

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](mailto:alex@apus.digital)



Abril 15, 2023

# About me

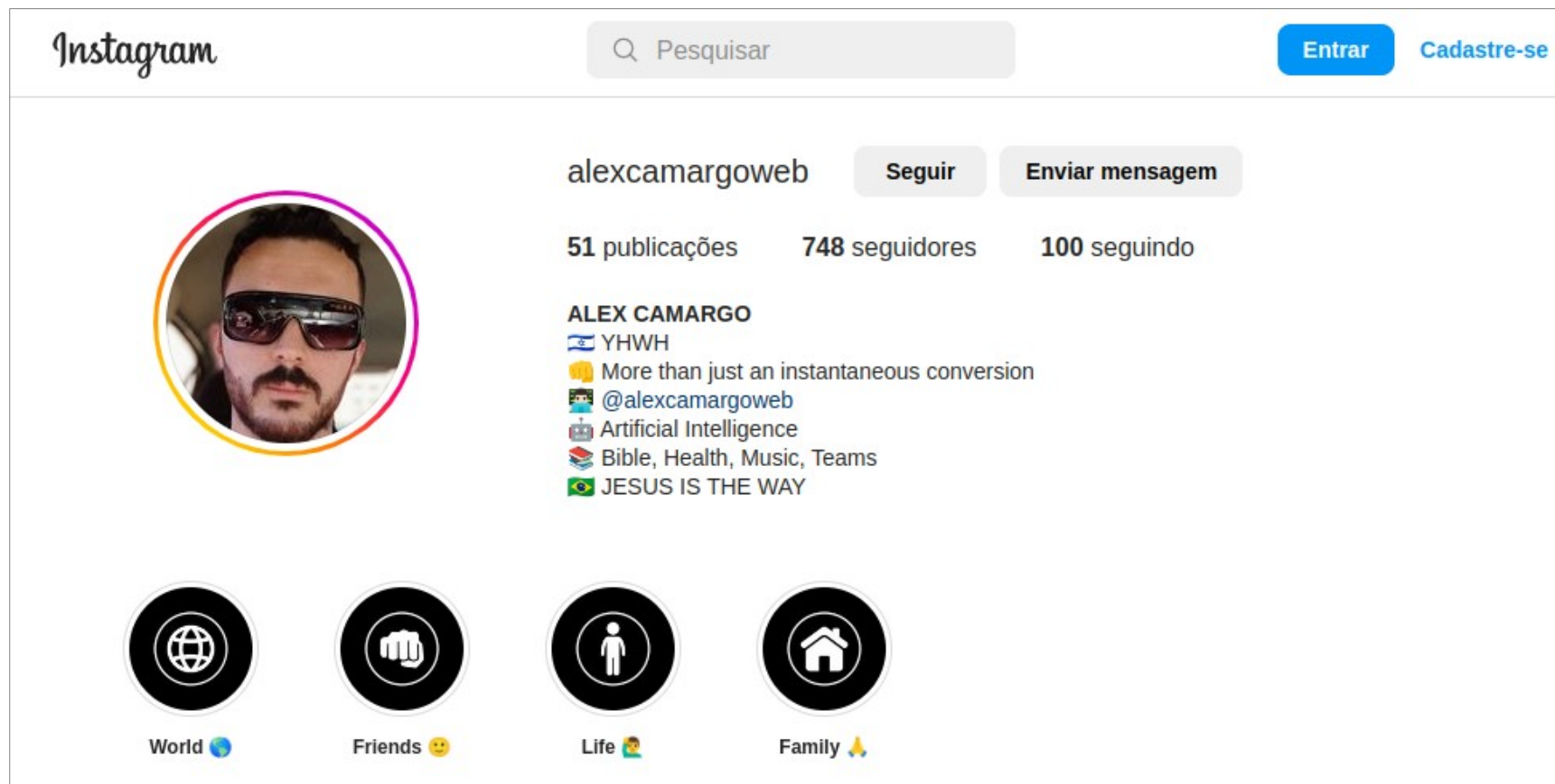
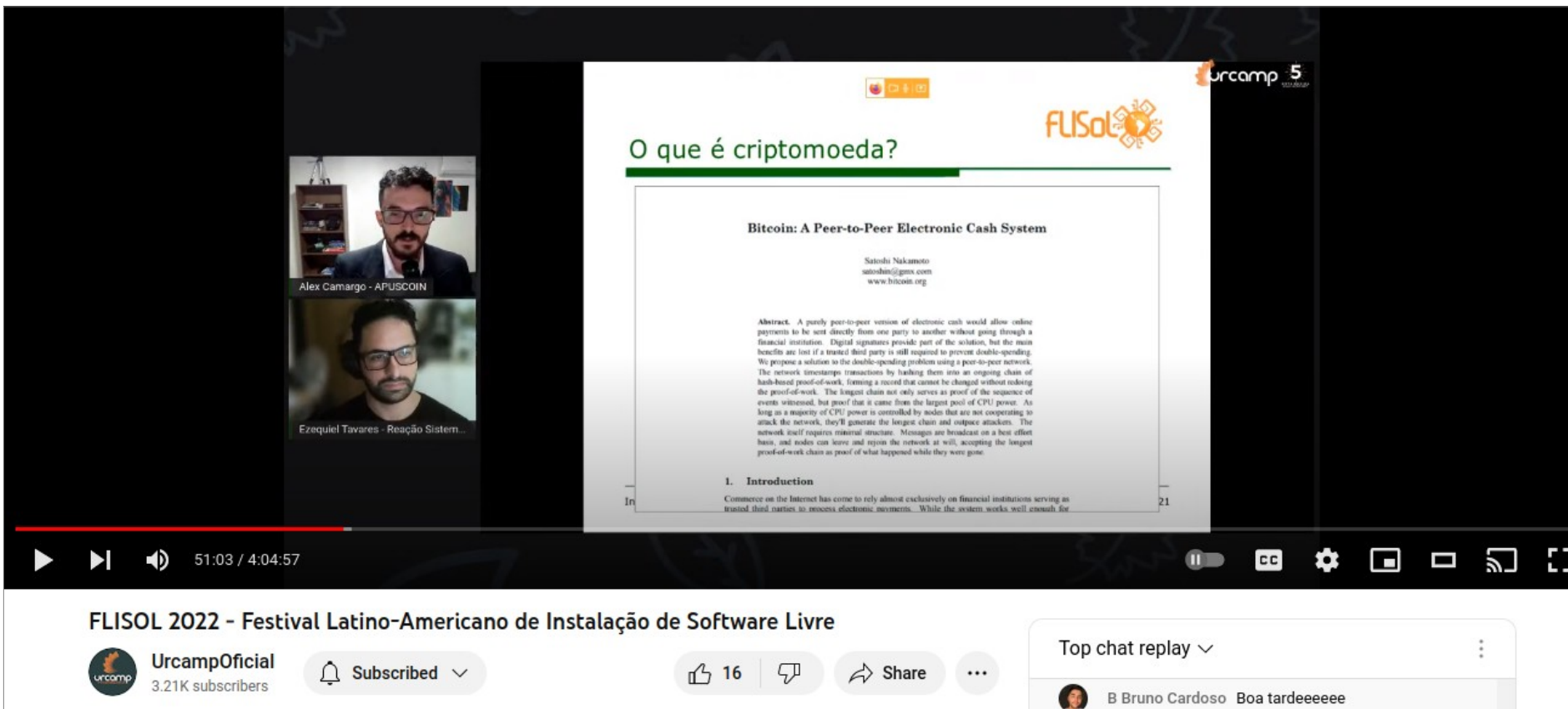


Figure: Instagram @alexcamargoweb.

# Edições anteriores



**O que é criptomoeda?**

**FLISOL**

**Bitcoin: A Peer-to-Peer Electronic Cash System**

Satoshi Nakamoto  
satoshi@gmx.com  
www.bitcoin.org

**Abstract.** A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

**1. Introduction**

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for

51:03 / 4:04:57

**FLISOL 2022 - Festival Latino-Americano de Instalação de Software Livre**

**UrcampOficial**  
3.21K subscribers

Subscribed

16

Share

Top chat replay

B Bruno Cardoso Boa tardeeeee

Figure: FLISOL 2022.



APUS Digital

Inovação em primeiro lugar.

# 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**.

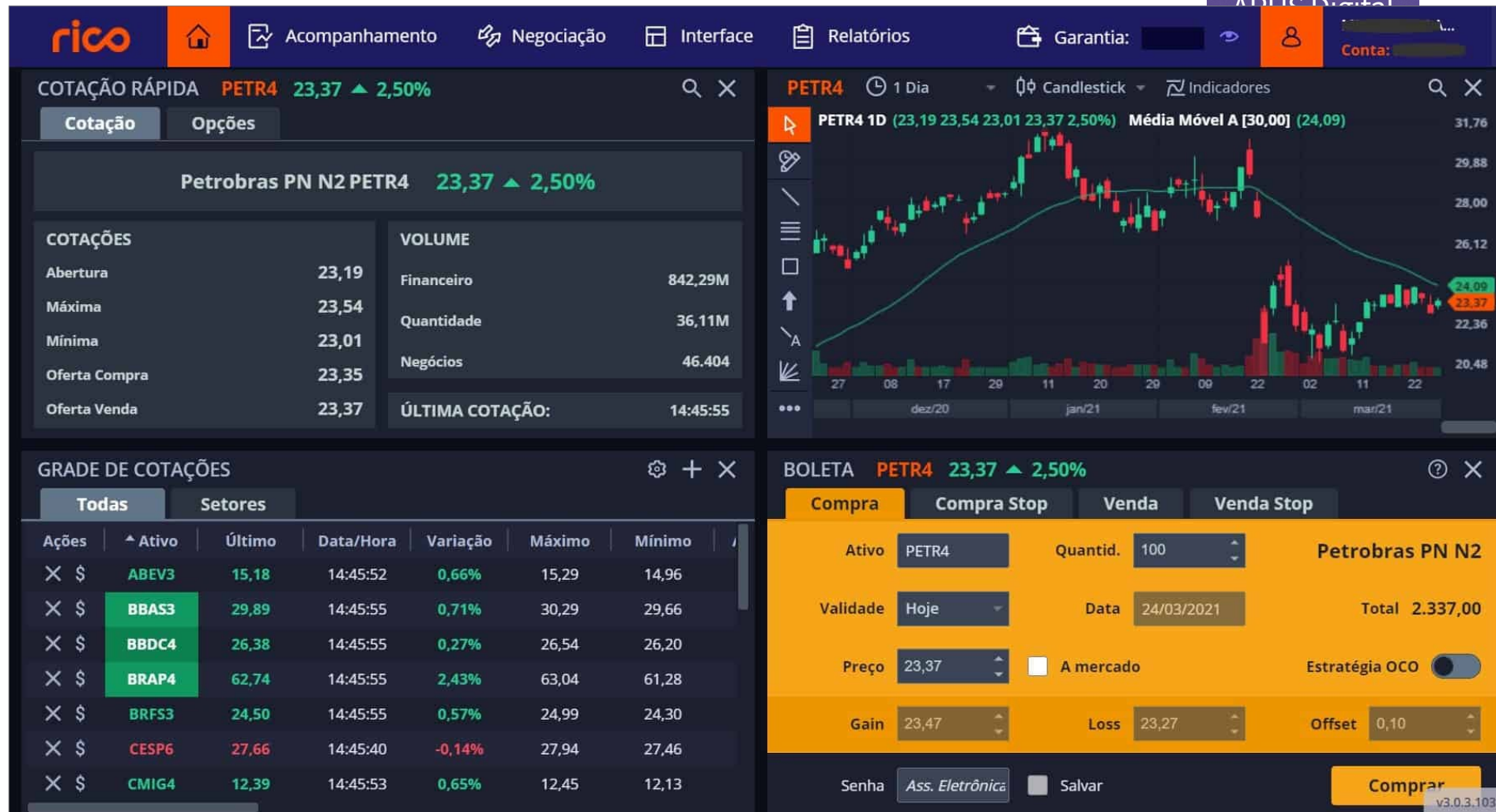


Figura: Tipos de mercado financeiro - Ações.







