UNIVERSIDADE DA REGIÃO DA CAMPANHA
CENTRO DE CIÊNCIAS EXATAS E AMBIENTAIS
CURSO DE SISTEMAS DE INFORMAÇÃO
FLISOL/BAGÉ

Python para finanças: explorando dados financeiros!

Alex Camargo

alex@apus.digital

Abril 15, 2023

About me

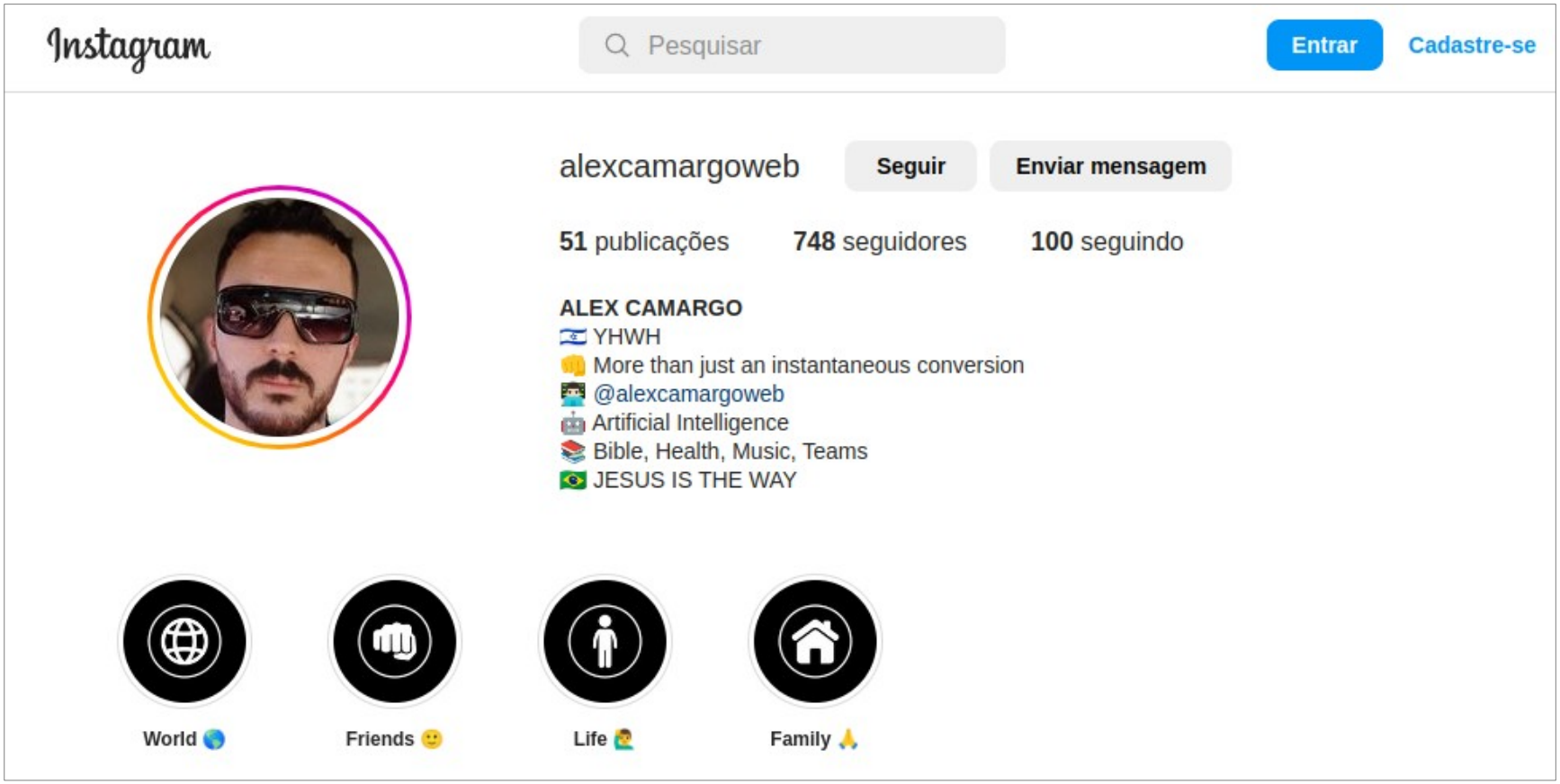

Figure: Instagram @alexcamargoweb.

Edições anteriores

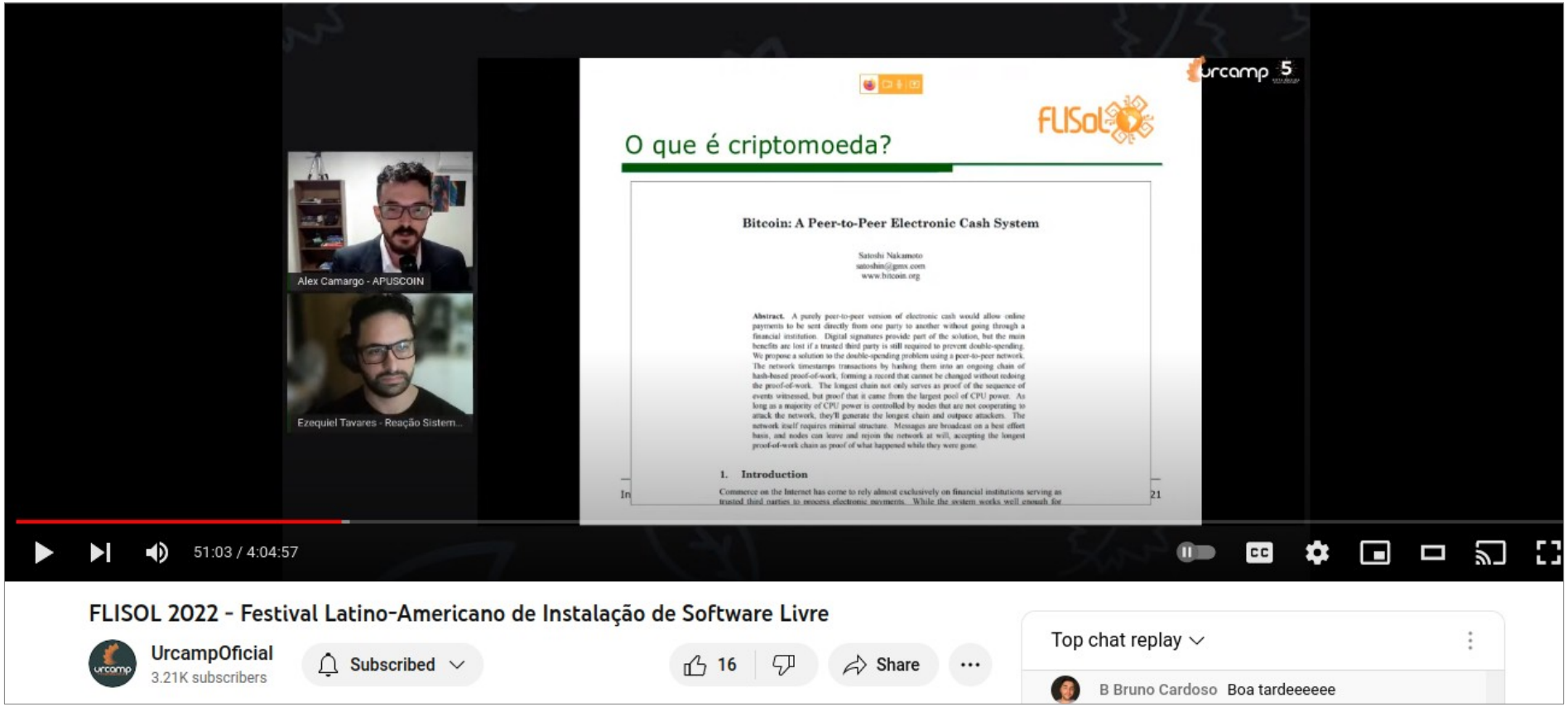

Figure: FLISOL 2022.

Agenda

- **Introdução ao mercado financeiro**
- **Python na área financeira**
- **Código *Python* para finanças**
- **Considerações finais**
- **Referências**

Introdução ao mercado financeiro

O mercado financeiro é um **sistema de negociação de ativos** composto por instituições financeiras, investidores e empresas **com o objetivo de obter um retorno** sobre o investimento.

- **Tipos de mercado**: mercado de **ações**, mercado de **títulos**, mercado de **câmbio** e o mercado de ***commodities***.

