



APOSTILA

# Automatizando tarefas

AULA 2

# Seja bem-vindo(a) à imersão

## Semana do Python na prática

É um prazer ter você aqui com a gente nessa imersão!

Essa apostila tem como objetivo trazer de uma forma simples e direta todo o conteúdo que foi passado na aula ao vivo.

Aqui na Empowerdata, acreditamos fortemente no aprendizado por projetos e por isso, cada dia da imersão traz um novo projeto a ser desenvolvido.

Lembre-se sempre na nossa metodologia ORA:

**OBSERVAR:** durante a aula ao vivo, apenas acompanhe o que está sendo feito e anote as explicações.

**REPETIR:** depois, refaça o projeto com o material em mãos e caso tenha alguma dificuldade, reveja a aula. Todas as aulas ficam gravadas e disponíveis no Youtube durante a semana da imersão.

**APLICAR:** agora que você observou e repetiu, é hora de aplicar o que aprendeu. Crie novos projetos com o conhecimento adquirido.

## O que vamos aprender?

- Instalar e utilizar bibliotecas
- Buscar dados de ações automaticamente
- Gerar estatísticas sobre esses dados
- Automatizar o controle do mouse e do teclado com o **pyautogui**
- Utilizar strings de múltiplas linhas
- Enviar e-mail de forma automática

## Yahoo Finanças

Site que fornece notícias sobre finanças, incluindo **cotações de ações**.

**Yahoo Finanças**

[Clique aqui para acessar](#)

## Obtendo ações de forma automática

### Instalando a biblioteca yfinance e matplotlib

#### TERMINAL

```
pip install yfinance
```

```
pip install matplotlib
```



## Buscando os dados de uma ação

### ENTRADA

```
import yfinance

ticker = input("Digite o código da ação: ")
dt_inicial = input("Digite a data inicial (aaaa-mm-dd): ")
dt_final = input("Digite a data final (aaaa-mm-dd): ")
dados = yfinance.Ticker(ticker)
tabela = dados.history(start=dt_inicial, end=dt_final)
tabela
```

### SAÍDA

	Open	High	Low	Close	Volume	Dividends	Stock Splits
Date							
2020-01-02 00:00:00-03:00	11.171028	11.240596	11.097800	11.240596	37774500	0.0	0.0
2020-01-03 00:00:00-03:00	11.306501	11.438313	11.149060	11.149060	71595600	0.0	0.0
2020-01-06 00:00:00-03:00	11.141739	11.328472	10.965991	11.280873	81844000	0.0	0.0
2020-01-07 00:00:00-03:00	11.284532	11.306500	11.156382	11.236934	32822000	0.0	0.0
2020-01-08 00:00:00-03:00	11.236934	11.266226	11.072170	11.167367	48215600	0.0	0.0
...	...	...	...	...	...	...	...
2020-12-22 00:00:00-03:00	9.959295	10.058155	9.904372	9.988587	46513200	0.0	0.0
2020-12-23 00:00:00-03:00	10.043507	10.343751	10.014215	10.233906	49038900	0.0	0.0
2020-12-28 00:00:00-03:00	10.384027	10.442611	10.318120	10.318120	36313300	0.0	0.0
2020-12-29 00:00:00-03:00	10.391352	10.409660	10.248554	10.351076	29891100	0.0	0.0
2020-12-30 00:00:00-03:00	10.380369	10.431630	10.325446	10.376707	36968200	0.0	0.0