# Introdução ao mercado financeiro



Figura: Tipos de mercado financeiro – Câmbio.





cal  
gar.



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

## Número de investidores pessoa física

POR ANO DESDE 2002

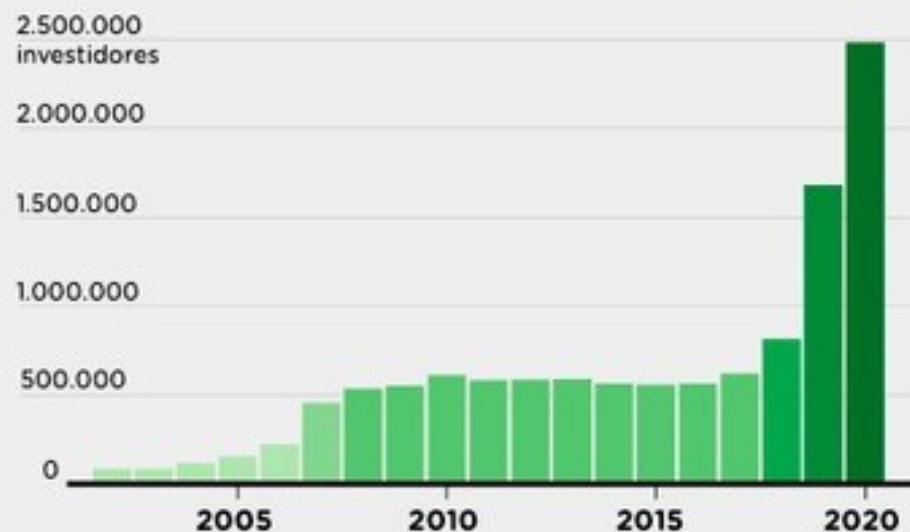


Figura: Crescimento de investidores no Brasil.



# 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**.



Portal do Bitcoin



CVM proíbe corretora forex de captar clientes no Brasil | Portal do Bitcoin

[Visit](#)

Images may be subject to copyright. [Learn More](#)

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).

# AS PRINCIPAIS EMPRESAS DE CADA SETOR NA COMPOSIÇÃO DO IBOVESPA

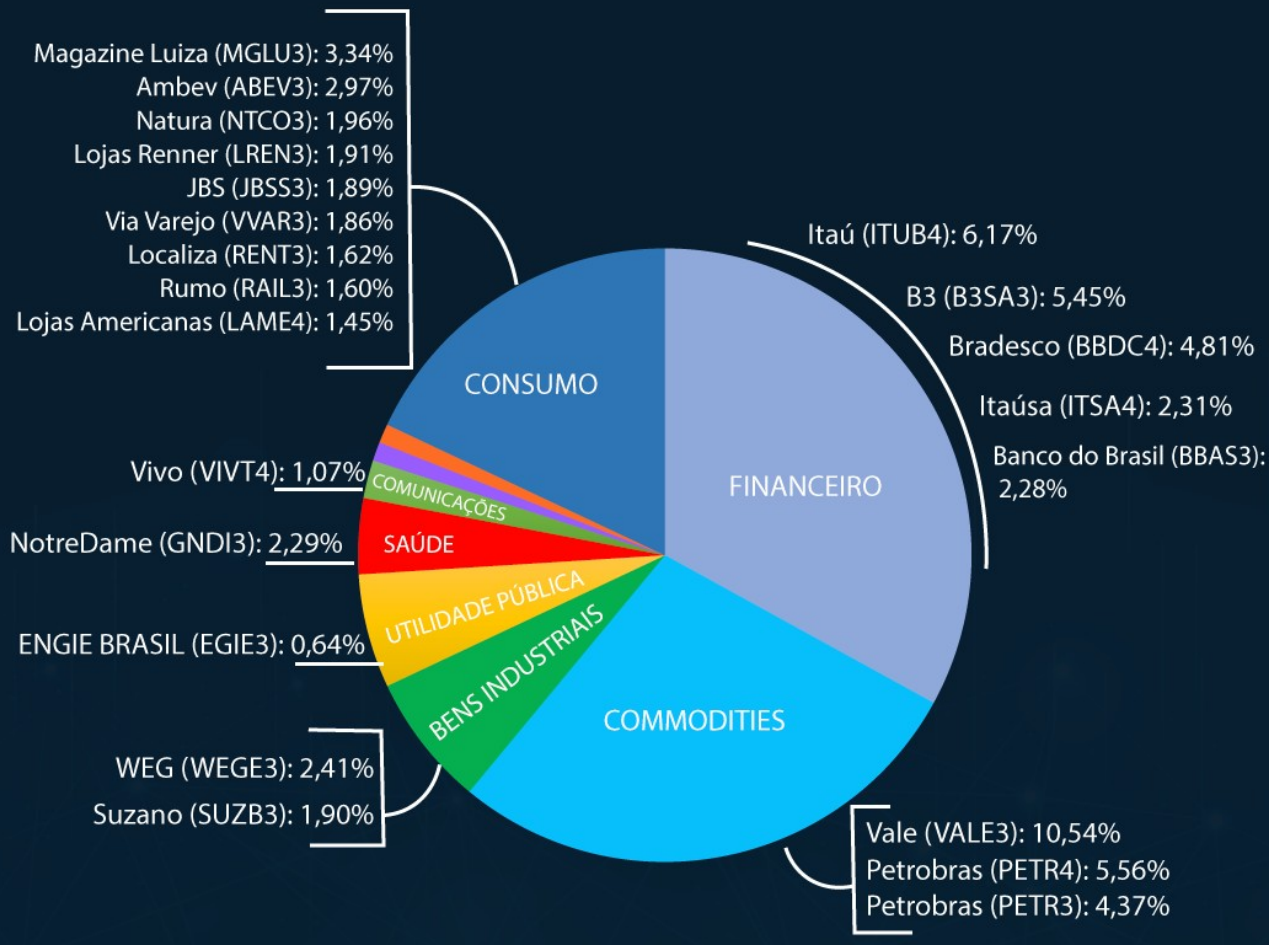


Figura: Principais empresas de cada setor na composição do índice IBOVESPA.



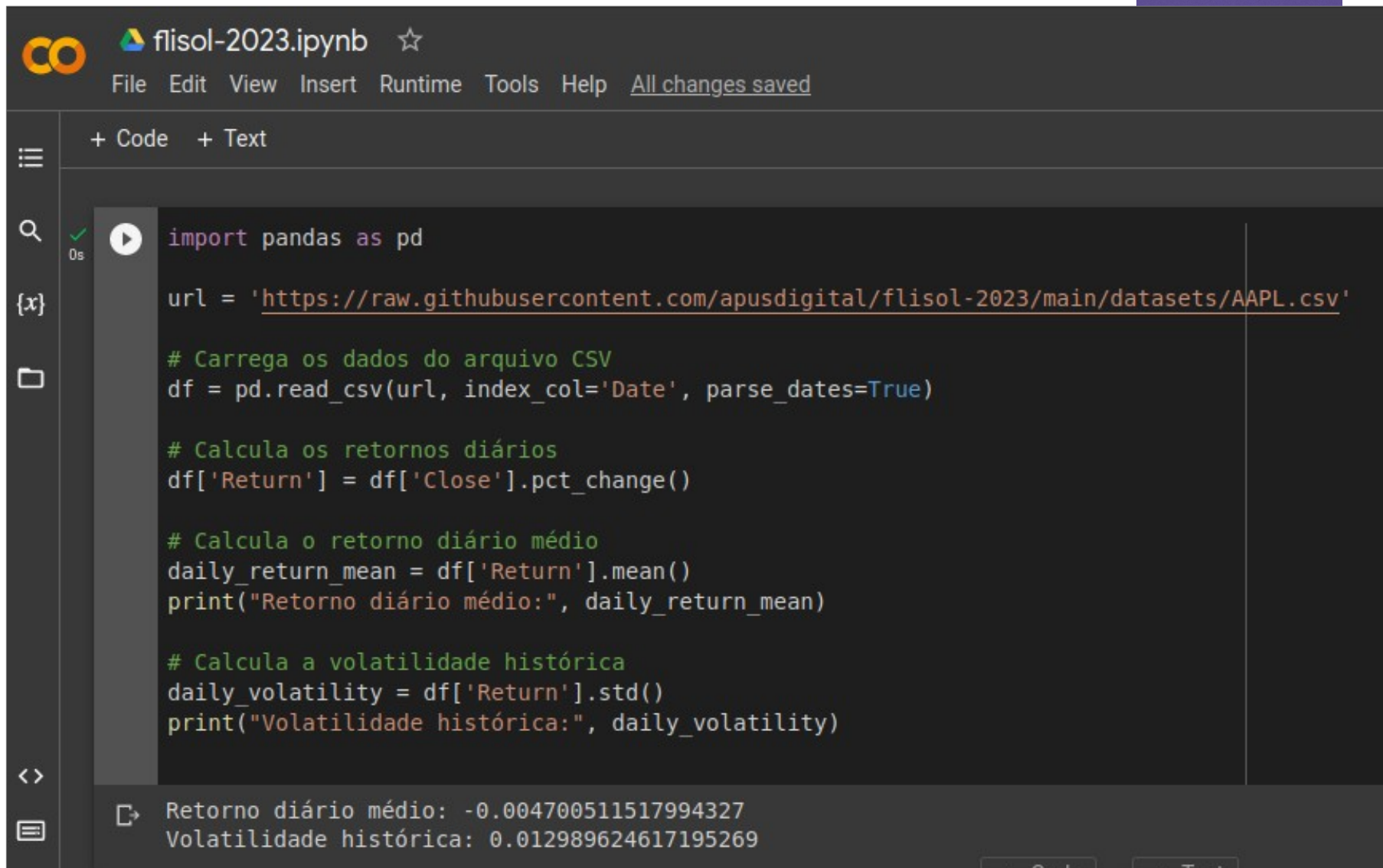


# Python na área financeira

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.





