

**Educação  
Profissional  
Paulista**

Técnico em  
**Ciência de  
Dados**

# Manipulação de arquivos

## Escrita de arquivos em Python

Aula 1

Código da aula: [DADOS]ANO1C2B3S17A1

**Programação  
aplicada à Ciência  
de Dados**

## **Mapa da unidade 4 Componente 3**

*semana*

**16**

Leitura de arquivos  
em Python

*semana*

**17**

**Você está aqui!**

Escrita de arquivos  
em Python

**Visualização  
de dados**

**Mapa  
da unidade 3  
Componente 4**

**Você está aqui!**

Escrita de arquivos em Python

**17**

**Aula 1**

Código da aula: [DADOS]ANO1C2B3S17A1



## Objetivos da aula

- Revisar e entender melhor os conceitos básicos de manipulação de arquivo, abrir, ler e escrever.
- Conhecer o que é módulo, pacote e biblioteca.



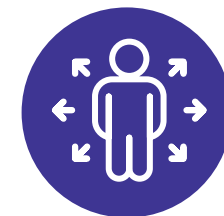
## Recursos didáticos

- Recurso audiovisual para exibição de vídeos e imagens.
- Acesso ao laboratório de informática e/ou à internet.
- Software Anaconda/Jupyter Notebook instalado ou similar.



## Duração da aula

50 minutos



## Competências técnicas

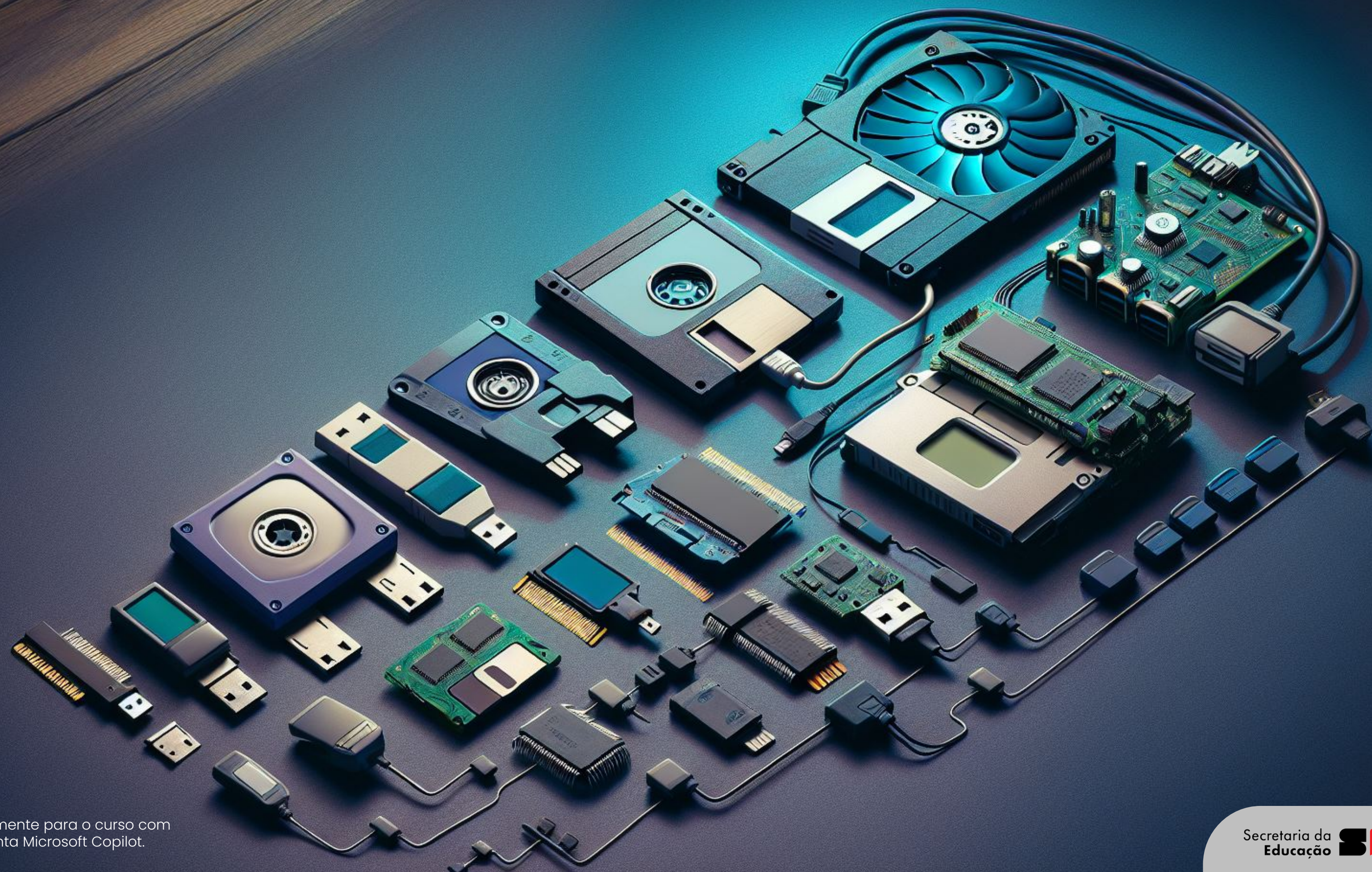
- Ser proficiente em linguagens de programação para manipular e analisar grandes conjuntos de dados.
- Usar técnicas para explorar e analisar dados, aplicar modelos estatísticos, identificar padrões, realizar inferências e tomar decisões com base em evidências.