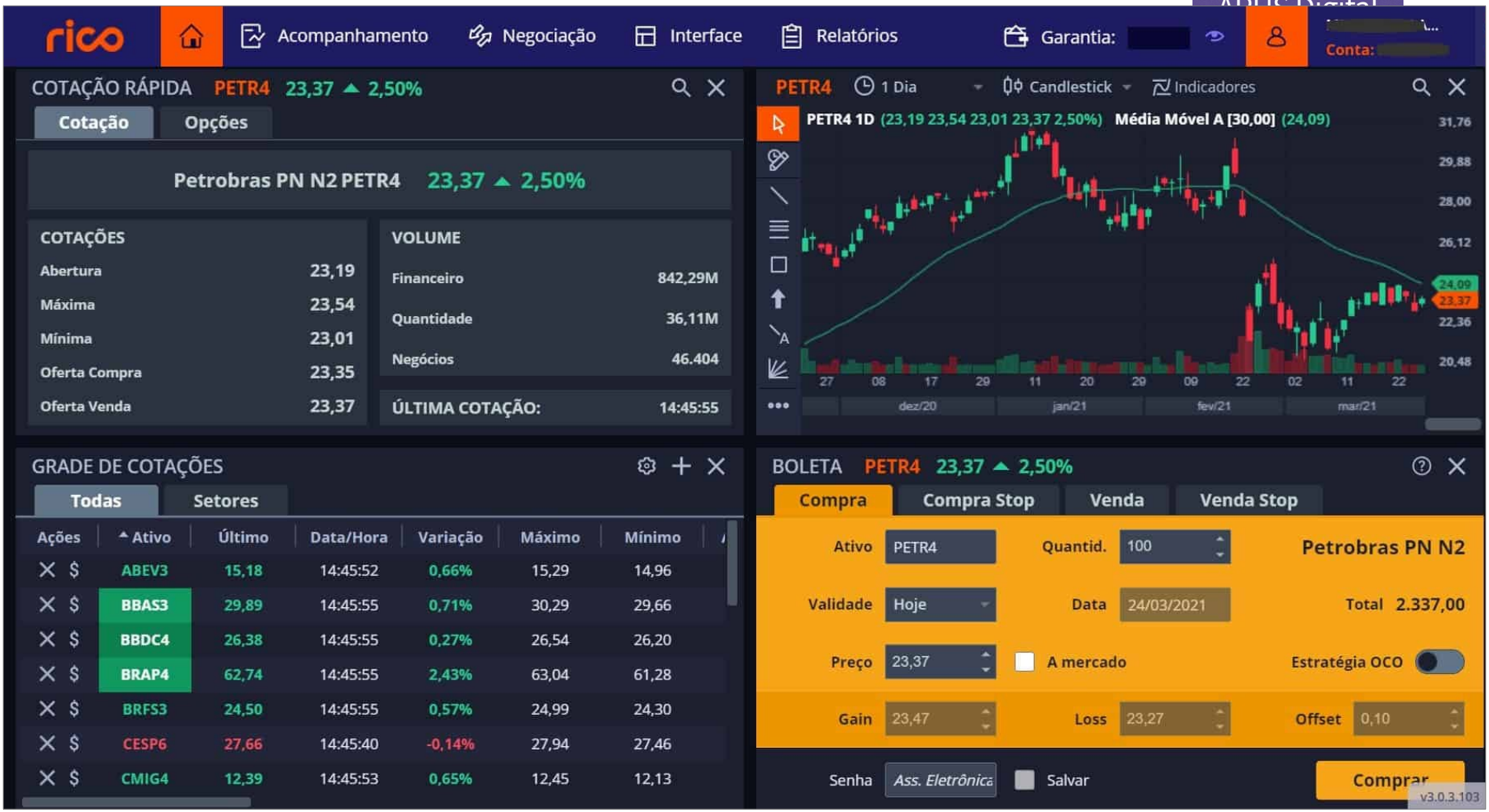

Figura: Tipos de mercado financeiro - Ações.

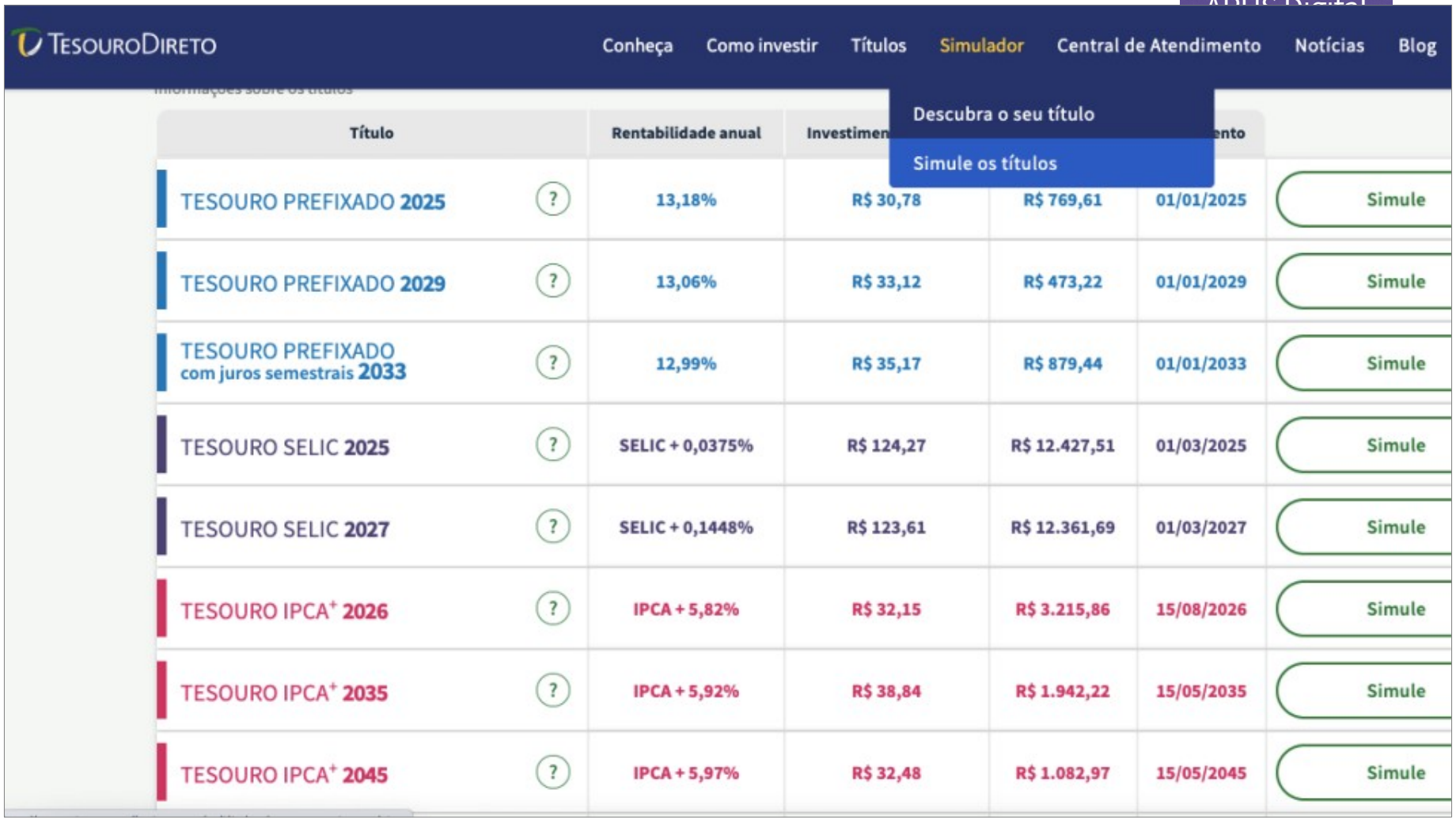

Figura: Tipos de mercado financeiro – Títulos públicos.

Introdução ao mercado financeiro

Figura: Tipos de mercado financeiro – Câmbio.

Figura: Tipos de mercado financeiro – *Commodities*.

Introdução ao mercado financeiro

O mercado financeiro é um **sistema de negociação de ativos** composto por instituições financeiras, investidores e empresas **com o objetivo de obter um retorno** sobre o investimento.

- **Tipos de mercado**: mercado de **ações**, mercado de **títulos**, mercado de **câmbio** e o mercado de ***commodities***.
- **Participantes**: investidores, ***traders***, bancos e outras instituições financeiras (seguradoras, fundos de pensão, etc).

Introdução ao mercado financeiro

Figura: Crescimento de investidores no Brasil.

Introdução ao mercado financeiro

O mercado financeiro é um **sistema de negociação de ativos** composto por instituições financeiras, investidores e empresas **com o objetivo de obter um retorno** sobre o investimento.

- **Tipos de mercado**: mercado de **ações**, mercado de **títulos**, mercado de **câmbio** e o mercado de ***commodities***.
- **Participantes**: investidores, ***traders***, bancos e outras instituições financeiras (seguradoras, fundos de pensão, etc).
- **Regulamentação**: regulamentado por agências do governo, como a SEC nos Estados Unidos e a **CVM no Brasil**.

Figura: CVM – Comissão de Valores Mobiliários.

Introdução ao mercado financeiro

O mercado financeiro é um **sistema de negociação de ativos** composto por instituições financeiras, investidores e empresas **com o objetivo de obter um retorno** sobre o investimento.

- **Tipos de mercado**: mercado de **ações**, mercado de **títulos**, mercado de **câmbio** e o mercado de ***commodities***.
- **Participantes**: investidores, ***traders***, bancos e outras instituições financeiras (seguradoras, fundos de pensão, etc).
- **Regulamentação**: regulamentado por agências do governo, como a SEC nos Estados Unidos e a **CVM no Brasil**.
- **Fluxo de capitais**: o fluxo de capitais é o **movimento de dinheiro** dentro e fora do mercado financeiro.