The image shows a Jupyter Notebook interface with a dark theme. The top bar includes the Colab logo, the filename 'flisol-2023.ipynb', and a star icon. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', 'Help', and 'All changes saved'. The left sidebar contains icons for file explorer, search, and other notebook functions. The main area displays a code cell with the following Python code:

```
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)
```

Below the code cell, the output is displayed:

```
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

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*.

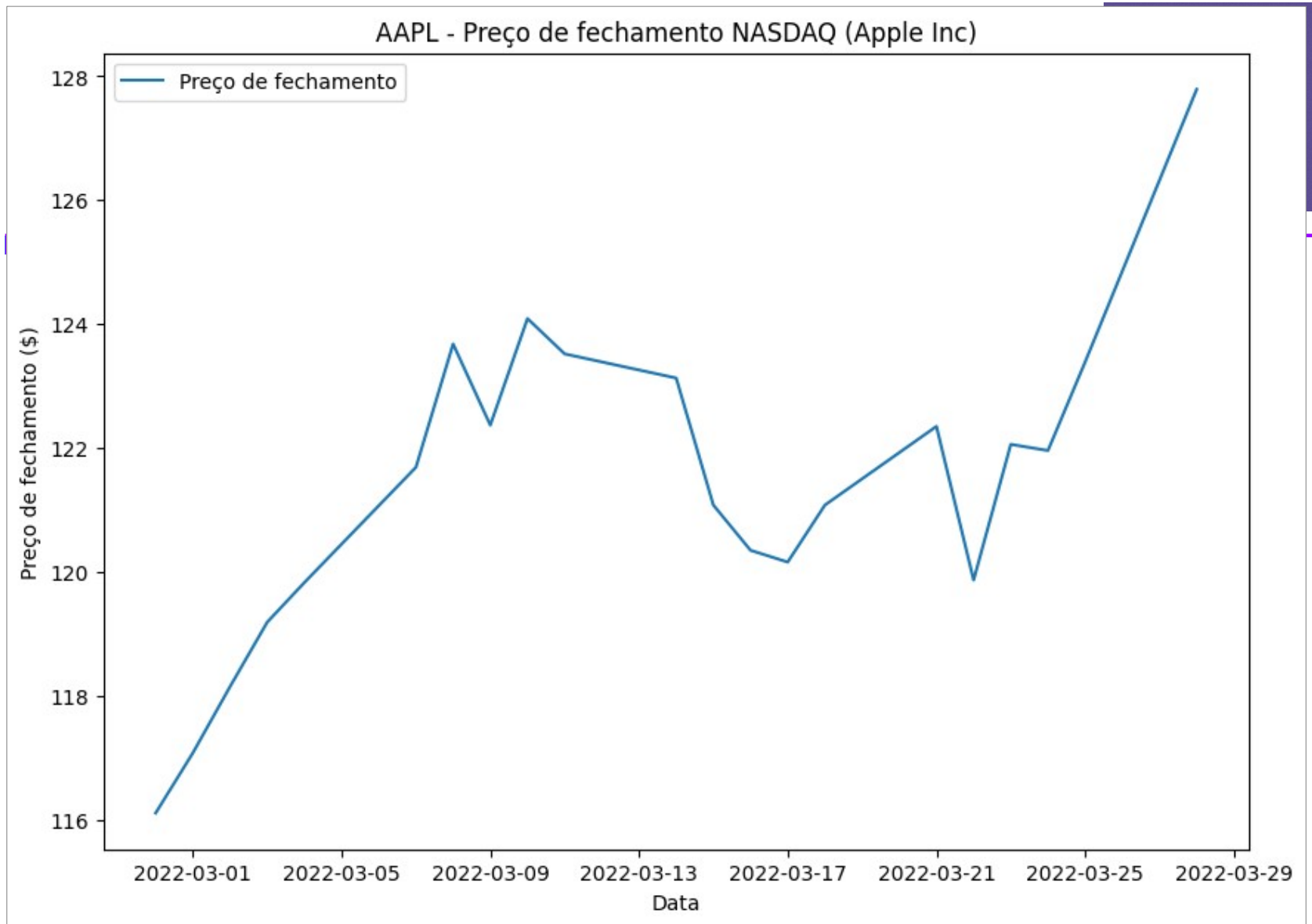
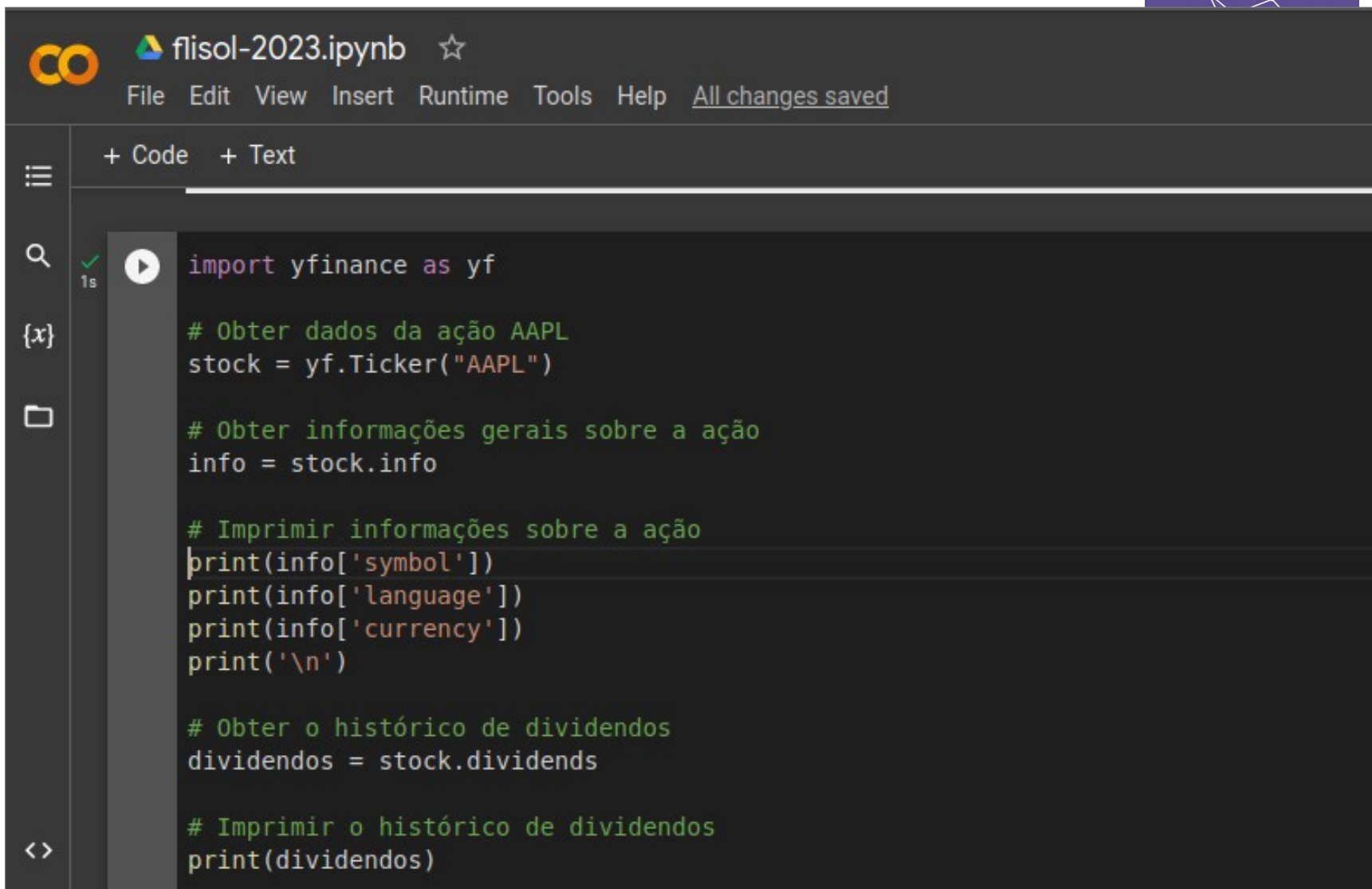


Figura: Bibliotecas *Python* usadas na área financeira - *matplotlib*.



The image shows a Jupyter Notebook window titled "flisol-2023.ipynb". The interface includes a top menu bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help", along with a status bar indicating "All changes saved". On the left, there is a sidebar with icons for file management and a search bar. The main area displays a code cell with the following Python code:

```
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*.