## Competências socioemocionais

- Colaborar efetivamente com outros profissionais, como cientistas de dados e engenheiros de dados; trabalhar em equipes multifuncionais, colaborando com colegas, gestores e clientes.









## Primeiras ideias

### Onde armazenamos as informações?

Vocês já pensaram na evolução dos dispositivos de armazenamento de dados portátil?

Alguém já viu ou conhece **disquete**? Sabe qual era sua capacidade de armazenamento?

Vocês sabem qual é a capacidade de armazenamento de um **pen drive**?

Ponto de partida

Já imaginou poder acessar informações da internet automaticamente sem precisar copiar um a um?

Neste site, tem a convocação de mais de cem alunos que passaram no vestibular. Suponha que você precise do número de identificação, do nome e do e-mail de todos esses alunos. O que você faria? Copiaria um a um? E se a lista mudar, você faria tudo novamente?

Vestibular

Alunos aprovados

Número da inscrição	Nome	Email
17236	Margarida Were Vomsirnjol	tanderson@yahoo.com
99746	Guimar Sanches Valverde Aoon	taylorbarron@hotmail.com
11123	Rose Jéssica Hedo Ecmson	uoneal@hotmail.com
74376	Gigi Melanie Migoya Kornfirnblosi	marshrichard@hotmail.com
29756	Elisângela Tainá Cidunu de Bragança	madisonowens@gmail.com
15562	Caio Itamar Pinhão	joshua33@hotmail.com
29178	Max de Alencar Naves	carrteresa@yahoo.com
50741	Andréia Lina de França	ujohnson@gmail.com
96397	Luiz Mário de Fonseca	jasonwilliams@gmail.com
10869	Rodolfo Wesley de Guimarães Júnior	christine18@yahoo.com
95556	Dionísio de Ayres dos Santos	tracyjackson@hotmail.com
59455	Omar Benedito Tufauzov Gatetani	lisa42@gmail.com
60777	Rita Genam	christopherrobbins@hotmail.com
91002	Reginaldo de Paiva Teles	paige43@yahoo.com
70587	Aparecida Daiane da Silva de Moreira	cgarcia@yahoo.com
83366	Juan Eduardo Emmi Banhos de Souza	richard21@hotmail.com
93213	Jackson Baltazar Ximenes Neto	teresa38@yahoo.com