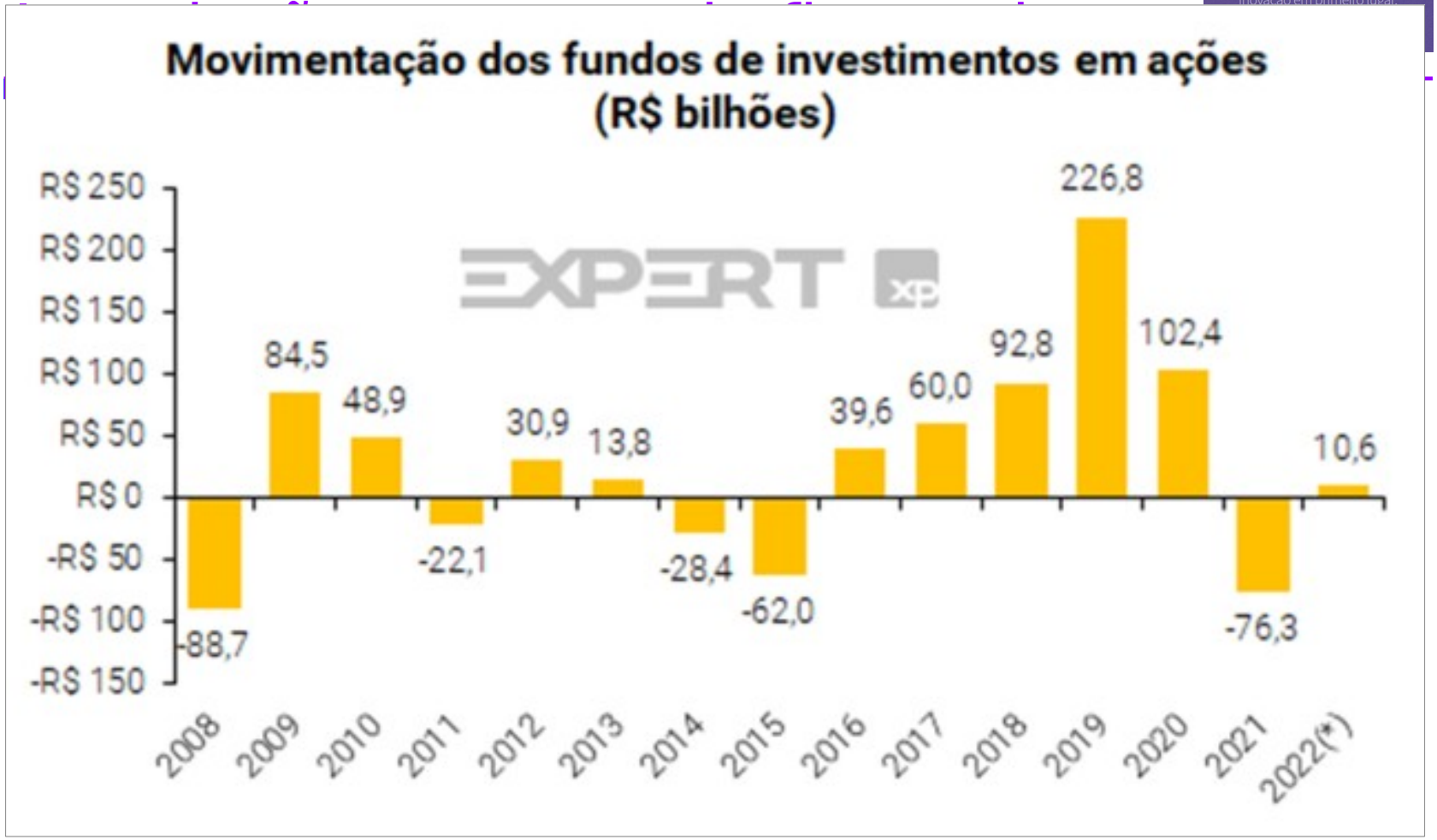

Figura: Fluxo de capital no mercado financeiro.

Introdução ao mercado financeiro

O mercado financeiro é um **sistema de negociação de ativos** composto por instituições financeiras, investidores e empresas **com o objetivo de obter um retorno** sobre o investimento.

- **Tipos de mercado**: mercado de **ações**, mercado de **títulos**, mercado de **câmbio** e o mercado de ***commodities***.
- **Participantes**: investidores, ***traders***, bancos e outras instituições financeiras (seguradoras, fundos de pensão, etc).
- **Regulamentação**: regulamentado por agências do governo, como a SEC nos Estados Unidos e a **CVM no Brasil**.
- **Fluxo de capitais**: o fluxo de capitais é o **movimento de dinheiro** dentro e fora do mercado financeiro.
- **Bolsa de valores**: local onde as **empresas podem emitir ações** para captação de recursos e os **investidores podem comprar e vender** essas ações.

Figura: B3 – Bolsa de Valores do Brasil (antiga Bovespa).

Figura: B3 – Bolsa de Valores do Brasil (antiga Bovespa).

Intro

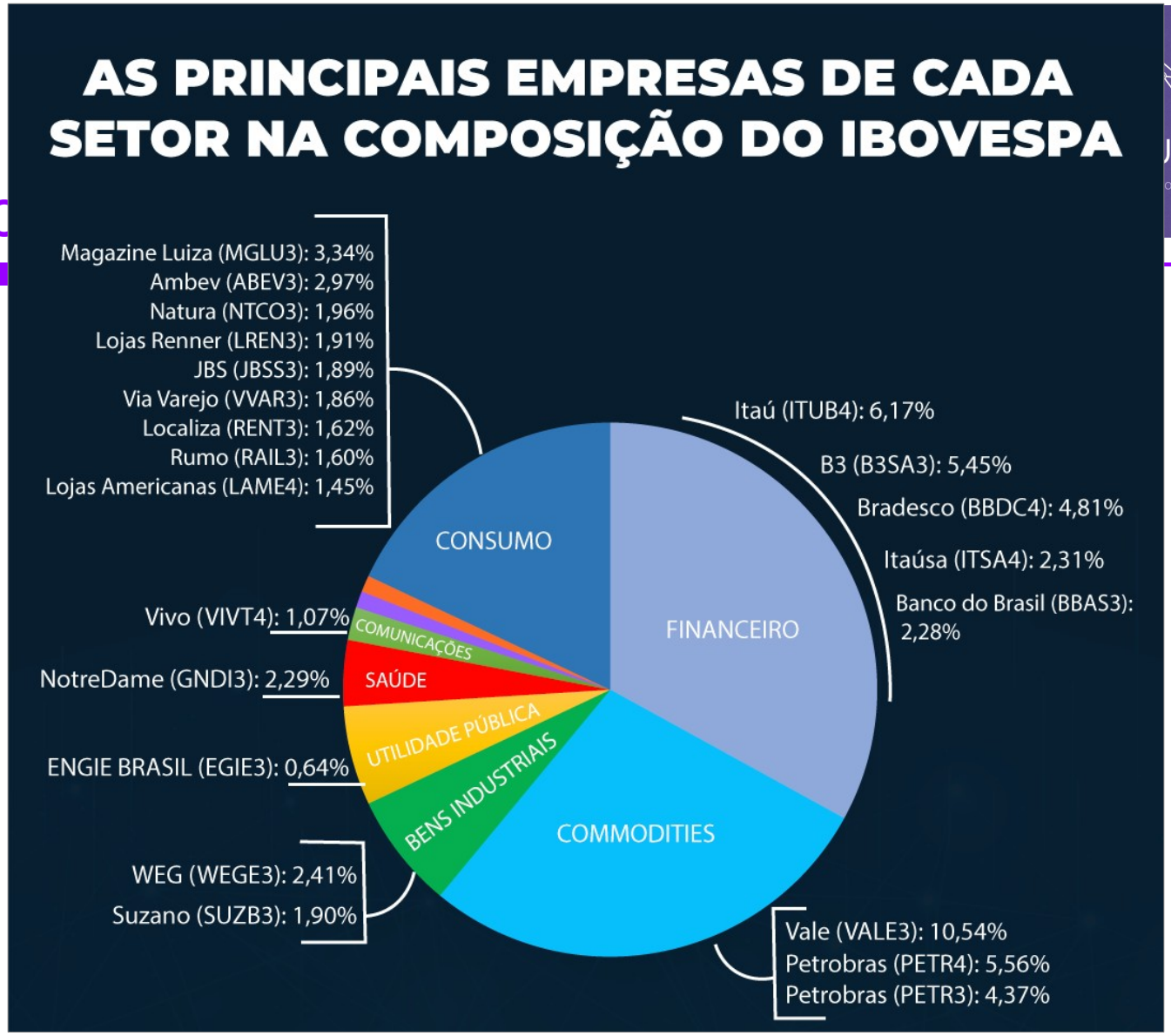

Figura: Principais empresas de cada setor na composição do índice IBOVESPA.

Python na área financeira

Python é utilizado em uma variedade de aplicações, incluindo **desenvolvimento *web***, **ciência de dados**, **automação de processos**, **inteligência artificial**, entre outras.

- **Bibliotecas**: são usadas *libs* gerais, como ***pandas***, ***numpy***, ***matplotlib***, e outras mais específicas de finanças, como o ***yfinance***.
- **Coleta de dados**: buscar, limpar e processar dados financeiros de diversas fontes, como **APIs de mercado**, **planilhas** e **bancos de dados**.
- **Visualização de dados**: o ***plotly*** e o ***seaborn*** são *libs* comumente usadas para criar visualizações personalizadas e interativas.
- **Modelagem financeira**: cria **modelos de precificação** de ativos, **simulações de Monte Carlo** para avaliar o risco e a incerteza em um modelo.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
import pandas as pd

url = 'https://raw.githubusercontent.com/apusdigital/flisol-2023/main/datasets/AAPL.csv'

# Carrega os dados do arquivo CSV
df = pd.read_csv(url, index_col='Date', parse_dates=True)

# Calcula os retornos diários
df['Return'] = df['Close'].pct_change()

# Calcula o retorno diário médio
daily_return_mean = df['Return'].mean()
print("Retorno diário médio:", daily_return_mean)

# Calcula a volatilidade histórica
daily_volatility = df['Return'].std()
print("Volatilidade histórica:", daily_volatility)
```

```
Retorno diário médio: -0.004700511517994327
Volatilidade histórica: 0.012989624617195269
```

Figura: Bibliotecas *Python* usadas na área financeira - *pandas*.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
import pandas as pd
import matplotlib.pyplot as plt

url = 'https://raw.githubusercontent.com/apusdigital/flisol-2023/main/datasets/AAPL.csv'

# Carrega os dados do arquivo CSV
df = pd.read_csv(url, index_col='Date', parse_dates=True)

# Cria um gráfico de linhas
fig, ax = plt.subplots(figsize=(10, 7))
ax.plot(df.index, df['Close'], label='Preço de fechamento')
ax.set_xlabel('Data')
ax.set_ylabel('Preço de fechamento ($)')
ax.set_title('AAPL - Preço de fechamento NASDAQ (Apple Inc)')
ax.legend()

# Mostra o gráfico
plt.show()
```

Figura: Bibliotecas *Python* usadas na área financeira - *matplotlib*.