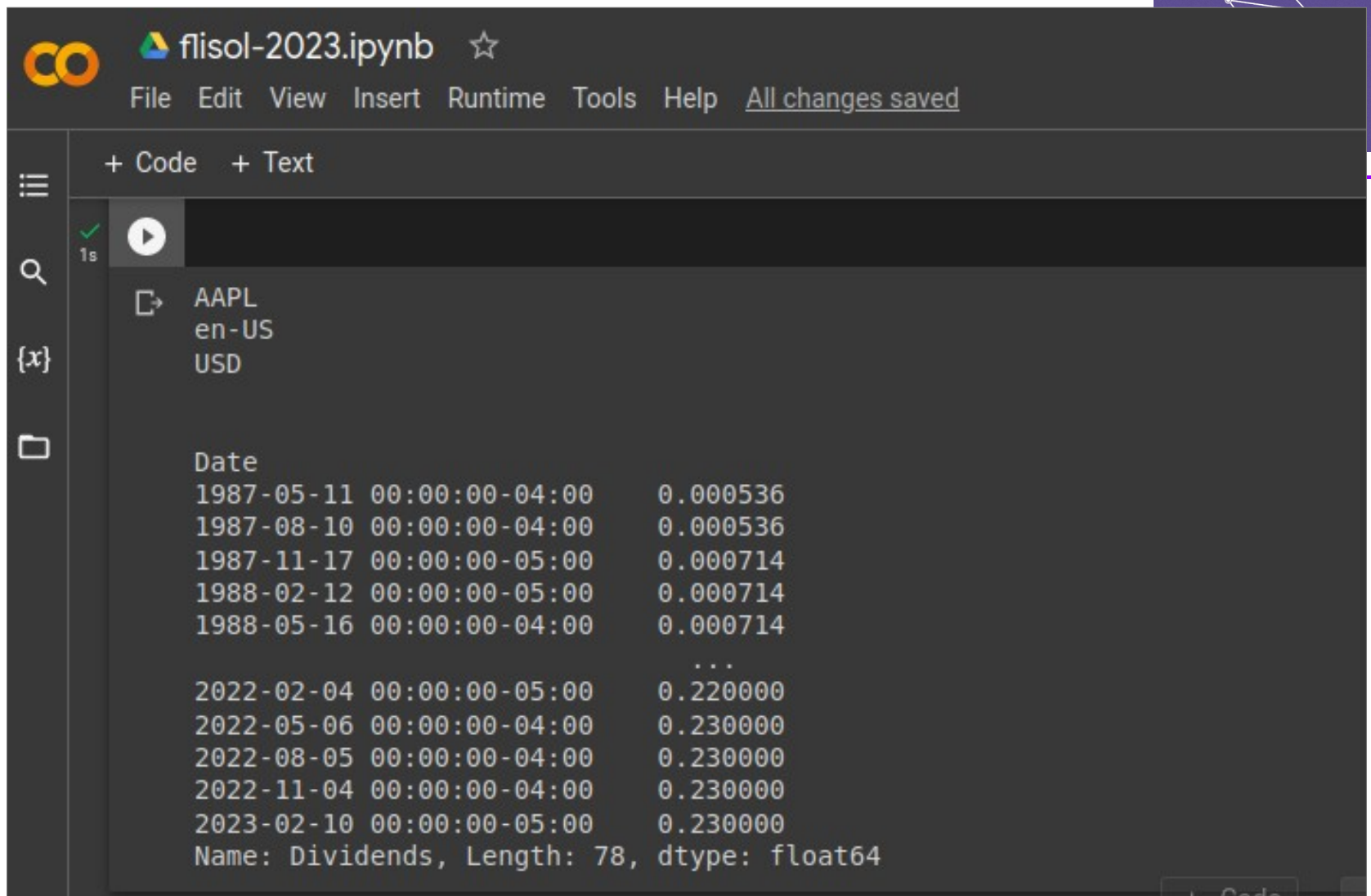
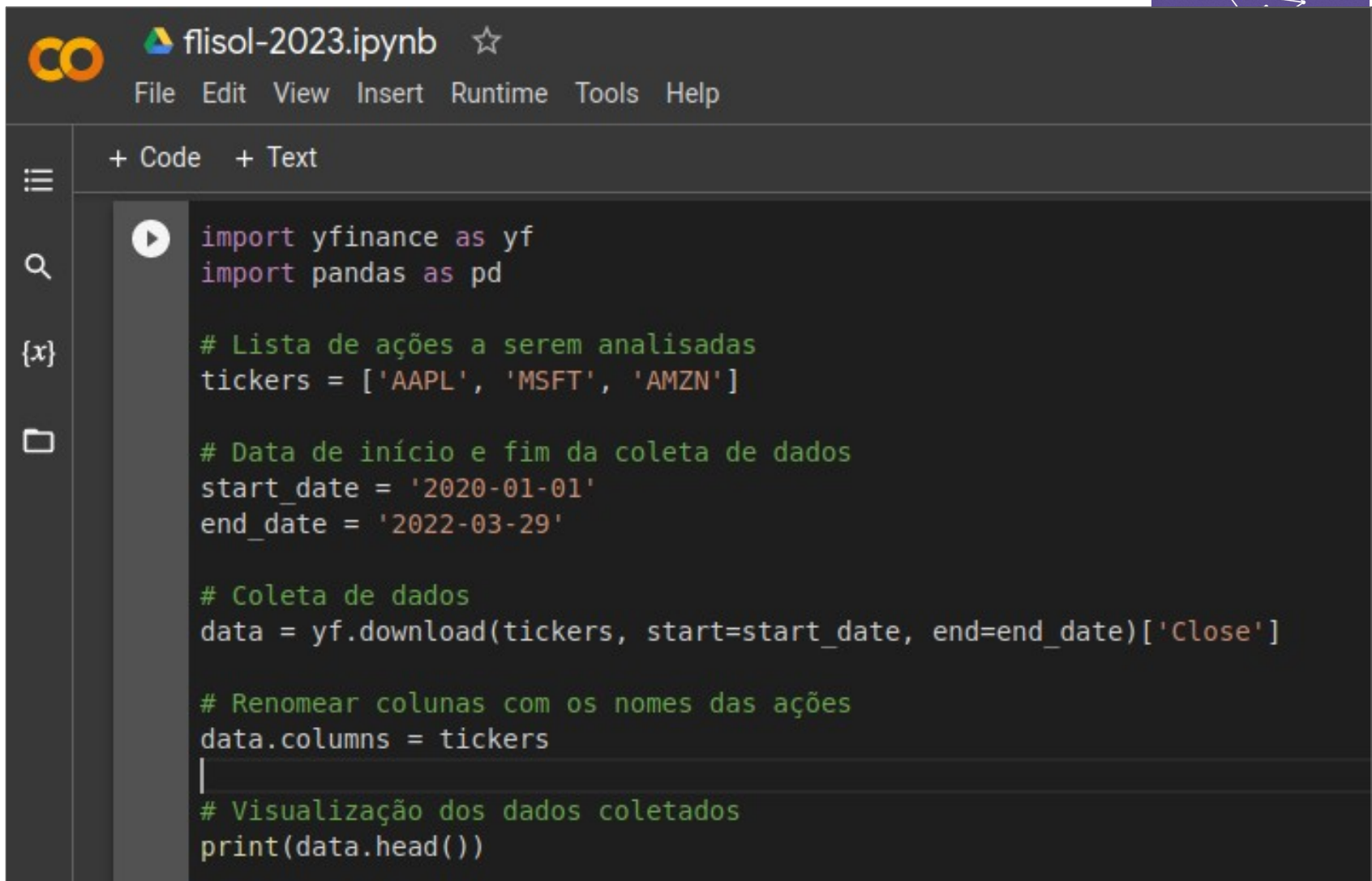


Figura: Bibliotecas *Python* usadas na área financeira - *yfinance*.





The image shows a Jupyter Notebook interface with a dark theme. The top bar displays the Colab logo, the filename 'flisol-2023.ipynb', and a star icon. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. On the left side, there is a sidebar with icons for a menu, search, variables, and a file explorer. The main area contains a code cell with the following Python code:

```
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

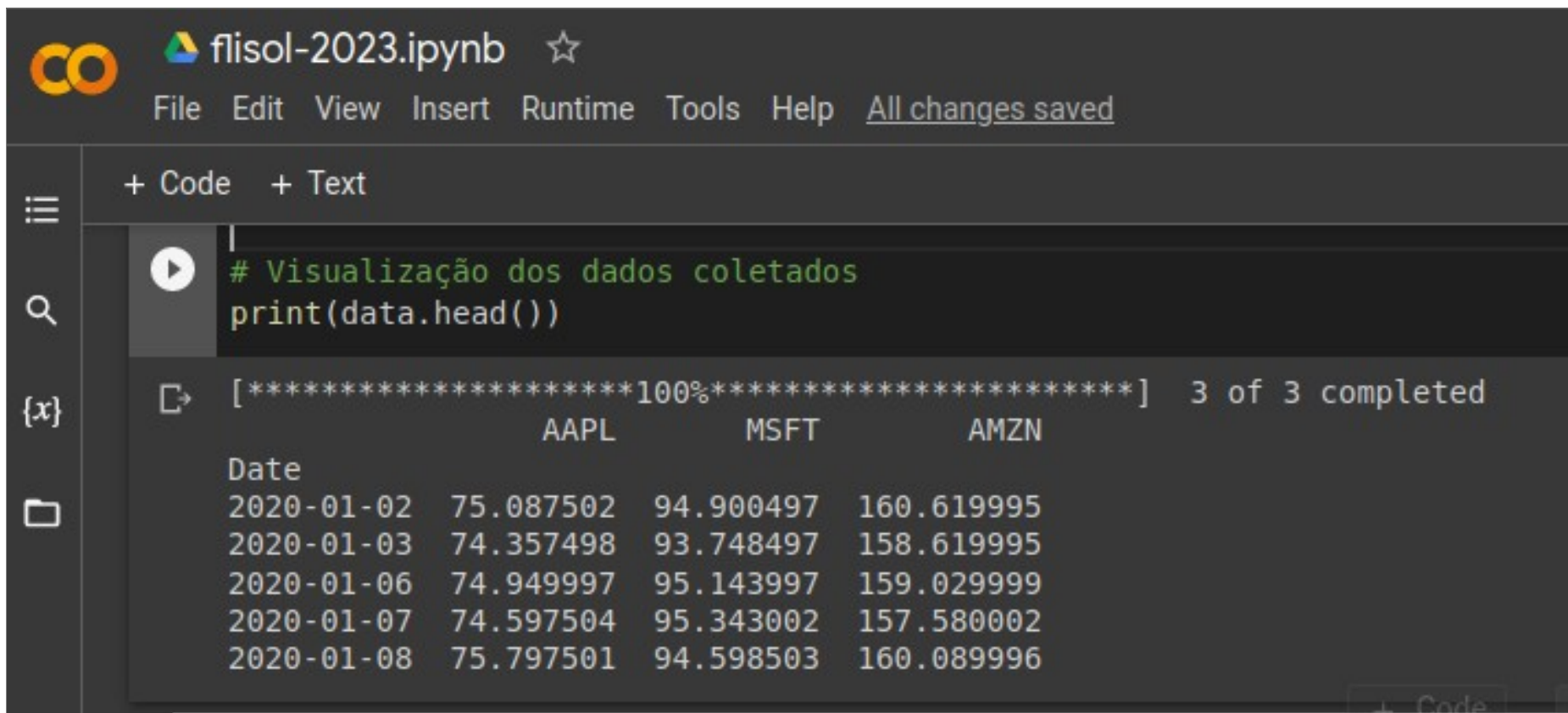


Figura: Coleta de dados - *yfinance*.



ADUC Digital

The screenshot shows a Jupyter Notebook titled 'flisol-2023.ipynb'. The interface includes a top menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help', along with a status 'All changes saved'. On the left, there is a sidebar with icons for file management and a search icon. The main area displays a code cell with the following Python code:

```
# 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 preço atual da ação
price = price_element.text
print('Preço da ação da Apple (AAPL):', price)
```

Below the code cell, the output is displayed:

```
<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*.

## Apple Inc. (AAPL)

NasdaqGS - NasdaqGS Real Time Price. Currency in USD

☆ Follow

**163.12** -0.64 (-0.39%)

As of 10:18AM EDT. Market open.