Site\_vestibular

Arquivo Editar Ver Inserir Formatar Dados Ferramentas Extensões Ajuda

F7

	A	B	C
1	Número da inscrição	Nome	Email
2	17236	Margarida Were Vomsirnjol	tanderson@yahoo.com
3	99746	Guimar Sanches Valverde Aoon	taylorbarron@hotmail.com
4	11123	Rose Jéssica Hedo Ecmson	uoneal@hotmail.com
5	74376	Gigi Melanie Migoya Kornfirnblosi	marshrichard@hotmail.com
6	29756	Elisângela Tainá Cidunu de Bragança	madisonowens@gmail.com
7	15562	Caio Itamar Pinhão	joshua33@hotmail.com
8	29178	Max de Alencar Naves	carrteresa@yahoo.com
9	50741	Andréia Lina de França	ujohnson@gmail.com
10	96397	Luiz Mário de Fonseca	jasonwilliams@gmail.com
11	10869	Rodolfo Wesley Isom de Guimarães Júnior	christine18@yahoo.com
12	95556	Dionísio de Ayres dos Santos	tracyjackson@hotmail.com
13	59455	Omar Benedito Tufauzov Gatetani	lisa42@gmail.com
14	60777	Rita Genam	christopherrobbins@hotmail.com
15	91002	Reginaldo de Paiva Teles	paige43@yahoo.com
16	70587	Aparecida Daiane Nunes da Silva de Moreira	cgarcia@yahoo.com
17	83366	Juan Eduardo Emmi Banhos de Souza	richard21@hotmail.com
18	93213	Jackson Baltazar Ximenes Neto	teresa38@yahoo.com
19	32875	Leo de Madureira	ragnerdaferene@hotmail.com
20	89835	João de Lima da Silva d'Ávila	joseph93@hotmail.com
21	49590	Eric Rivaldo Ugal	ktaylor@gmail.com
22	96229	Tina Rosilda de Gímenes Fagolchason	stephen93@gmail.com
23	12521	Vera Ana Reayo Cernugo Pibopa	jilhall@hotmail.com



Construindo  
o **conceito**

## Como salvar um texto extraído da internet?

Existe a técnica de **web scraping** que extrai dados de sites da web.

No próximo slide, vocês vão ver o exemplo de um código de como extrair dados.

E como salvar o resultado desse **web scraping**? Será que é permitido fazer isso em qualquer site?



### Curiosidade

**Web scraping** é o processo de extrair dados de sites da web de forma automatizada, geralmente por meio do *parsing* do HTML das páginas web. Pode-se usar a biblioteca **requests** para fazer uma solicitação HTTP para a página da web e, em seguida, usar a biblioteca **BeautifulSoup** para analisar o HTML retornado e extrair o texto desejado.

Construindo  
o **conceito**

# Como salvar um texto extraído da internet?

Esse código é um exemplo de como extrair os dados de um site:

```
import requests
from bs4 import BeautifulSoup

url = "https://sites.google.com/view/vestibular123"

response = requests.get(url)
content = response.content

soup = BeautifulSoup(content, "html.parser")
meta_tags = soup.find_all('meta', {'property': 'og:description'})

texto_salvar = meta_tags[0]['content']

print(texto_salvar)
```

Número da inscrição	Nome
Email	
17236	Margarida Were Vomsirnjol
tanderson@yahoo.com	
99746	Guiomar Sanches Valverde Aoson
taylorbarron@hotmail.com	
11123	Rose Jéssica Hedo Ecemson
uoneal@hotmail.com	
74376	Gigi Melanie Migoya

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



## Construindo o **conceito**

Para ler um arquivo de texto em Python, você pode usar:

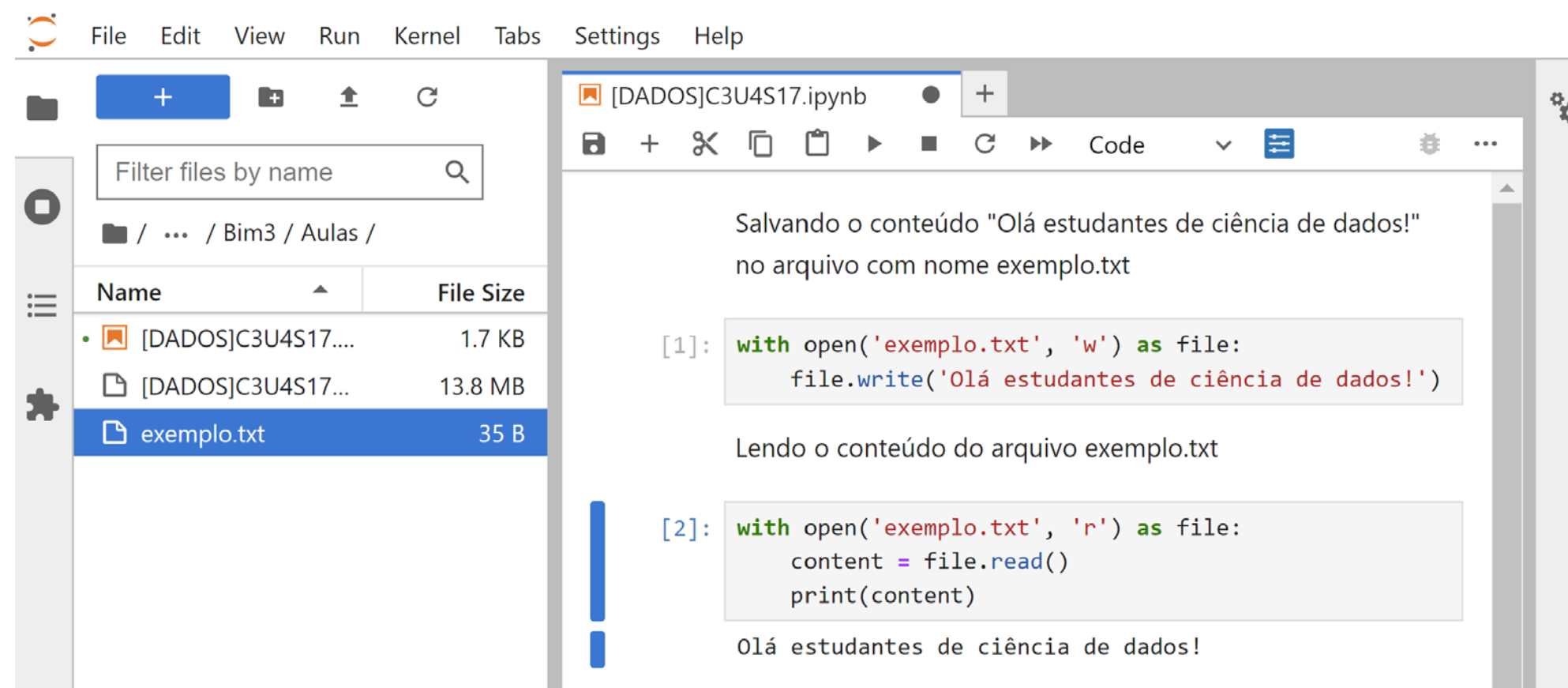
- o método `open()` para abrir o arquivo.
- o método `read()` para ler o conteúdo do arquivo.

Para salvar um arquivo de texto em Python, você pode usar:

- o método `open()` para abrir o arquivo no modo de escrita.
- o método `write()` para escrever o conteúdo no arquivo.