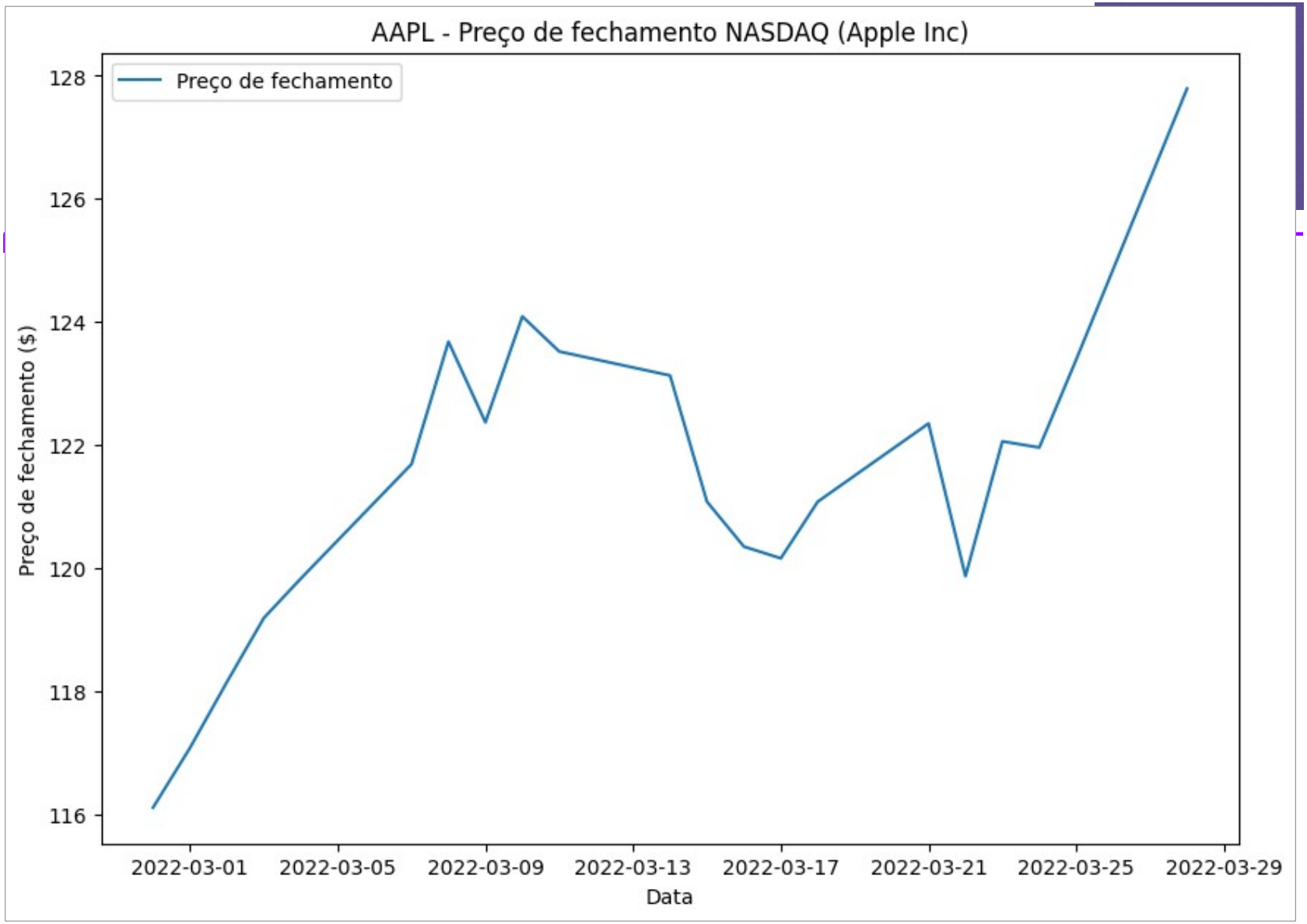

Figura: Bibliotecas *Python* usadas na área financeira - *matplotlib*.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
import yfinance as yf

# Obter dados da ação AAPL
stock = yf.Ticker("AAPL")

# Obter informações gerais sobre a ação
info = stock.info

# Imprimir informações sobre a ação
print(info['symbol'])
print(info['language'])
print(info['currency'])
print('\n')

# Obter o histórico de dividendos
dividendos = stock.dividends

# Imprimir o histórico de dividendos
print(dividendos)
```

Figura: Bibliotecas *Python* usadas na área financeira - *yfinance*.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
AAPL
en-US
USD


Date
1987-05-11 00:00:00-04:00    0.000536
1987-08-10 00:00:00-04:00    0.000536
1987-11-17 00:00:00-05:00    0.000714
1988-02-12 00:00:00-05:00    0.000714
1988-05-16 00:00:00-04:00    0.000714
                               ...
2022-02-04 00:00:00-05:00    0.220000
2022-05-06 00:00:00-04:00    0.230000
2022-08-05 00:00:00-04:00    0.230000
2022-11-04 00:00:00-04:00    0.230000
2023-02-10 00:00:00-05:00    0.230000
Name: Dividends, Length: 78, dtype: float64
```

Figura: Bibliotecas *Python* usadas na área financeira - *yfinance*.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

```python
import yfinance as yf
import pandas as pd

# Lista de ações a serem analisadas
tickers = ['AAPL', 'MSFT', 'AMZN']

# Data de início e fim da coleta de dados
start_date = '2020-01-01'
end_date = '2022-03-29'

# Coleta de dados
data = yf.download(tickers, start=start_date, end=end_date)['Close']

# Renomear colunas com os nomes das ações
data.columns = tickers

# Visualização dos dados coletados
print(data.head())
```

Figura: Coleta de dados - *yfinance*.

Python na área financeira

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
# Visualização dos dados coletados
print(data.head())
```

```
[*********************100%***********************]  3 of 3 completed
                 AAPL       MSFT        AMZN
Date
2020-01-02  75.087502  94.900497  160.619995
2020-01-03  74.357498  93.748497  158.619995
2020-01-06  74.949997  95.143997  159.029999
2020-01-07  74.597504  95.343002  157.580002
2020-01-08  75.797501  94.598503  160.089996
```

Figura: Coleta de dados - *yfinance*.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
# Faz a requisição HTTP para a URL e obtém o conteúdo da página
response = requests.get(url)
html_content = response.content

# Cria um objeto BeautifulSoup com o conteúdo da página
soup = BeautifulSoup(html_content, 'html.parser')

# Extrai o preço atual da ação da Apple (AAPL) e exibe na tela
price_element = soup.find('fin-streamer', {'class': 'Fw(b) Fz(36px) Mb(-4px) D(ib)'})

# Imprime o elemento completo
print(price_element)

# Imprime somente o preco atual da ação
price = price_element.text
print('Preço da ação da Apple (AAPL):', price)
```

```
<fin-streamer active="" class="Fw(b) Fz(36px) Mb(-4px) D(ib)" data-field="regularMarketPrice"
Preço da ação da Apple (AAPL): 163.07
```

Figura: Coleta de dados - *BeautifulSoup*.

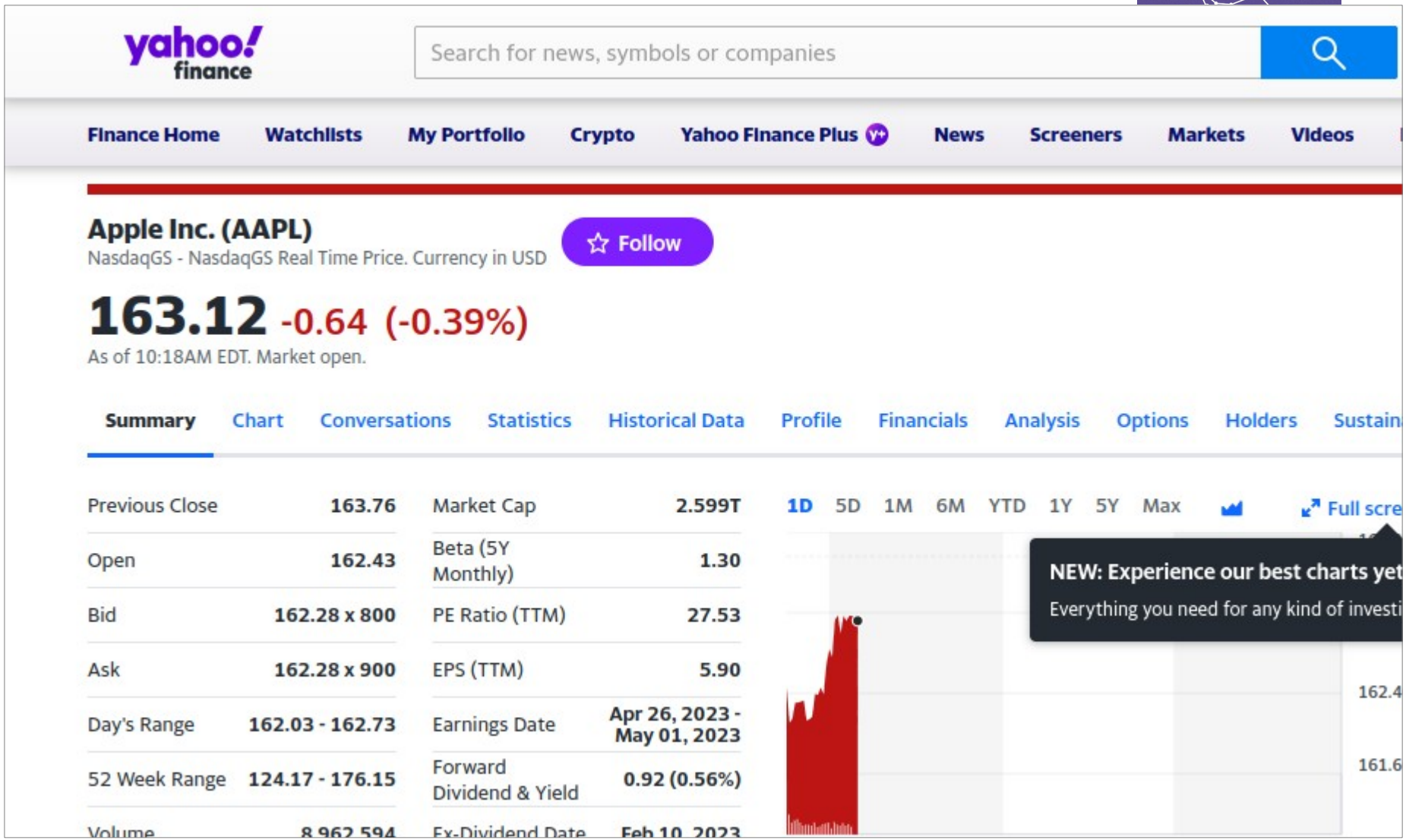

Figura: Coleta de dados - *BeautifulSoup*.