Summary

Chart

Conversations

Statistics

Historical Data

Profile

Financials

Analysis

Options

Holders

Sustain

Previous Close	163.76	Market Cap	2.599T
Open	162.43	Beta (5Y Monthly)	1.30
Bid	162.28 x 800	PE Ratio (TTM)	27.53
Ask	162.28 x 900	EPS (TTM)	5.90
Day's Range	162.03 - 162.73	Earnings Date	Apr 26, 2023 - May 01, 2023
52 Week Range	124.17 - 176.15	Forward Dividend & Yield	0.92 (0.56%)
Volume	8 962 594	Ex-Dividend Date	Feb 10, 2023

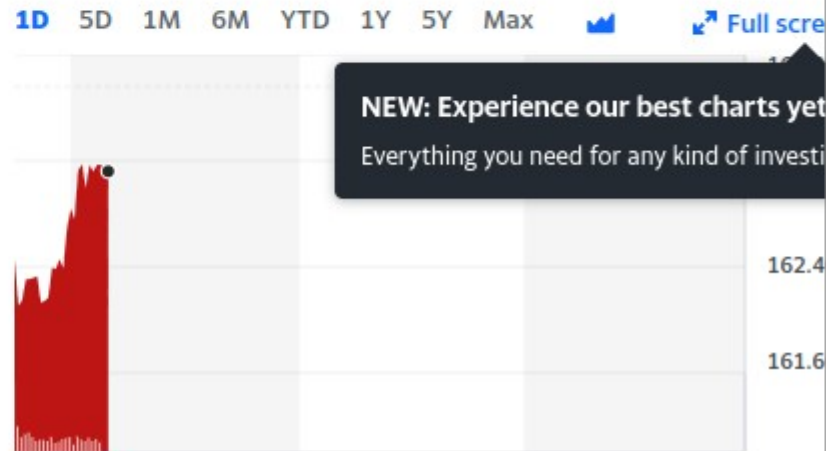


Figura: Coleta de dados - *BeautifulSoup*.

The screenshot shows the NasdaqGS real-time price for Apple Inc. (AAPL) at 10:19AM EDT. The current price is 162.97, down 0.79 (-0.48%) from the previous close of 163.76. The market cap is 2.599T. The browser's developer tool is open, displaying the HTML structure of the price display. The selected element is a `<span class="e3b14781 e59c8479">` tag, which contains the price 162.97. The HTML structure is as follows:

```
<div class="D(ib) Mend(20px)">
  <fin-streamer class="Fw(b) Fz(36px) Mb(-4px) D(ib)"
    symbol="AAPL" data-test="qsp-price" data-field="regularMarketPrice" data-trend="none" data-pr
    value="162.425" active=""> custom...
  <span class="e3b14781 e59c8479">
    ::before
    162.97
    ::after
  </span>
</fin-streamer>
```

Figura: Coleta de dados - *BeautifulSoup*.





# Python na área financeira

```
flisol-2023.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

# 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()
```

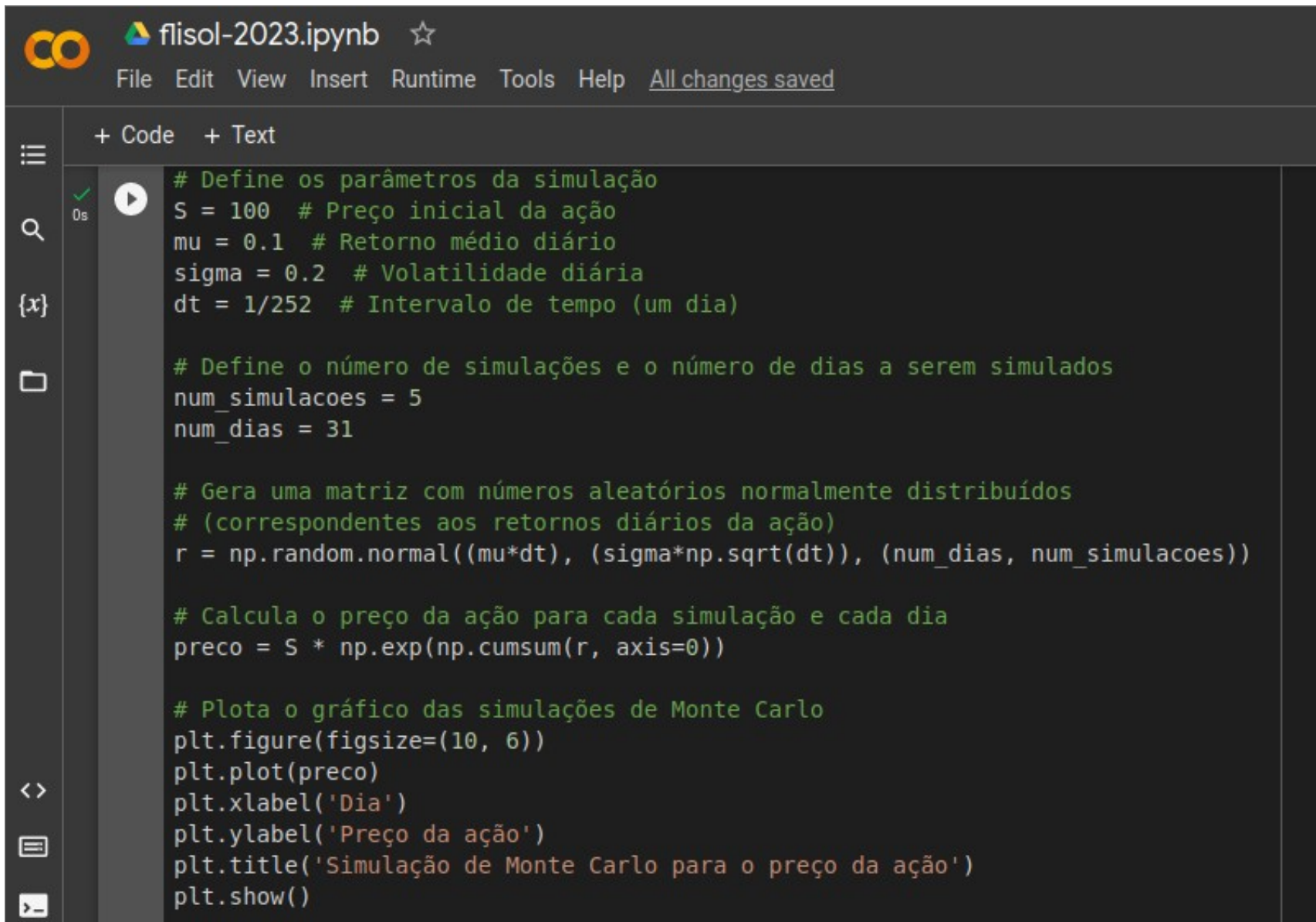
Figura: Visualização de dados - *plotly*.

# Python na área financeira



Figura: Visualização de dados - *plotly*.