Construindo  
o **conceito**

## Ler um arquivo



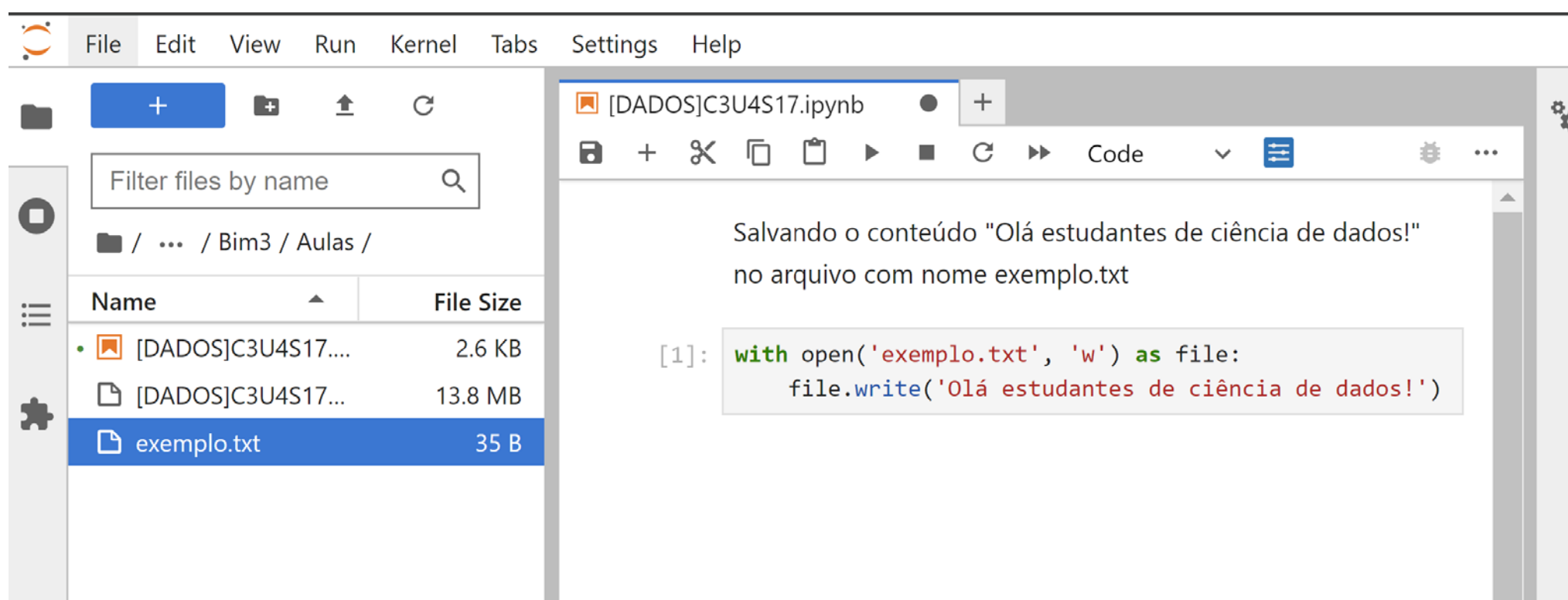
Nesse exemplo de código, o conteúdo de um arquivo de texto chamado exemplo.txt é lido e impresso na tela.

Elaborado especialmente para o curso com a ferramenta Jupyter Lab.



Construindo  
o **conceito**

## Salvar um arquivo



Observe que o arquivo com nome exemplo.txt foi salvo. Lembre-se de que o modo de escrita ('w') sobrescreverá o conteúdo existente no arquivo.

Se você quiser adicionar conteúdo ao final do arquivo, use o modo de anexação ('a') em vez disso.

Elaborado especialmente para o curso com a ferramenta Jupyter Lab.

Construindo  
o **conceito**

## Módulos em Python

Um módulo em Python é:

- um arquivo que contém definições e instruções Python;
- usado para organizar projetos e programas em pedaços menores, arquivos .py, de assuntos específicos e relacionados entre si.



### Tome nota

O módulo `math.py` é usado para trabalhar com matemática, enquanto o módulo `random.py` é usado para trabalhar com números aleatórios. O módulo `os.py` é usado para trabalhar com arquivos, enquanto o módulo `time.py` é usado para trabalhar com tempo (data, dia, ano etc.).



## Construindo o **conceito**

# Módulos em Python

Para usar um módulo em um programa Python, você precisa importá-lo. Existem várias maneiras de fazer isso, como:

- `import;`
- `from;`
- `as.`

Além disso, você pode usar importações relativas dentro de um pacote.