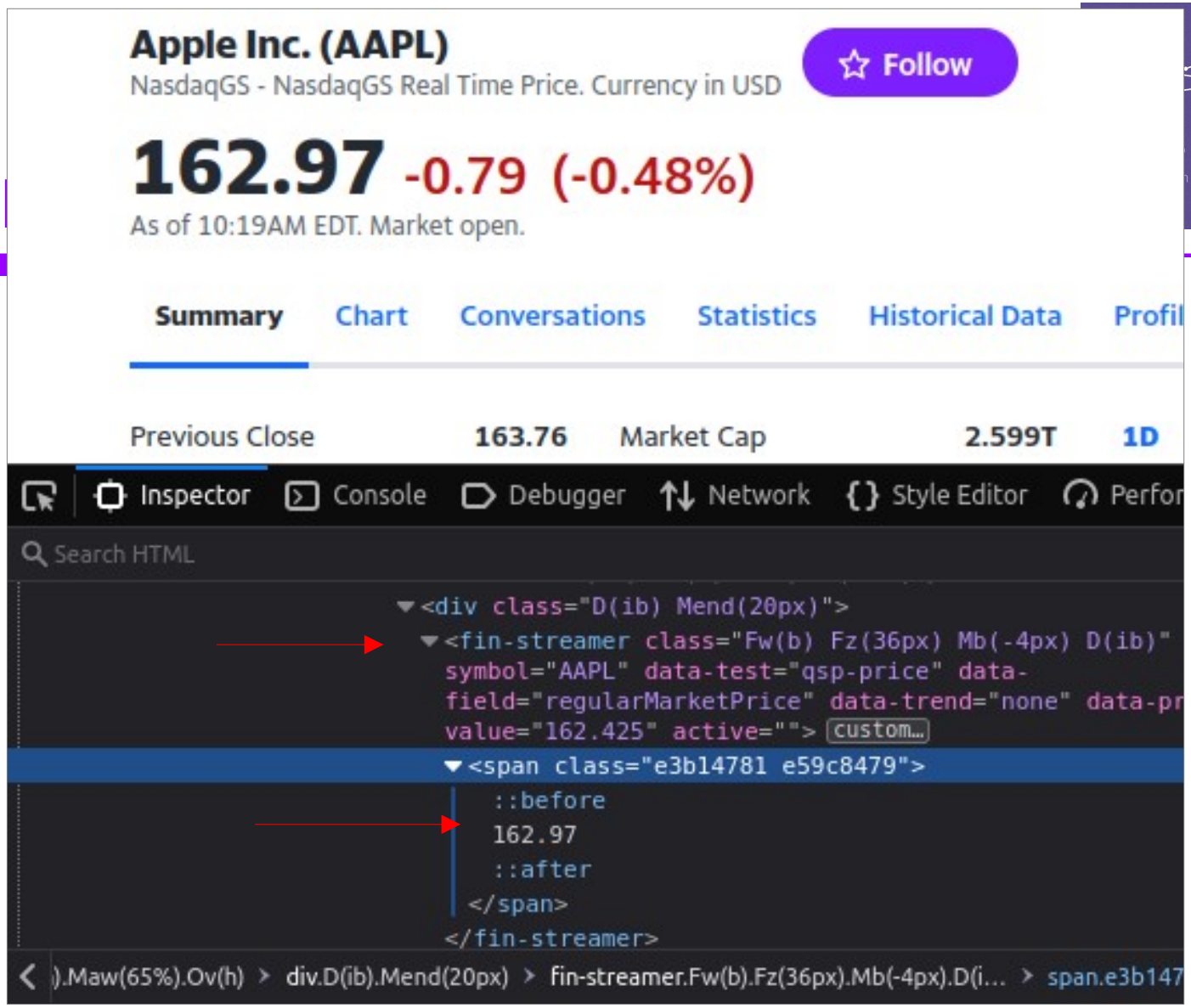

Figura: Coleta de dados - *BeautifulSoup*.

Python na área financeira

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
# Obtém os dados das ações utilizando o yfinance
data = yf.download(tickers, start=start_date, end=end_date)['Adj Close']

# Cria um gráfico interativo com o plotly
fig = go.Figure()

for ticker in tickers:
    fig.add_trace(go.Scatter(x=data.index, y=data[ticker], name=ticker, line=dict(width=2),))

fig.update_layout(
    title='Preços das ações',
    xaxis_title='Data',
    yaxis_title='Preço (USD)',
    xaxis_rangeslider_visible=True,
    height=600,
)

fig.show()
```

Figura: Visualização de dados - *plotly*.

Python na área financeira

Figura: Visualização de dados - *plotly*.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
# Define os parâmetros da simulação
S = 100  # Preço inicial da ação
mu = 0.1  # Retorno médio diário
sigma = 0.2  # Volatilidade diária
dt = 1/252  # Intervalo de tempo (um dia)

# Define o número de simulações e o número de dias a serem simulados
num_simulacoes = 5
num_dias = 31

# Gera uma matriz com números aleatórios normalmente distribuídos
# (correspondentes aos retornos diários da ação)
r = np.random.normal((mu*dt), (sigma*np.sqrt(dt)), (num_dias, num_simulacoes))

# Calcula o preço da ação para cada simulação e cada dia
preco = S * np.exp(np.cumsum(r, axis=0))

# Plota o gráfico das simulações de Monte Carlo
plt.figure(figsize=(10, 6))
plt.plot(preco)
plt.xlabel('Dia')
plt.ylabel('Preço da ação')
plt.title('Simulação de Monte Carlo para o preço da ação')
plt.show()
```

Figura: Modelagem financeira com simulações de Monte Carlo - *matplotlib*.

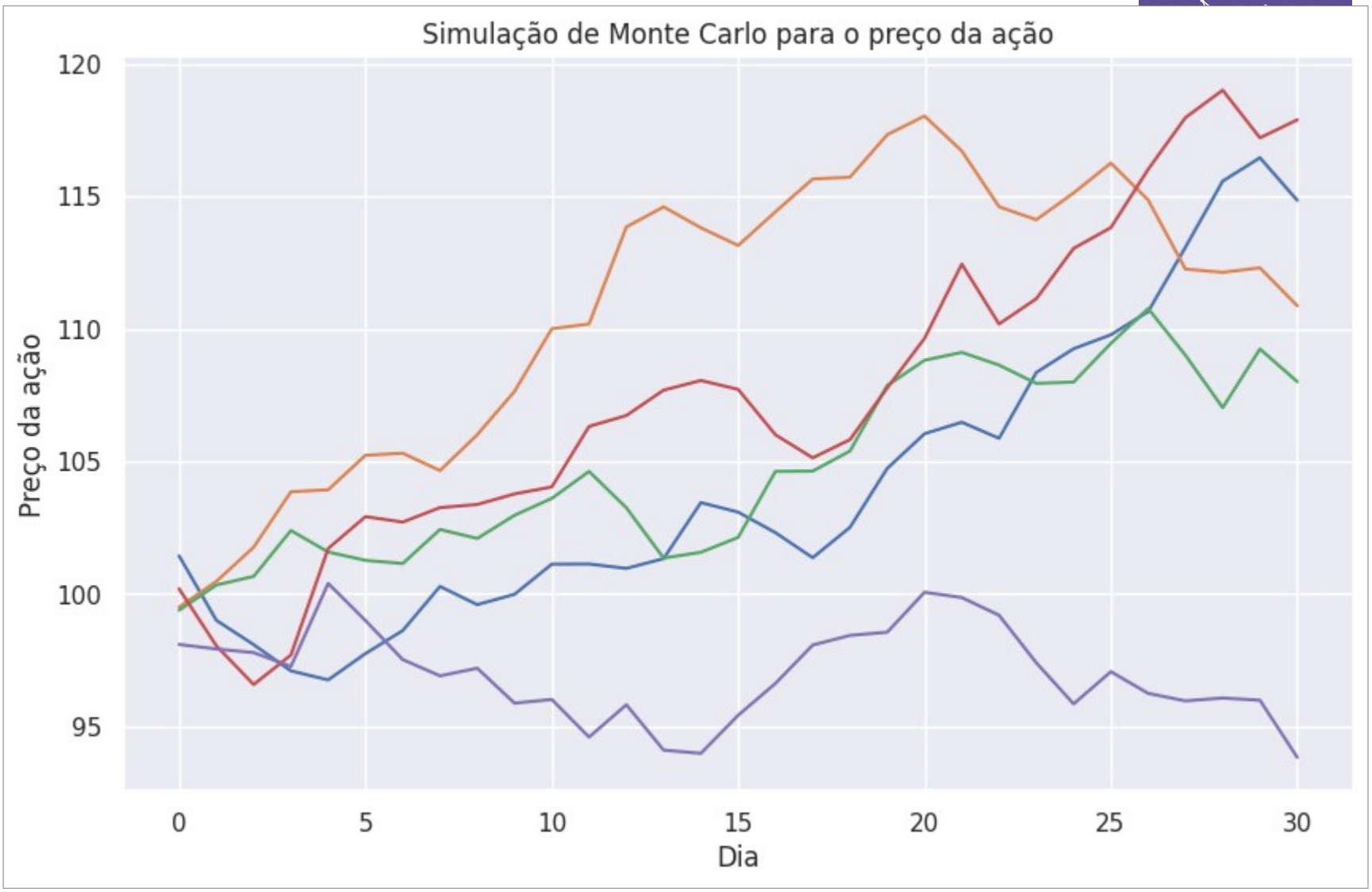

Figura: Modelagem financeira com simulações de Monte Carlo - *matplotlib*.

Código *Python* para finanças

Usaremos o ***Google Colab***, uma plataforma de desenvolvimento **em nuvem** que permite aos usuários **escrever, executar e compartilhar código em *Python***.

