



escola
britânica de
artes criativas
& tecnologia

Desenvolvedor Full Stack Python

Módulos & Pacotes

Módulo | Python: Módulos & Pacotes

Caderno de Aula

Professor [André Perez](#)

Tópicos

1. from / import / as;
 2. Módulo;
 3. Pacote;
 4. Baixando pacotes.
-

Aulas

1. from / import / as

Os módulos nativos do Python podem ser encontrados neste [link](#).

1.1. import

Exemplo: `random`

```
In [ ]: import random
```

```
In [ ]: escolha = random.choice([1, 2, 3])  
print(escolha)
```

```
In [ ]: numero_aleatorio = random.random() # entre [0,1)  
print(numero_aleatorio)
```

Exemplo: `math`

```
In [ ]: import math
```

```
In [ ]: potencia = math.pow(10, 10)
        print(potencia)
```

```
In [ ]: num = math.ceil(10.1)
        print(num)
```

```
In [ ]: print(math.pi)
```

1.2. from, import

Exemplo: `time`

```
In [ ]: from time import time
```

```
In [ ]: print(time())
```

```
In [ ]: sleep(5)
```

```
In [ ]: from time import time, sleep
```

```
In [ ]: sleep(5)
```

1.3. from, import, as

Exemplo: `datetime`

```
In [ ]: from datetime import datetime as dt
```

```
In [ ]: print(dt.now())
```

```
In [ ]: print(dt.now().day)
```

```
In [ ]: print(dt.now().year)
```

2. Módulos

2.1. Motivação

- Classe Arquivo CSV

```
In [ ]: class ArquivoCSV(object):
        def __init__(self, arquivo: str):
            self.arquivo = arquivo
            self.conteudo = self._extrair_conteudo()
            self.colunas = self._extrair_nome_colunas()
```

```

def _extrair_conteudo(self):
    conteudo = None
    with open(file=self.arquivo, mode='r', encoding='utf8') as arquivo:
        conteudo = arquivo.readlines()
    return conteudo

def _extrair_nome_colunas(self):
    return self.conteudo[0].strip().split(sep=',')

def extrair_coluna(self, indice_coluna: str):
    coluna = list()
    for linha in self.conteudo:
        conteudo_linha = linha.strip().split(sep=',')
        coluna.append(conteudo_linha[indice_coluna])
    coluna.pop(0)
    return coluna

```

- Arquivo banco.csv

```

In [ ]: %%writefile banco.csv
age,job,marital,education,default,balance,housing,loan
30,unemployed,married,primary,no,1787,no,no
33,services,married,secondary,no,4789,yes,yes
35,management,single,tertiary,no,1350,yes,no
30,management,married,tertiary,no,1476,yes,yes
59,blue-collar,married,secondary,no,0,yes,no
35,management,single,tertiary,no,747,no,no
36,self-employed,married,tertiary,no,307,yes,no
39,technician,married,secondary,no,147,yes,no
41,entrepreneur,married,tertiary,no,221,yes,no
43,services,married,primary,no,-88,yes,yes

```

```

In [ ]: arquivo_banco = ArquivoCSV(arquivo='./banco.csv')

```

- Extrair a coluna de `education`

```

In [ ]: education = arquivo_banco.extrair_coluna(indice_coluna=3)
print(education)

```

2.2. Definição

Vamos criar um módulo (arquivo) com o nome `arquivo_csv.py` com o código da classe `ArquivoCSV`.

2.3. Revisitando a motivação

Vamos importar a classe `ArquivoCSV` do módulo (arquivo) `arquivo_csv.py`.

```

In [ ]: from arquivo_csv import ArquivoCSV

arquivo_banco_modulo = ArquivoCSV(arquivo='./banco.csv')

```

```

In [ ]: education = arquivo_banco_modulo.extrair_coluna(indice_coluna=3)
print(education)

```

3. Pacotes

3.1. Motivação

- Classe Arquivo Texto

```
In [ ]: class ArquivoTXT(object):

    def __init__(self, arquivo: str):
        self.arquivo = arquivo
        self.conteudo = self._extrair_conteudo()

    def _extrair_conteudo(self):
        conteudo = None
        with open(file=self.arquivo, mode='r', encoding='utf8') as arquivo:
            conteudo = arquivo.readlines()
        return conteudo

    def extrair_linha(self, numero_linha: int):
        return self.conteudo[numero_linha-1]
```

- Arquivo noticia.txt

```
In [ ]: %%writefile noticia.txt
Egito cobra quase US$ 1 bi para liberar navio que bloqueou Canal de Suez
Segundo autoridades, valor será utilizado para recompor as perdas provocados
pelo encalhamento da embarcação de quase 400 metros
```

```
In [ ]: arquivo_noticia = ArquivoTXT(arquivo='./noticia.txt')
```

```
In [ ]: titulo = arquivo_noticia.extrair_linha(numero_linha=1)
print(titulo)
```

3.2. Definição

Vamos criar um módulo (arquivo) com o nome `arquivo_txt.py` com o código da classe `ArquivoTXT`.

Vamos criar um pacote (pasta) com o nome `arquivo` e mover os módulos (arquivos) `arquivo_csv.py` e `arquivo_txt.py` para ela.

3.3. Revisitando a motivação

```
In [ ]: from arquivo.arquivo_csv import ArquivoCSV
        from arquivo.arquivo_txt import ArquivoTXT
```

```
In [ ]: arquivo_banco_pacote = ArquivoCSV(arquivo='./banco.csv')
arquivo_noticia_pacote = ArquivoTXT(arquivo='./noticia.txt')
```

```
In [ ]: education = arquivo_banco_pacote.extrair_coluna(indice_coluna=3)
print(education)
```

```
In [ ]: titulo = arquivo_noticia_pacote.extrair_linha(numero_linha=1)
        print(titulo)
```

4. Baixando pacotes

4.1. PyPI

Repositório oficial de pacotes Python ([link](#)).

4.2. PIP

Ferramenta oficial para instalar pacotes Python armazenados no PyPI.

- Instalando pacotes: `pip install <pacote>==<versão>`

```
In [ ]: !pip install requests==2.25.1
```

- Listando pacotes: `pip freeze`

```
In [ ]: !pip freeze
```

```
In [ ]: !pip install -r requirements.txt
```

- Removendo pacotes: `pip uninstall <pacote>`

```
In [ ]: !pip uninstall requests
```

4.3. Requests

Pacote para interação com o protocolo web HTTP ([link](#)).

Exemplo: Extrair a taxa CDI do site da B3.

```
In [ ]: import requests as req

        response = req.get(
            'https://www2.cetip.com.br/ConsultarTaxaDi/ConsultarTaxaDICetip.aspx'
        )
```

```
In [ ]: print(f'status code: {response.status_code}')
```

```
In [ ]: print(response.text)
```

```
In [ ]: import json

        data = json.loads(response.text)
        print(data)
        print(type(data))
```

In []:

```
cdi = None

for key, value in data.items():
    if key == 'taxa':
        cdi = value.replace(',', '.')
        cdi = float(cdi)

print(cdi)
print(type(cdi))
```