### Dica

É possível, também, organizar módulos em pacotes, que são coleções de módulos com uma estrutura de nomes com pontos. Isso permite que você organize seus projetos de maneira ainda mais eficiente.

## Construindo o conceito

# Módulos em Python

```
import requests
from bs4 import BeautifulSoup

url = "https://sites.google.com/view/vestibular123"

response = requests.get(url)
content = response.content

soup = BeautifulSoup(content, "html.parser")
meta_tags = soup.find_all('meta', {'property': 'og:description'})

texto_salvar = meta_tags[0]['content']

print(texto_salvar)
```

Número da inscrição	Nome
Email	
17236	Margarida Were Vomsirnjol
tanderson@yahoo.com	
99746	Guiomar Sanches Valverde Aoson
taylorbarron@hotmail.com	
11123	Rose Jéssica Hedo Ecemson
uoneal@hotmail.com	
74376	Gigi Melanie Migoya

Vocês repararam nas duas primeiras linhas do código ao lado?

Representam a instrução de importação de um módulo.

O sistema de importação do Python é o mecanismo responsável por carregar módulos e pacotes de diferentes fontes.

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



## Módulos e pacotes

- ▶ Em Python, um **módulo** é um arquivo com a extensão .py contendo um script Python.
- ▶ Um **pacote** é um conjunto de módulos organizados hierarquicamente em diretórios.
- ▶ A **notação ponto** é usada internamente pelo Python para separar os módulos em espaços de nomes diferentes.



### Dica

Não se preocupe com a diferença entre **módulo**, **pacote** e **biblioteca**. Saiba como importar um módulo para facilitar nossas tarefas.

# Construindo o **conceito**

## Python: curiosidades

01

**Um módulo é um arquivo** que contém definições e instruções Python. O nome do arquivo é o nome do módulo acrescido do sufixo `.py`. Dentro de um módulo, o nome do módulo (como uma string) está disponível como valor da variável global `__name__`.

02

**Os pacotes são diretórios** que contêm um arquivo especial chamado `__init__.py` e, opcionalmente, outros módulos ou subpacotes. Um pacote é um tipo de módulo Python que pode ser usado para organizar e reutilizar código em projetos maiores.

04

**Um pacote é um módulo Python** que possui o atributo `__path__`. O atributo `__path__` é uma lista de strings que especifica os diretórios que contêm os módulos do pacote. Quando um pacote é importado, o Python adiciona o diretório que contém o arquivo `__init__.py` à lista `sys.path`. Isso permite que o Python encontre os módulos do pacote quando eles são importados.

03

**Uma biblioteca é um conjunto de módulos ou pacotes** que podem ser usados para realizar tarefas específicas. Por exemplo, a biblioteca padrão do Python contém módulos para manipulação de arquivos, expressões regulares, matemática etc. Além disso, existem muitas bibliotecas de terceiros disponíveis para Python que podem ser instaladas usando o gerenciador de pacotes `pip`.

Elaborado especialmente para o curso com imagem © Getty Images.

Construindo  
o **conceito**

## Resumindo...



### Módulo

É um arquivo Python que contém definições e instruções.



### Pacote

É um diretório que contém um arquivo `__init__.py` e outros módulos ou subpacotes.



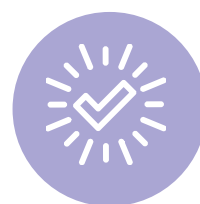
### Biblioteca

É um conjunto de módulos ou pacotes que podem ser usados para realizar tarefas específicas.



## Construindo o conceito

# Bibliotecas – vantagens



### Aumento da produtividade

Ao utilizar bibliotecas em Python, é possível aproveitar o trabalho de outros programadores, economizando tempo e esforço na implementação de funcionalidades comuns. Isso permite que os desenvolvedores foquem em aspectos mais complexos do projeto.