- **Perfomance**: acesso gratuito (ou planos pagos) a **GPUs e TPUs** para processamento acelerado de dados.

flisol-2023.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```python
from tensorflow.python.client import device_lib
device_lib.list_local_devices()
```

```
[name: "/device:CPU:0"
 device_type: "CPU"
 memory_limit: 268435456
 locality {
 }
 incarnation: 4553555212738941057
 xla_global_id: -1,
 name: "/device:GPU:0"
 device_type: "GPU"
 memory_limit: 14343274496
 locality {
   bus_id: 1
   links {
   }
 }
 incarnation: 9020637400098626012
 physical_device_desc: "device: 0, name: Tesla T4, pci bus id: 0000:00:04.0,
 xla_global_id: 416903419]
```

Figura: Código *Python* para finanças – Recursos do *Google Colab*.

Código *Python* para finanças

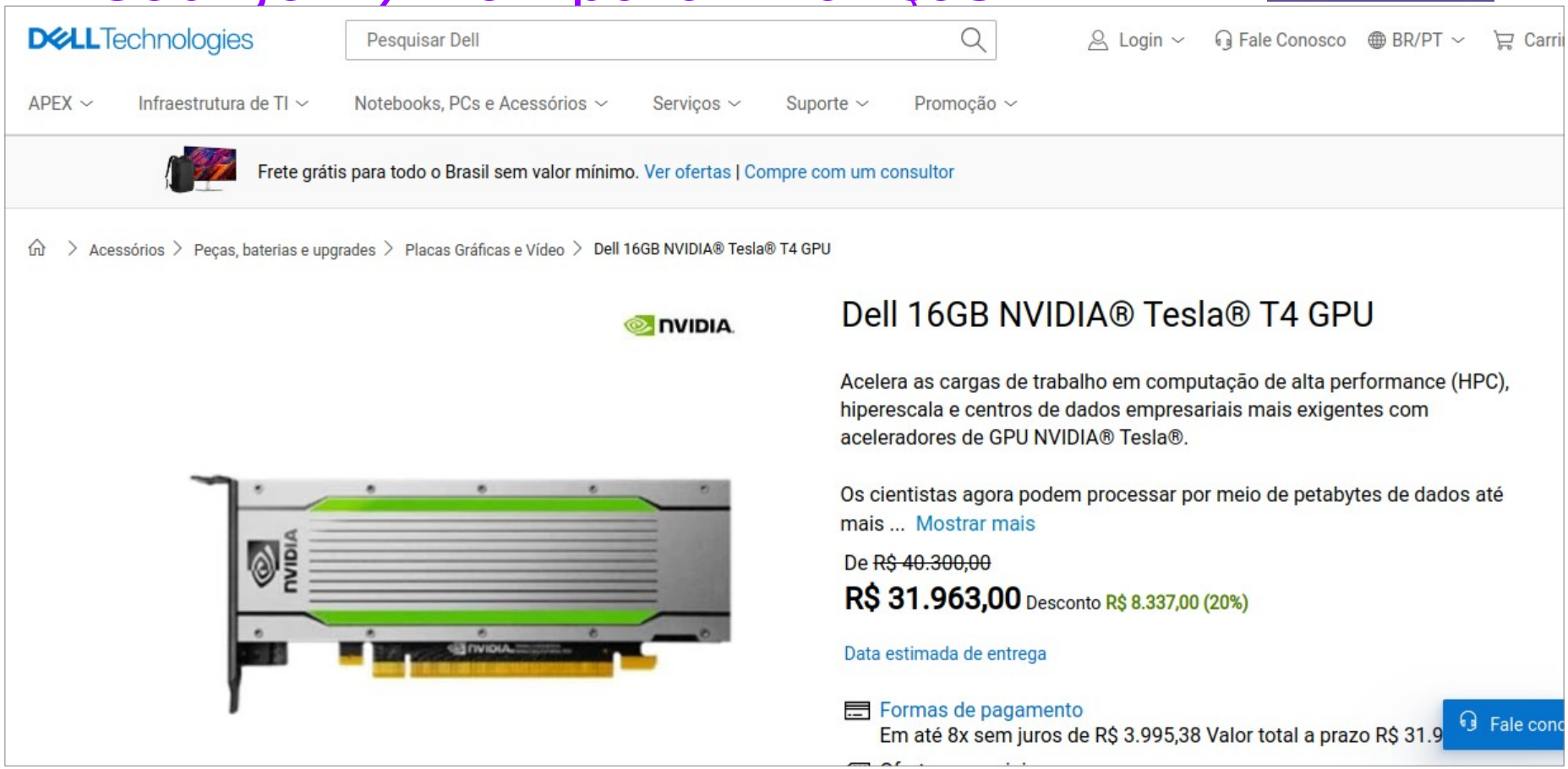

Figura: Código *Python* para finanças – Recursos do *Google Colab*.

Código *Python* para finanças

Usaremos o ***Google Colab***, uma plataforma de desenvolvimento **em nuvem** que permite aos usuários **escrever, executar e compartilhar código em *Python***.

- **Perfomance**: acesso gratuito (ou planos pagos) a **GPUs e TPUs** para processamento acelerado de dados.
- **Armazenamento e uso**: os arquivos **ficam em nuvem** vinculados ao ***Google Drive***. É possível instalar bibliotecas durante as sessões de uso via ***Jupyter Notebooks***.

Código *Python* para finanças

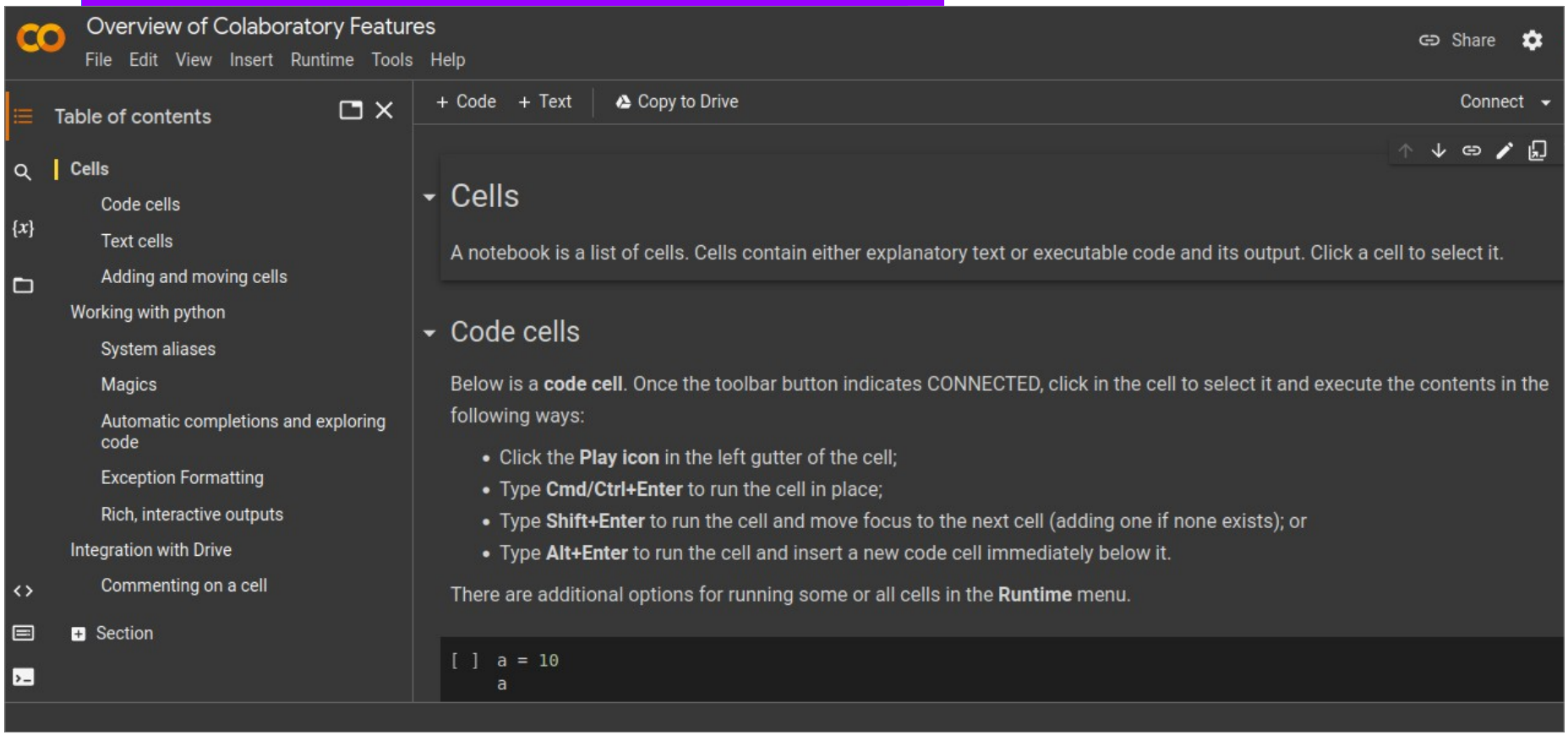

Figura: Código *Python* para finanças – *Jupyter Notebook*.

Código *Python* para finanças