## Selecionando apenas a coluna de Fechamento (Close)

### ENTRADA

```
fechamento = tabela.Close
fechamento
```

## SAÍDA

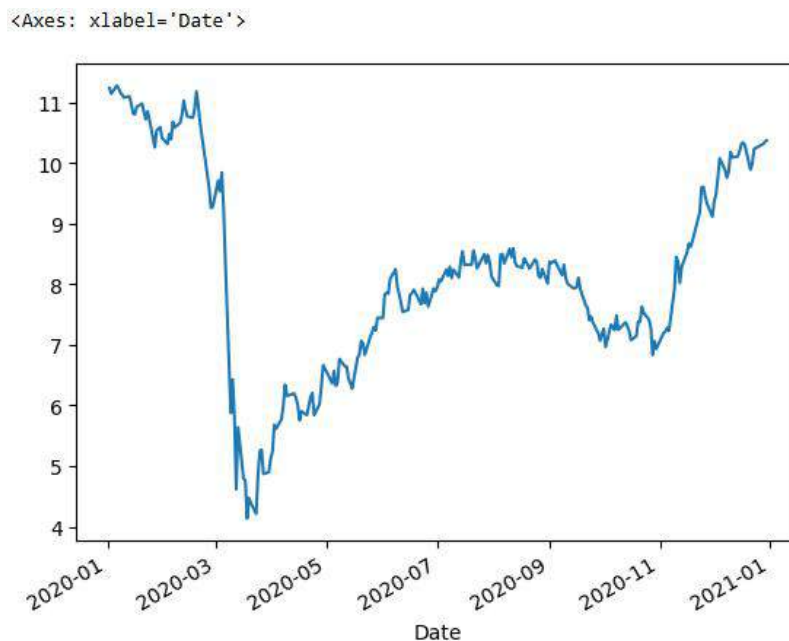
```
Date
2020-01-02 00:00:00-03:00    11.240596
2020-01-03 00:00:00-03:00    11.149060
2020-01-06 00:00:00-03:00    11.280873
2020-01-07 00:00:00-03:00    11.236934
2020-01-08 00:00:00-03:00    11.167367
...
2020-12-22 00:00:00-03:00     9.988587
2020-12-23 00:00:00-03:00    10.233906
2020-12-28 00:00:00-03:00    10.318120
2020-12-29 00:00:00-03:00    10.351076
2020-12-30 00:00:00-03:00    10.376707
Name: Close, Length: 248, dtype: float64
```

**Gerando um gráfico de linha (usa a biblioteca matplotlib)**

## ENTRADA

```
fechamento.plot()
```

## SAÍDA



## Gerando as Estatísticas

Muitas vezes, precisamos fazer conversões entre tipos de dados e no Python isso é feito de uma forma muito simples.

### ENTRADA

```
maxima = round(fechamento.max(), 2)
minima = round(fechamento.min(), 2)
media = round(fechamento.mean(), 2)

print(maxima)
print(minima)
print(media)
```

### SAÍDA

```
11.28
4.13
8.17
```

## Enviando e-mail de forma automática

### Instalando a biblioteca pyautogui

#### TERMINAL

```
pip install pyautogui
```

### Coordenadas na tela

As coordenadas utilizadas no código irão variar de acordo com o monitor e a resolução da tela que você está usando. Utilize o código abaixo para mapear as suas coordenadas.

## Obtendo as coordenadas na tela

### ENTRADA

```
import time
import pyautogui

time.sleep(5)
print(pyautogui.position())
```

## Código

```
import pyautogui
import pyperclip
import webbrowser
from time import sleep

destinatario = "seu_email@gmail.com"
assunto = "Análises Projeto 2020"

mensagem = f"""
Bom dia,

Segue abaixo as análises da ação {ticker} do período solicitado: {dt_inicial} a {dt_final}:

Cotação máxima: R${maxima}
Cotação mínima: R${minima}
Valor médio: R${media}
"""

# configurar uma pausa entre as ações do pyautogui
pyautogui.PAUSE = 3

# abrir o navegador no gmail
webbrowser.open("www.gmail.com")
sleep(3)
```

## Continuação

```
# clicar no botão "Escrever"
pyautogui.click(x=2503, y=314)

# Preencher Para
pyperclip.copy(destinatario)
pyautogui.hotkey("ctrl", "v")
pyautogui.hotkey("tab")

# Preencher Assunto
pyperclip.copy(assunto)
pyautogui.hotkey("ctrl", "v")
pyautogui.hotkey("tab")

# Preencher corpo do e-mail
pyperclip.copy(mensagem)
pyautogui.hotkey("ctrl", "v")

# Clicar no botão enviar
pyautogui.click(x=4056, y=1360)

# fechar a aba
pyautogui.hotkey("ctrl", "f4")

print("E-mail enviado com sucesso!")
```



SEMANA PYTHON NA PRÁTICA



ACOMPANHE MAIS CONTEÚDOS EM



Instagram @empowerdata



Instagram @empowerpython



Canal no Youtube



Empowerdata