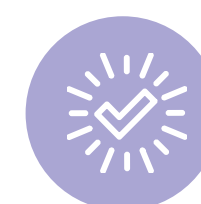
### Maior qualidade do código

As bibliotecas em Python são geralmente desenvolvidas e testadas por uma comunidade de programadores experientes. Isso significa que o código das bibliotecas passa por revisões e melhorias constantes, resultando em um software mais confiável e robusto.



### Acesso a funcionalidades avançadas

As bibliotecas em Python oferecem uma ampla gama de funcionalidades que podem ser utilizadas para resolver problemas específicos. Por exemplo, se você precisa processar dados em formato CSV, pode utilizar a biblioteca **pandas**, que fornece ferramentas para manipulação e análise de dados tabulares.



### Facilidade de integração

As bibliotecas em Python são projetadas para serem facilmente integradas em outros projetos. Isso significa que você pode combinar diferentes bibliotecas para obter funcionalidades personalizadas e adaptadas às suas necessidades.

## Construindo o **conceito**

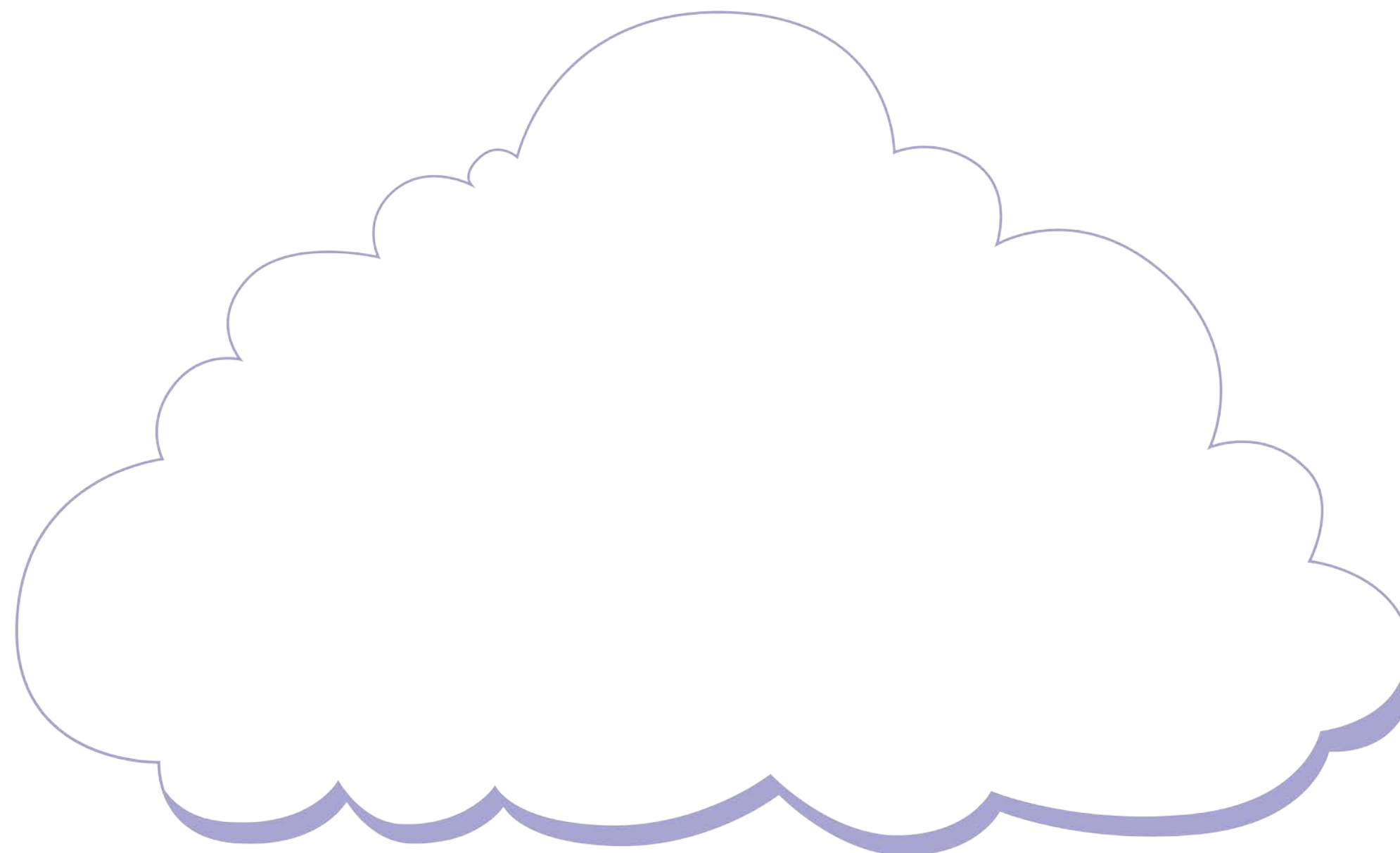
# Bibliotecas

Existem várias bibliotecas em Python que são comumente usadas em Ciência de Dados. Estas são algumas das mais populares:

	Definição
NumPy	É uma biblioteca fundamental para a computação numérica em Python, permitindo operações eficientes em <i>arrays</i> multidimensionais.
Pandas	É uma biblioteca para análise e manipulação de dados que se utiliza de DataFrames para a criação, a exportação e o manuseio de informações tabulares.
Matplotlib	É uma biblioteca para visualização de dados em Python. Ela permite a criação de gráficos, histogramas, gráficos de dispersão, entre outros.
Scikit-learn	É uma biblioteca para aprendizado de máquina em Python. Ela fornece ferramentas para classificação, regressão, agrupamento, redução de dimensionalidade, seleção de modelos e pré-processamento de dados.
NLTK	É uma biblioteca para processamento de linguagem natural em Python. Ela fornece ferramentas para tokenização, análise de sentimento, classificação de texto, entre outras.