The image shows a Jupyter Notebook interface with a dark theme. At the top, there's a header bar with the 'CO' logo, the filename 'flisol-2023.ipynb', and a star icon. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', 'Help', and a link 'All changes saved'. The main area is divided into a left sidebar with icons for file explorer, search, and other functions, and a central code editor. The code editor contains Python code for a Monte Carlo simulation of stock prices. The code defines parameters like initial price (S=100), mean return (mu=0.1), volatility (sigma=0.2), and time interval (dt=1/252). It then simulates 5 scenarios over 31 days, generating random returns and calculating the final stock price for each scenario. Finally, it plots the results using matplotlib, showing the price evolution over time for each simulation.

```
+ Code + Text

# 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*.

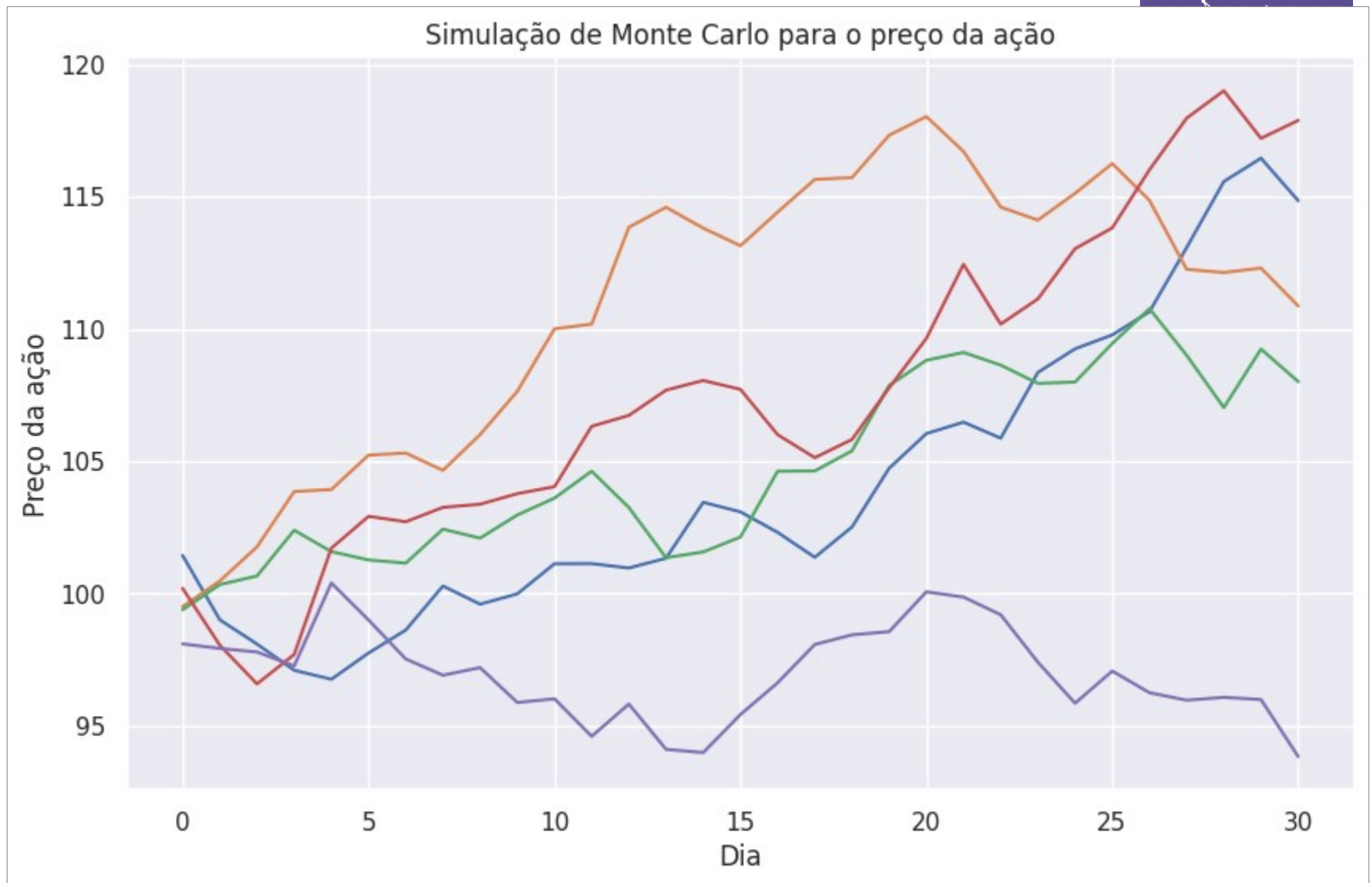


Figura: Modelagem financeira com simulações de Monte Carlo - *matplotlib*.

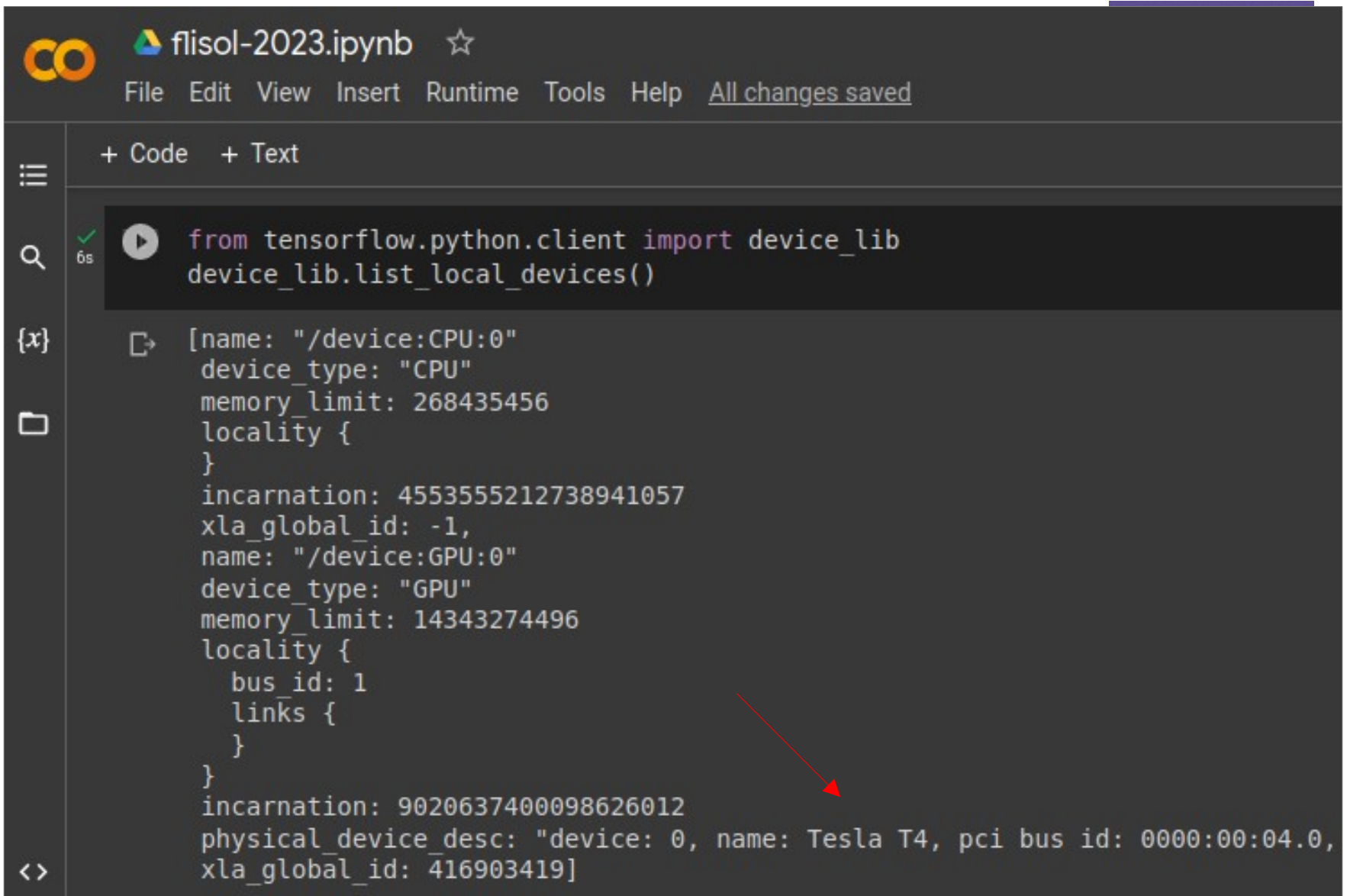


# Código *Python* para finanças

---

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

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



The screenshot shows a Google Colab notebook interface. At the top, the notebook is titled 'flisol-2023.ipynb' with a star icon. Below the title is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', 'Help', and 'All changes saved'. The left sidebar contains icons for file management and a search icon. The main area shows a code cell with the following Python code:

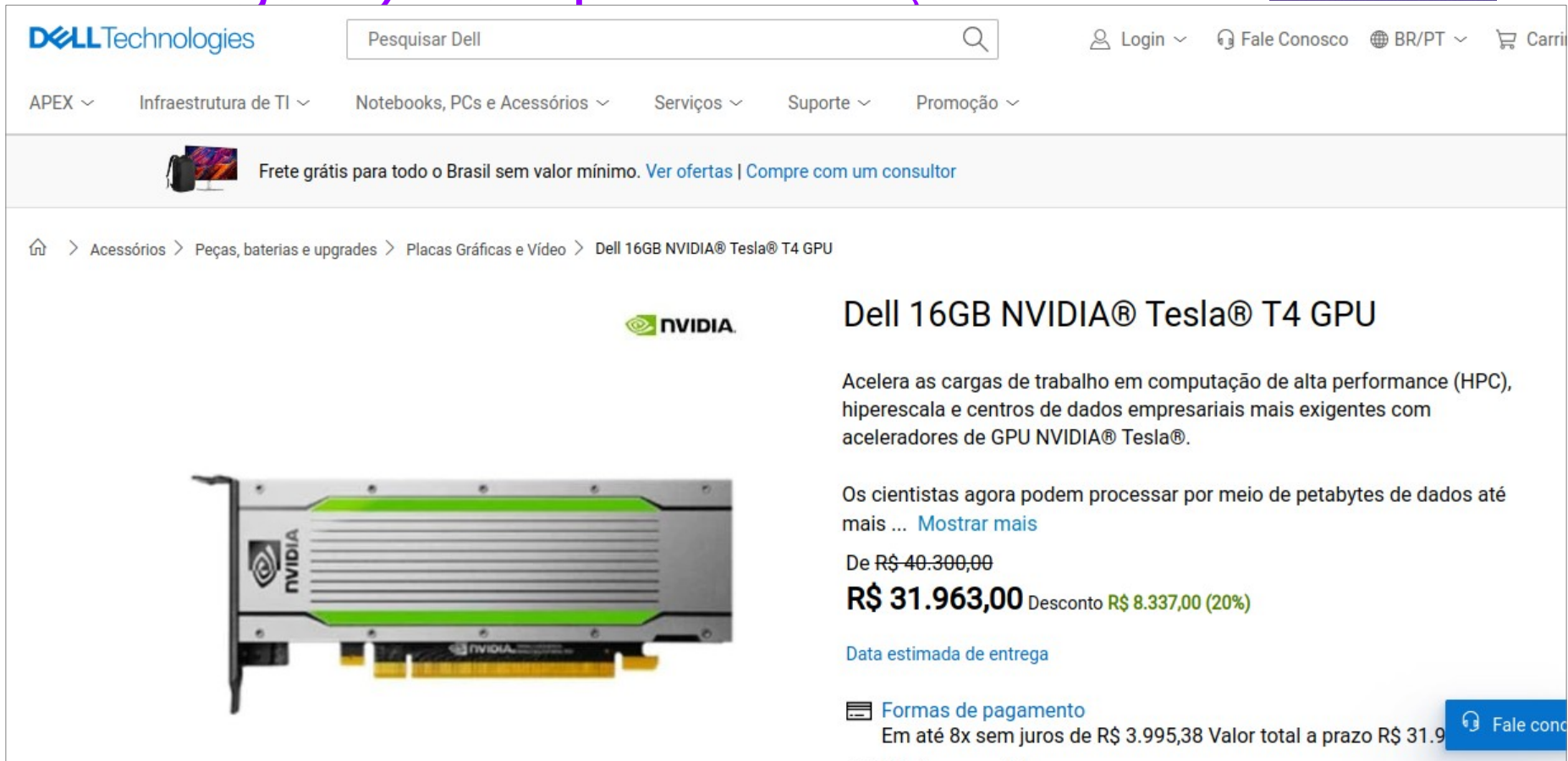
```
from tensorflow.python.client import device_lib
device_lib.list_local_devices()
```

The output of the code is a list of two device dictionaries. A red arrow points to the second dictionary, which represents a GPU:

```
[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



The screenshot shows the Dell Technologies website. At the top, there's a navigation bar with the Dell logo, a search bar, and links for Login, Fale Conosco, BR/PT, and Carrito. Below this is a secondary navigation bar with categories like APEX, Infraestrutura de TI, Notebooks, PCs e Acessórios, Serviços, Suporte, and Promoção. A banner below the navigation bar promotes free shipping for Brazil. The main content area features the Dell 16GB NVIDIA® Tesla® T4 GPU. It includes the NVIDIA logo, a detailed image of the GPU, and a description of its capabilities for high-performance computing (HPC), hyper-scale, and enterprise data centers. The price is listed as R\$ 31.963,00, with a discount of R\$ 8.337,00 (20%) from the original price of R\$ 40.300,00. There are links for 'Mostrar mais', 'Data estimada de entrega', and 'Formas de pagamento'. A blue button for 'Fale conosco' is also visible.

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***.

