253 rows x 6 columns] | | | | | | |
| | High | Low | Open | Close | Volume | Adj Close |
| Date | | | | | | |
| 2020-08-20 | 15.245000 | 14.730000 | 15.040000 | 15.19 | 3140900 | 15.045506 |

2020-08-21 15.040000

| | | | | | | |
|------------|-----------|-----------|-----------|-------|---------|-----------|
| 2020-08-24 | 15.130000 | 14.700000 | 14.850000 | 15.12 | 3171200 | 14.967611 |
| 2020-08-25 | 15.250000 | 14.820000 | 15.120000 | 15.05 | 2363300 | 14.906838 |
| 2020-08-26 | 15.440000 | 15.020000 | 15.090000 | 15.07 | 3158500 | 14.926647 |
| 2020-12-14 | 23.375000 | 21.667999 | 23.250000 | 21.76 | 6286200 | 21.553309 |
| 2020-12-15 | 22.600000 | 21.758999 | 22.150000 | 22.17 | 4226500 | 21.959108 |
| 2020-12-16 | 22.129999 | 21.304001 | 22.059999 | 21.58 | 5423800 | 21.374723 |
| 2020-12-17 | 23.209999 | 21.740000 | 22.280001 | 22.18 | 7318400 | 21.969011 |
| 2020-12-18 | 22.610001 | 21.860001 | 22.209999 | 22.01 | 5137200 | 21.800631 |