Elaborado especialmente para o curso.

# Nuvem de palavras



© Getty Images

O que nós  
**aprendemos  
hoje?**





© Getty Images

O que nós  
**aprendemos**  
**hoje?**

## Então ficamos assim...

- 1** Na teoria, podemos extrair dados de sites da internet usando uma técnica chamada de *web scraping*.
- 2** Usamos o método **read()** para ler o conteúdo de um arquivo e **write()** para escrever – salvar – o conteúdo em um arquivo.
- 3** Bibliotecas, pacotes e módulos ajudam a realizar algumas tarefas, pois é um conjunto de código já escrito e resolve problemas específicos usando funcionalidades extras.

# Saiba mais

Quer saber mais sobre quais as possibilidades de quem trabalha com Ciência de Dados?

Nesse episódio do Hipsters Ponto Tech, a conversa é sobre a carreira em Ciência de Dados, sobre as diversas vertentes e possibilidades de atuação das pessoas que trabalham com dados ou em áreas muito próximas a ela.

ALURA. Carreira em Dados: Hipsters Ponto Tech #273. *Alura*, 5 out. 2021.

Disponível em:

<https://cursos.alura.com.br/extra/hipsterstech/carreira-em-dados-hipsters-ponto-tech-273-a1160>.

Acesso em: 29 abr. 2024.

# Referências da aula

MENEZES, N. N. C. *Introdução à programação com Python: algoritmos e lógica de programação para iniciantes*. São Paulo: Novatec, 2019.

PYTHON ORG. 6. *Módulos*. Disponível em: <https://docs.python.org/pt-br/3/tutorial/modules.html>. Acesso em: 29 abr. 2024.

PYTHON ORG. 7.11. A instrução Import. Disponível em: [https://docs.python.org/pt-br/3/reference/simple\\_stmts.html#the-import-statement](https://docs.python.org/pt-br/3/reference/simple_stmts.html#the-import-statement). Acesso em: 29 abr. 2024.

PYTHON ORG. 5. *O sistema de importação*. Disponível em: <https://docs.python.org/pt-br/3/reference/import.html#importsystem>. Acesso em: 29 abr. 2024.

Identidade visual: imagens © Getty Images.



**Educação  
Profissional  
Paulista**

Técnico em  
**Ciência de  
Dados**