- **Performance:** 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

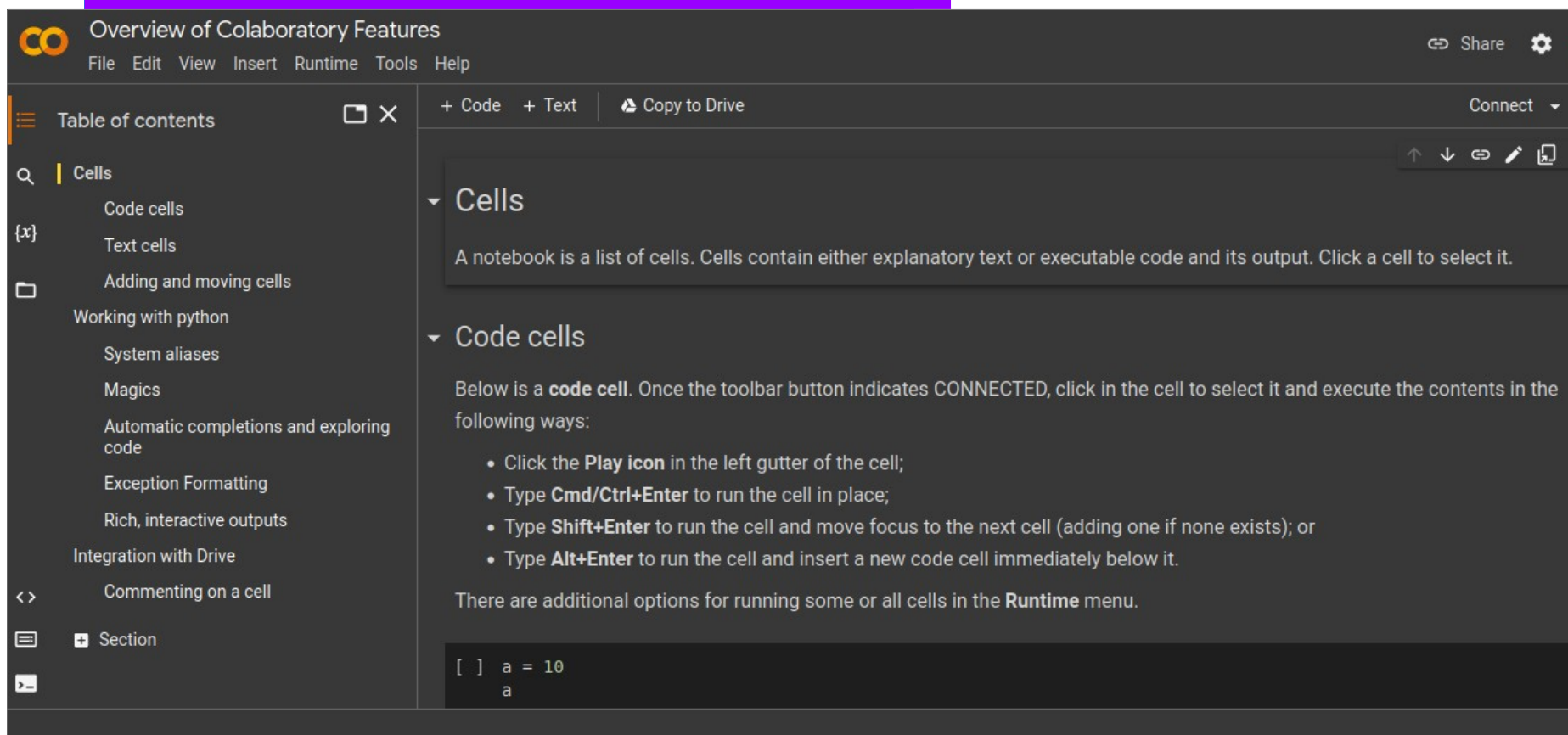


Figura: Código *Python* para finanças – *Jupyter Notebook*.

# 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***.

- **Performance:** 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

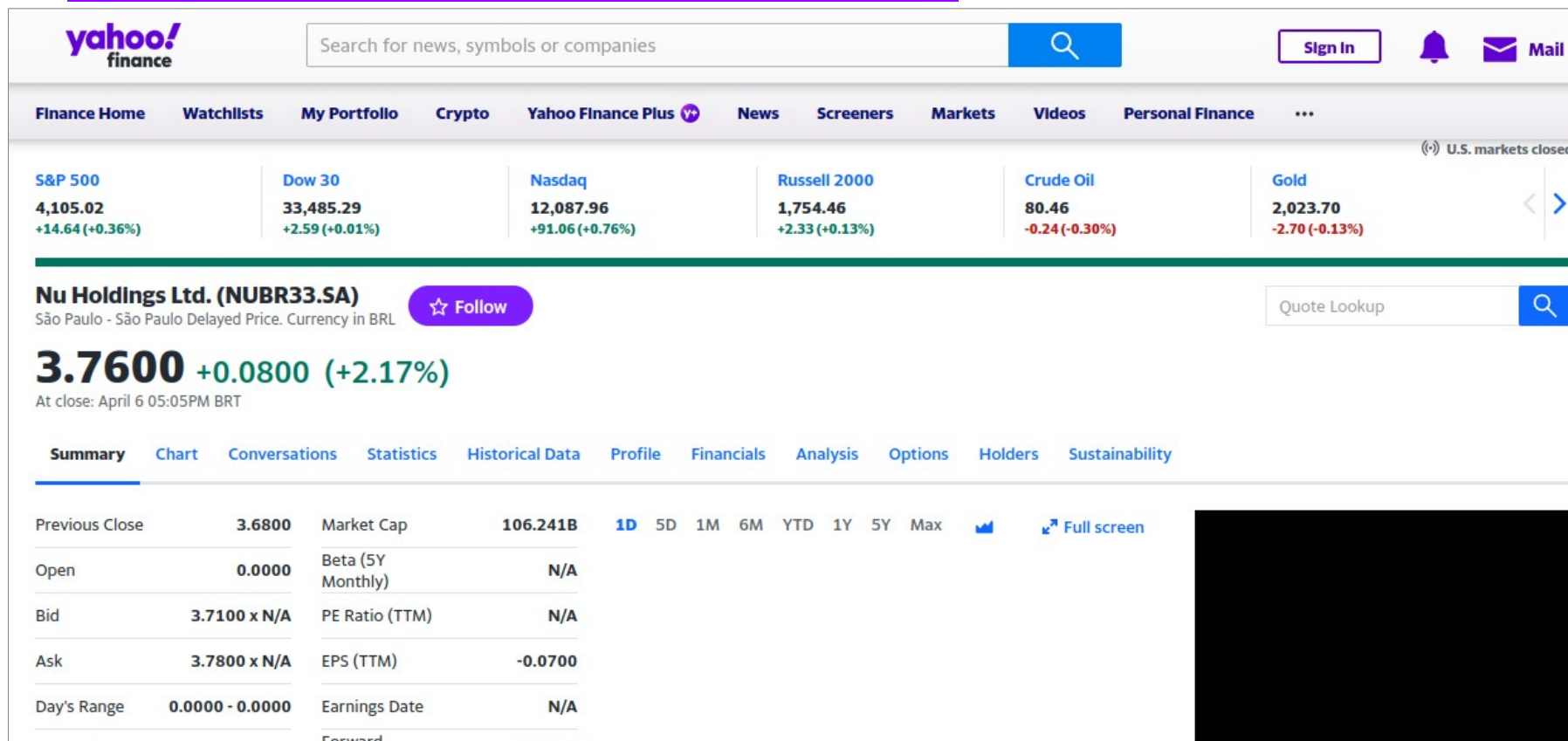


Figura: Código *Python* para finanças – *Yahoo Finance*.

# Código *Python* para finanças

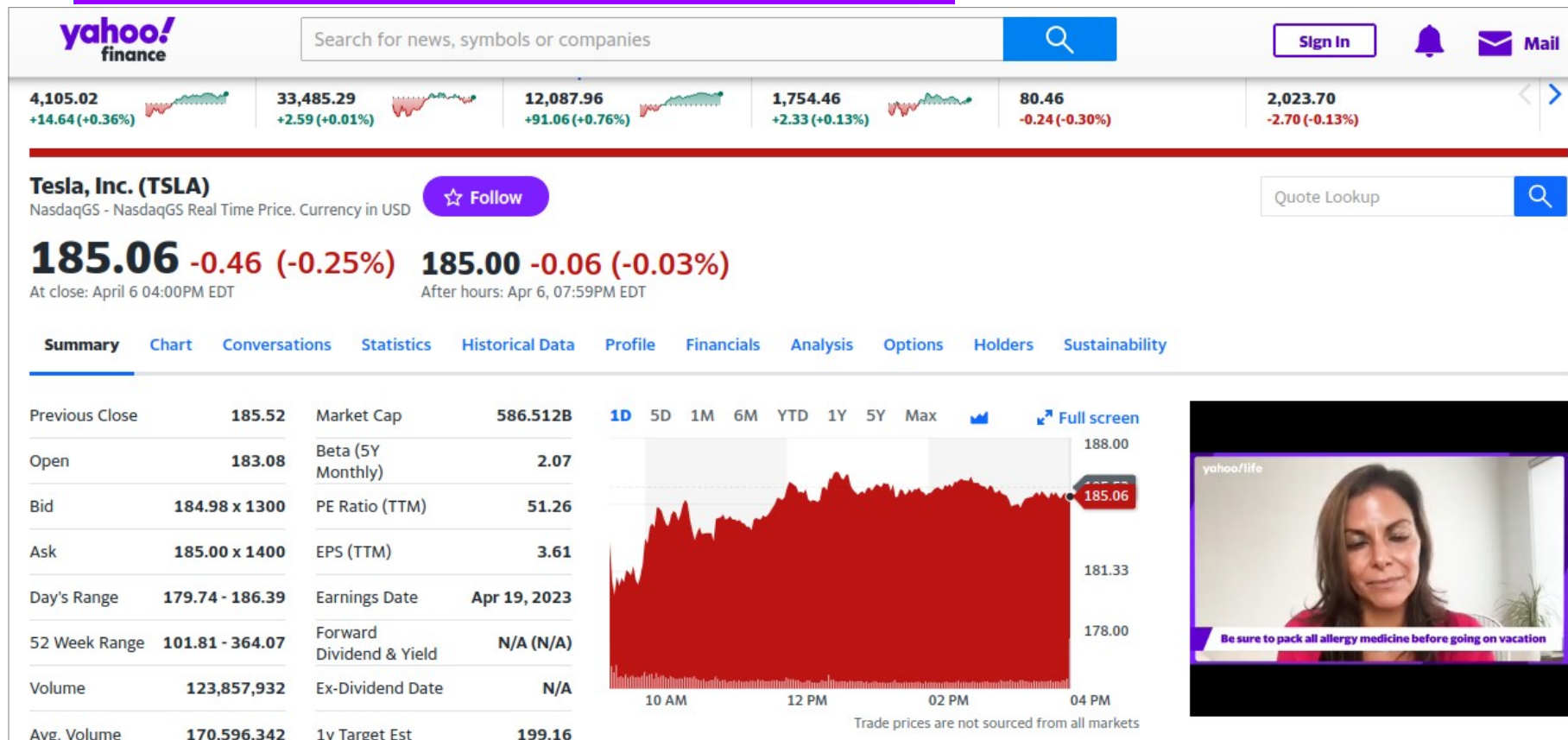


Figura: Código *Python* para finanças – Yahoo Finance.