Usaremos o ***Google Colab***, uma plataforma de desenvolvimento **em nuvem** que permite aos usuários **escrever, executar e compartilhar código em *Python***.

- **Perfomance**: acesso gratuito (ou planos pagos) a **GPUs e TPUs** para processamento acelerado de dados.
- **Armazenamento e uso**: os arquivos **ficam em nuvem** vinculados ao ***Google Drive***. É possível instalar bibliotecas durante as sessões de uso via ***Jupyter Notebooks***.
- **Estudo de caso**: como acessar fontes de dados financeiros do ***Yahoo Finance*** e visualizar as ações do **NUBANK**, **TESLA**, **AMAZON** e **MAGALU**.

Código *Python* para finanças

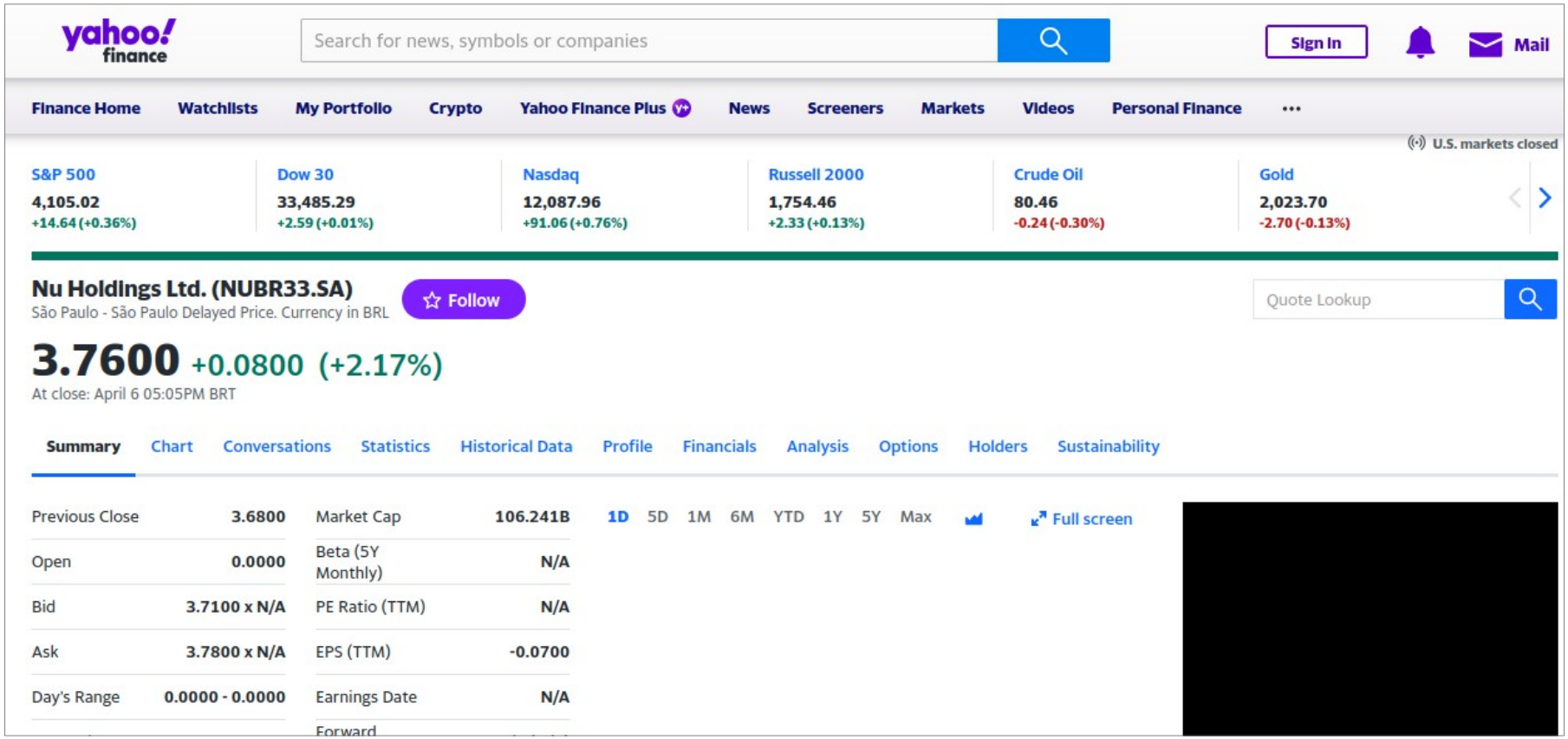

Figura: Código *Python* para finanças – *Yahoo Finance*.

Código *Python* para finanças

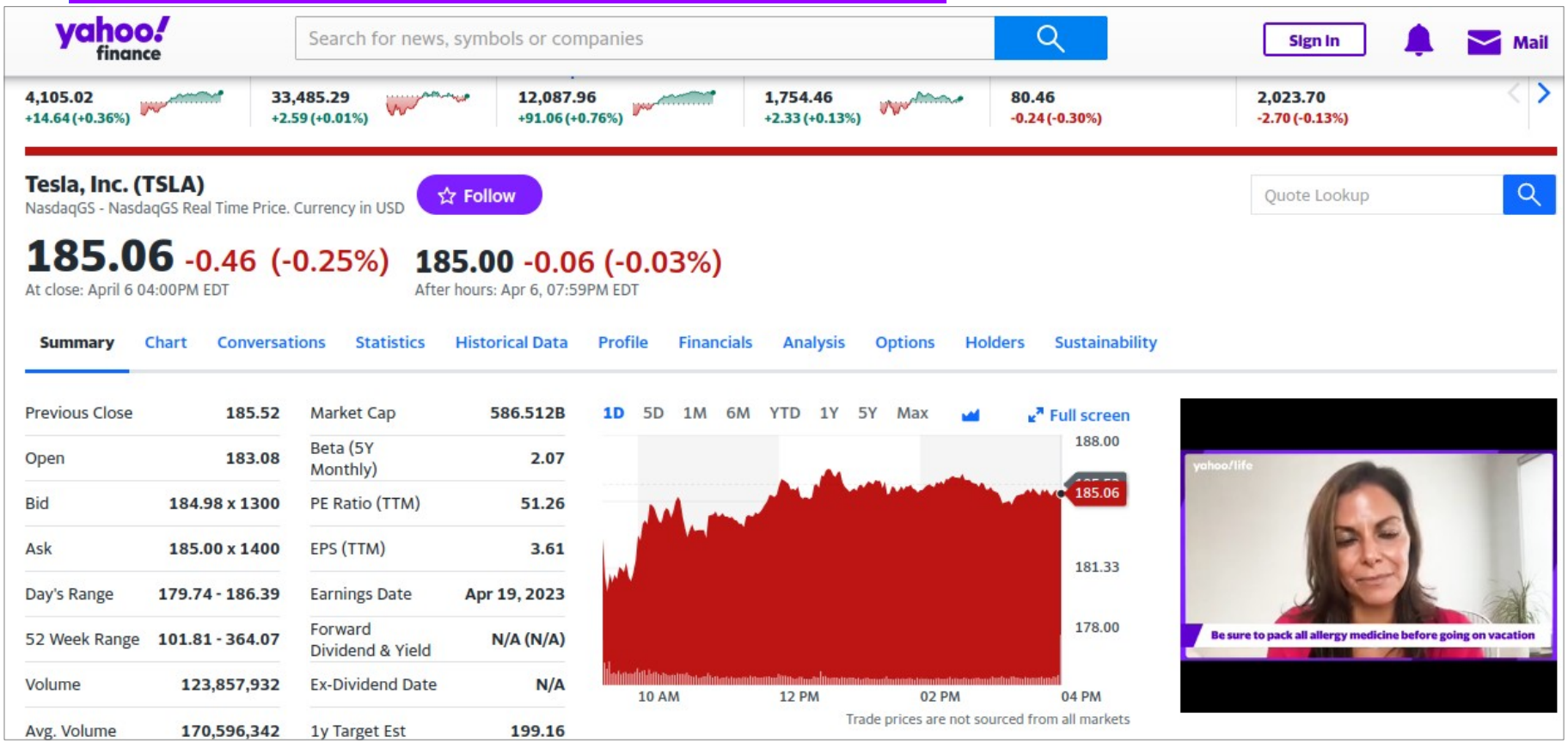

Figura: Código *Python* para finanças – *Yahoo Finance*.

Código *Python* para finanças

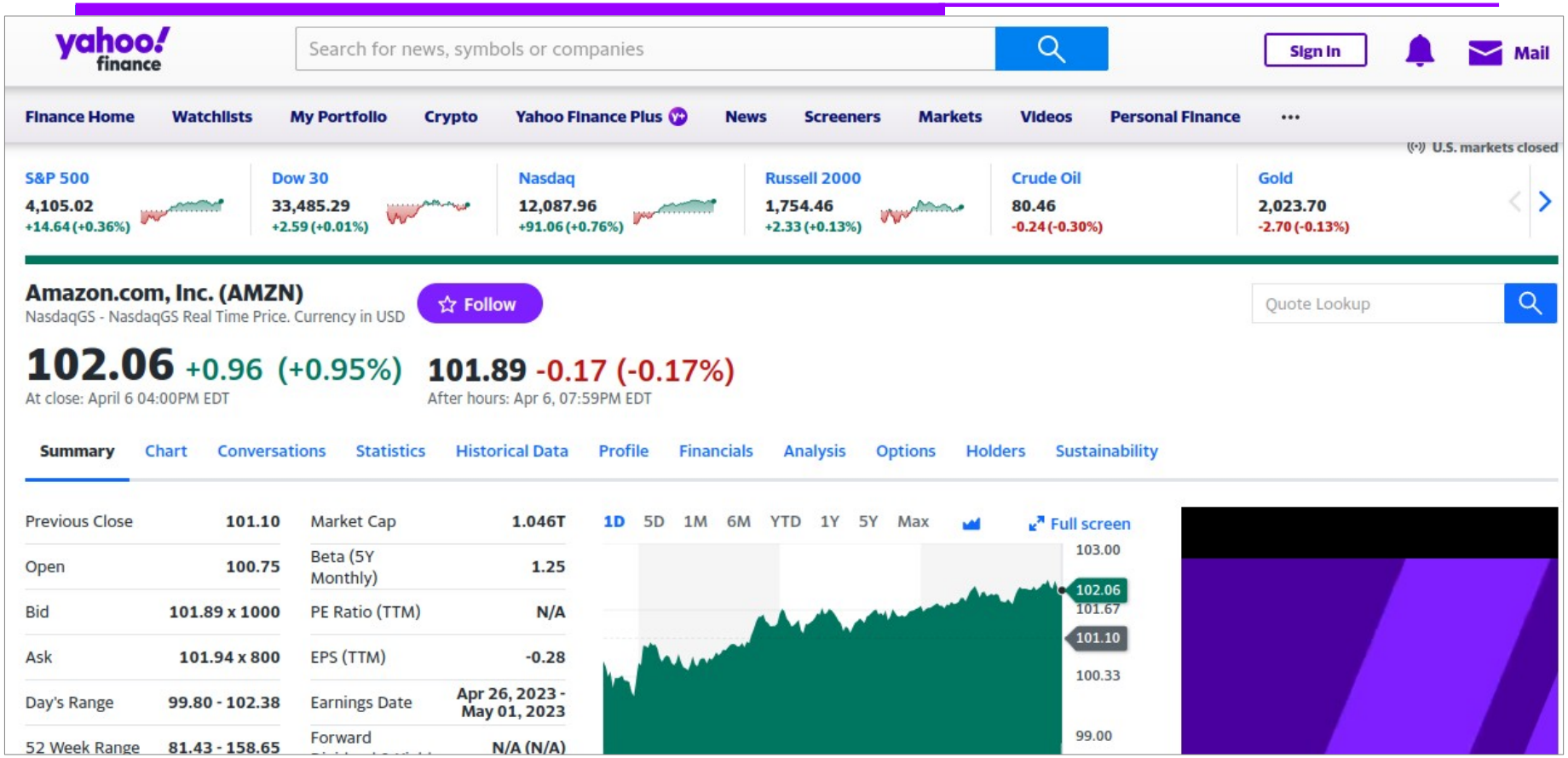

Figura: Código *Python* para finanças – *Yahoo Finance*.

Código *Python* para finanças

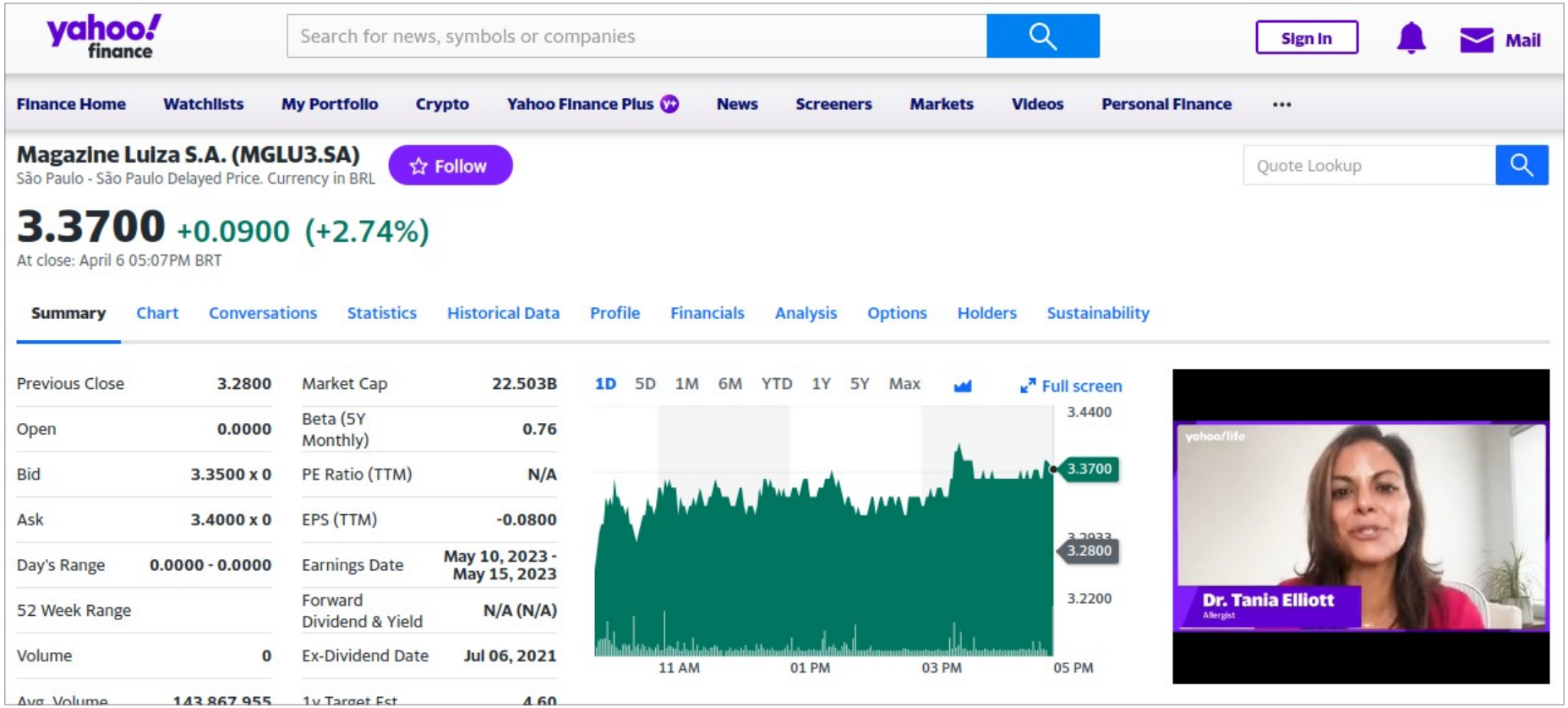

Figura: Código *Python* para finanças – *Yahoo Finance*.

Código *Python* para finanças

Usaremos o ***Google Colab***, uma plataforma de desenvolvimento **em nuvem** que permite aos usuários **escrever, executar e compartilhar código em *Python***.

- **Perfomance**: acesso gratuito (ou planos pagos) a **GPUs e TPUs** para processamento acelerado de dados.
- **Armazenamento e uso**: os arquivos **ficam em nuvem** vinculados ao ***Google Drive***. É possível instalar bibliotecas durante as sessões de uso via ***Jupyter Notebooks***.