# Código *Python* para finanças

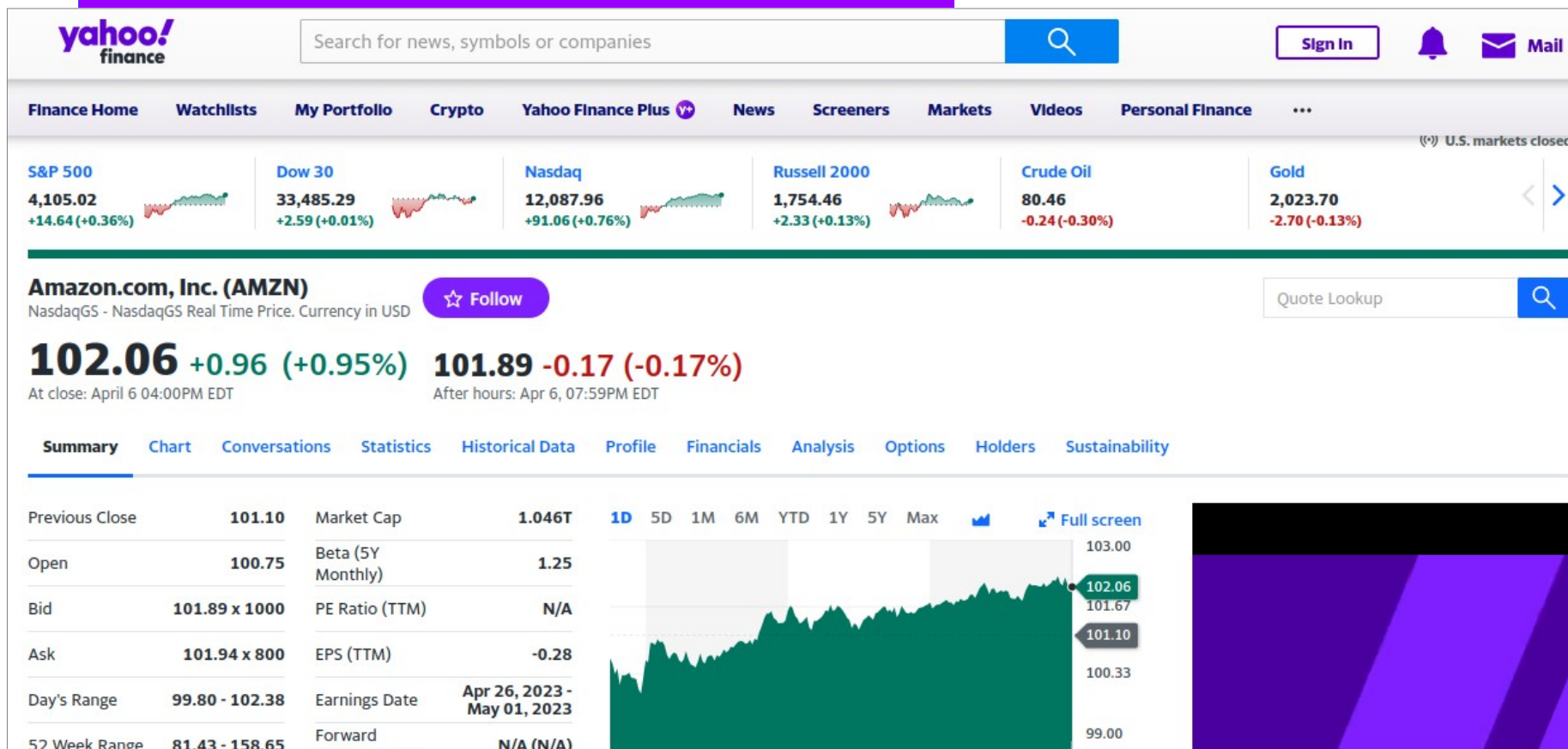


Figura: Código *Python* para finanças – Yahoo Finance.

# Código *Python* para finanças

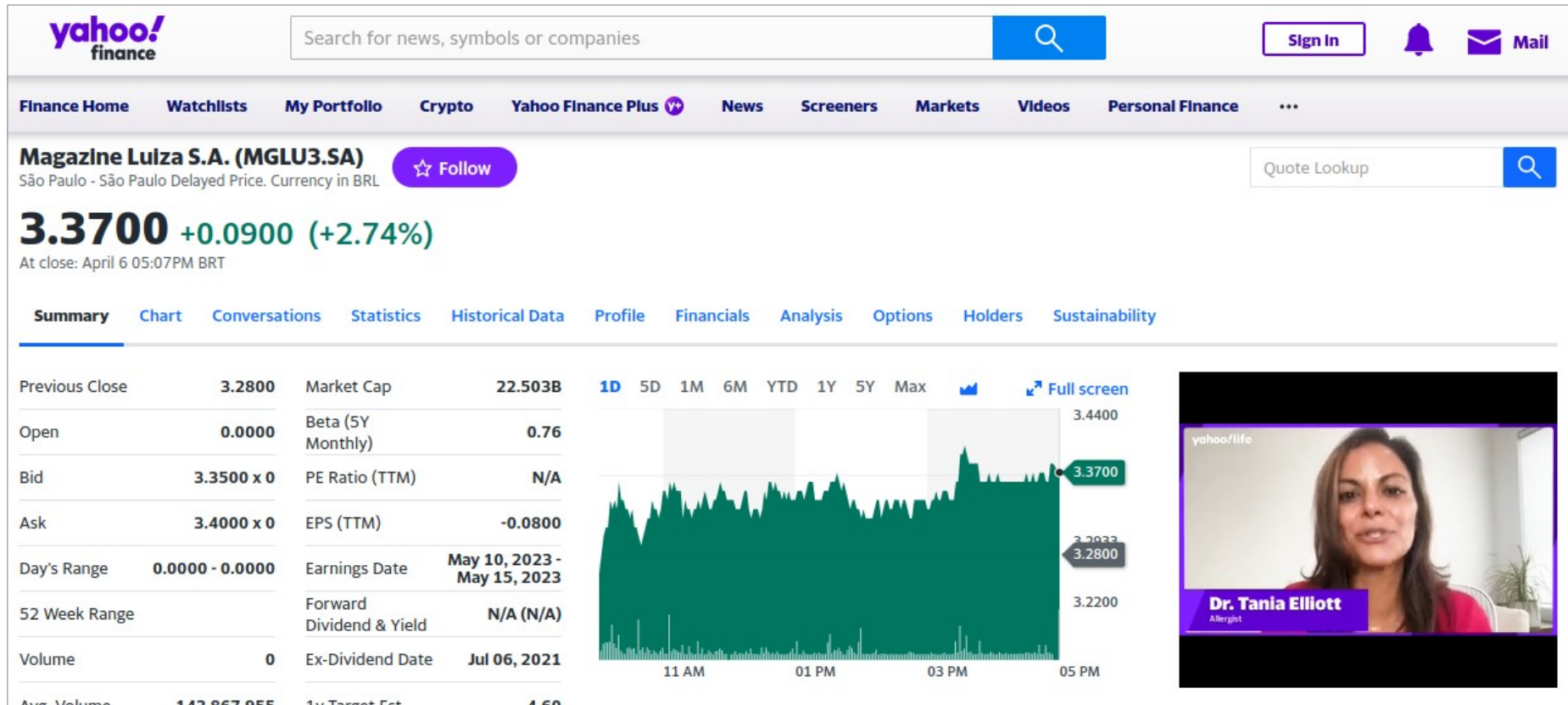


Figura: Código *Python* para finanças – Yahoo Finance.

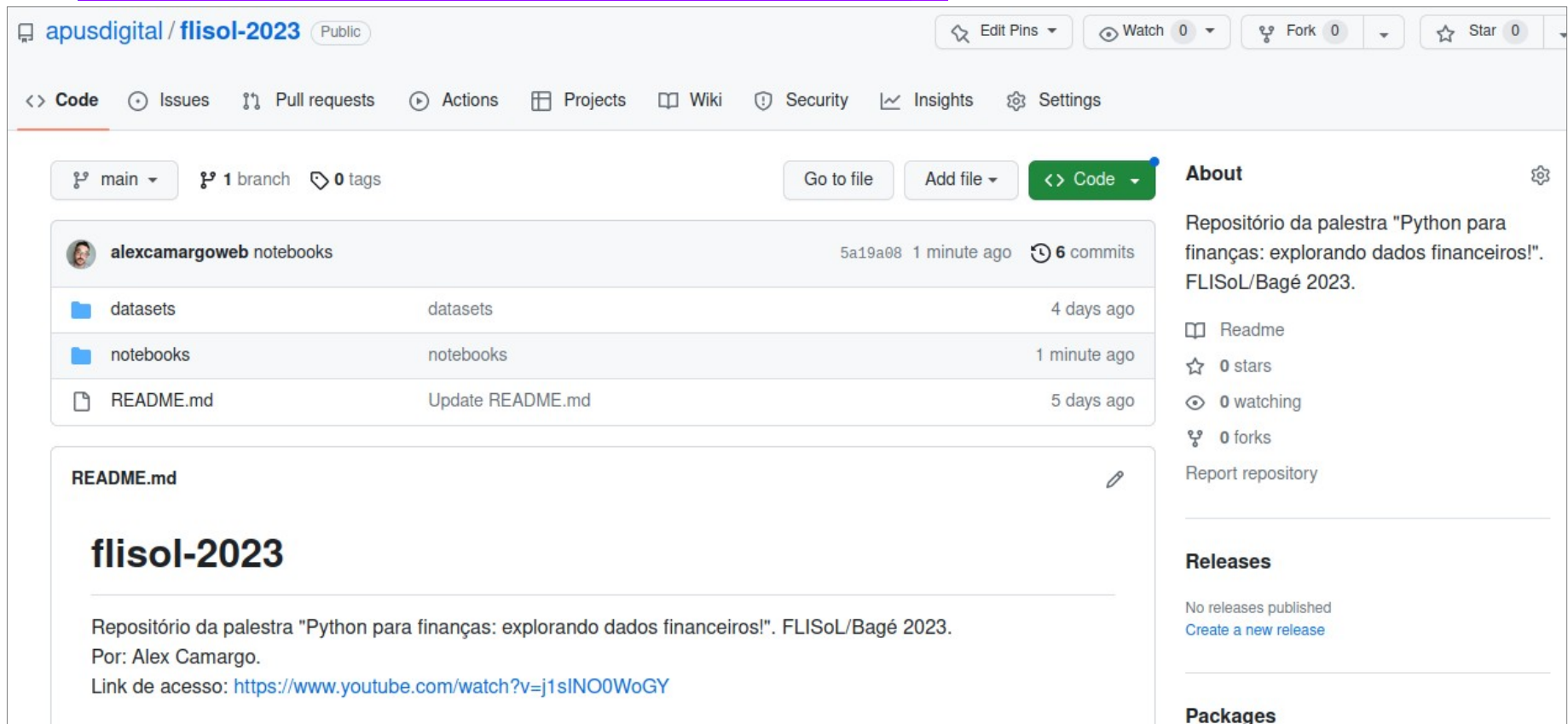
# Código *Python* para finanças

---

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

- **Performance:** 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



apusdigital / flisol-2023 Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

alexcamargoweb notebooks 5a19a08 1 minute ago 6 commits

File	Commit	Time
datasets	datasets	4 days ago
notebooks	notebooks	1 minute ago
README.md	Update README.md	5 days ago

README.md

## flisol-2023

Repositório da palestra "Python para finanças: explorando dados financeiros!". FLISoL/Bagé 2023.  
Por: Alex Camargo.  
Link de acesso: <https://www.youtube.com/watch?v=j1sINO0WoGY>

About  
Repositório da palestra "Python para finanças: explorando dados financeiros!". FLISoL/Bagé 2023.  
Readme  
0 stars  
0 watching  
0 forks  
Report repository

Releases  
No releases published  
[Create a new release](#)

Packages

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 performance através de **GPUs** e **TPUs** “gratuitas”.
- ***Python* é apenas uma ferramenta! Uma análise financeira requer conhecimentos financeiros sólidos e ajuda profissional.**





APUS Digital

Inovação em primeiro lugar.

# Principais referências

---

## **B3: A Bolsa do Brasil**

[https://www.b3.com.br/pt\\_br/](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/>