- **Estudo de caso**: como acessar fontes de dados financeiros do ***Yahoo Finance*** e visualizar as ações do **NUBANK**, **TESLA**, **AMAZON** e **MAGALU**.
- **Material utilizado**: os códigos e arquivos do projeto estão disponíveis no repositório **"flisol-2023"** da conta **APUS Digital** no *GitHub*.

Código *Python* para finanças

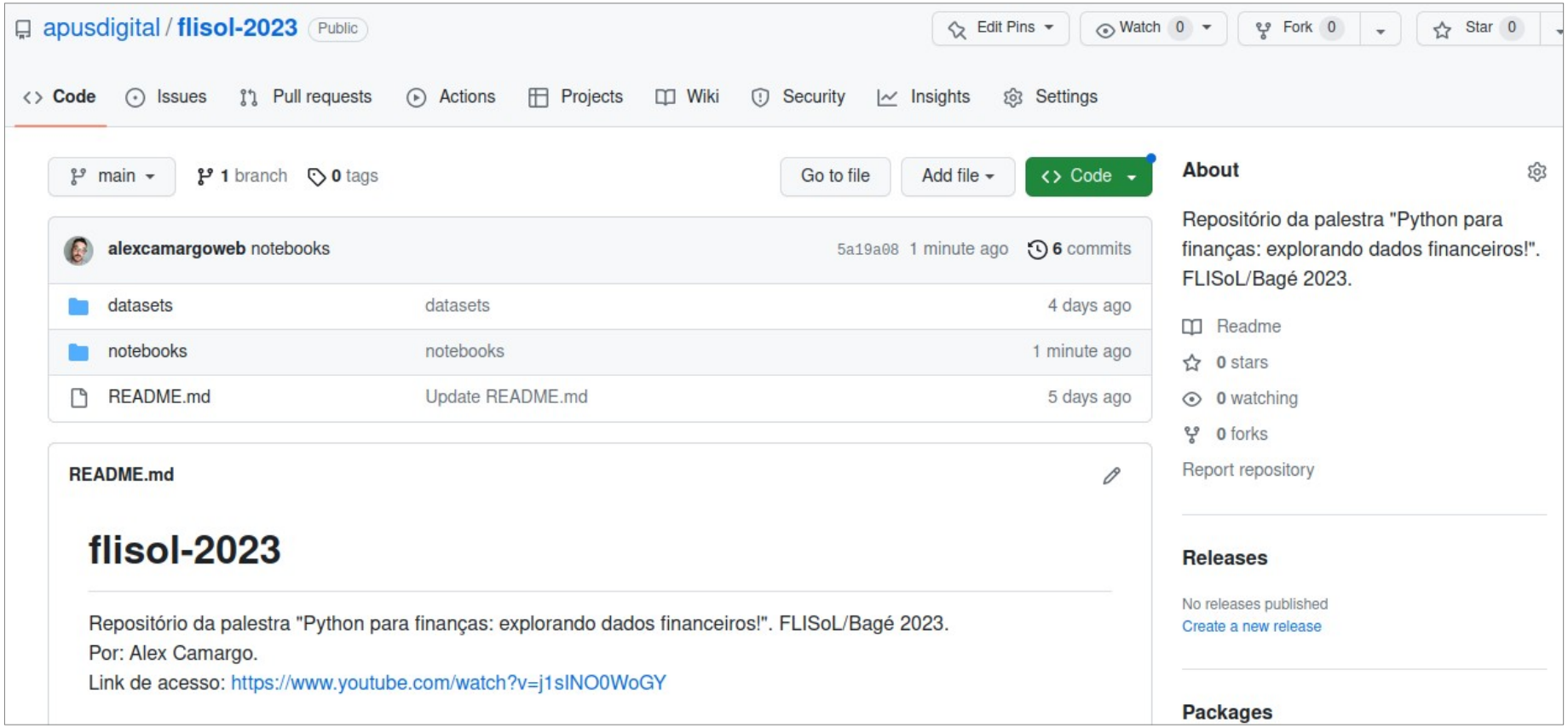

Figura: Código *Python* para finanças – Repositório no *GitHub*.

Considerações finais

É crucial que as pessoas **compreendam o que estão analisando** e como as **ferramentas e técnicas de programação podem ser aplicadas** de forma eficaz para tomar decisões.

- *Python* é uma das **linguagens de programação mais populares** no mundo da finança.
- Bibliotecas específicas para finanças, como ***yfinance***, ***pyfolio***, ***QuantLib*** e outras, tornam a análise financeira **mais eficiente e produtiva**.
- ***Google Colab*** é uma ferramenta poderosa para usar o *Python* em alta perfomance através de **GPUs** e **TPUs** “gratuitas”.
- ***Python* é apenas uma ferramenta! Uma análise financeira requer conhecimentos financeiros sólidos e ajuda profissional.**

Principais referências

B3: A Bolsa do Brasil
https://www.b3.com.br/pt_br/

CVM Comissão de Valores Mobiliários - Governo Federal
https://www.gov.br/cvm/pt-br

Introducing *ChatGPT*
https://openai.com/blog/chatgpt

Rico Investimentos - Plataforma de Investimento
https://www.rico.com.vc/

Site Oficial do Tesouro Direto
https://www.tesourodireto.com.br/

Welcome To Colaboratory - Google
https://colab.research.google.com/

Welcome to Python.org
https://www.python.org/