



### Exercício 1: Extração e Visualização de Dados CSV

**Objetivo:** Extrair dados de um ficheiro CSV simulado e exibir as primeiras linhas para verificação.

**Cenário:** Uma empresa regista as vendas diárias num ficheiro CSV.

Dados de Entrada (Simulados - vendas\_diarias.csv):

```
Data,Produto,Quantidade,Preco_Unitario
2024-05-01,Teclado,10,25.00
2024-05-01,Rato,5,15.50
2024-05-02,Monitor,2,150.00
2024-05-02,Webcam,8,30.00
2024-05-03,Teclado,12,25.00
```

**Questões:** Escreva o código Python/Pandas para:

1. Criar o ficheiro vendas\_diarias.csv com os dados acima.
2. Extrair os dados para um DataFrame.
3. Exibir as 3 primeiras linhas e o tipo de dados de cada coluna.

### Proposta de Resolução:



```
import pandas as pd

# 1. Criar o ficheiro vendas_diarias.csv
dados = """Data,Produto,Quantidade,Preco_Unitario
2024-05-01,Teclado,10,25.00
2024-05-01,Rato,5,15.50
2024-05-02,Monitor,2,150.00
2024-05-02,Webcam,8,30.00
2024-05-03,Teclado,12,25.00
"""

with open("vendas_diarias.csv", "w") as f:
    f.write(dados)

# 2. Ler o CSV para DataFrame
df = pd.read_csv("vendas_diarias.csv")

# 3. Mostrar primeiras 3 linhas e tipos de dados
print(df.head(3))
print("\nTipos de dados por coluna:")
print(df.dtypes)
```

## Exercício 2: Extração de Dados JSON e Carregamento para CSV

**Objetivo:** Extrair dados de um ficheiro JSON simulado e carregá-los para um novo ficheiro CSV.

**Cenário:** Os dados de stock de produtos são fornecidos num formato JSON.

Dados de Entrada (Simulados - stock\_produtos.json):

```
[
  {"id": "P001", "nome": "Teclado", "stock": 150, "localizacao": "A1"},
  {"id": "P002", "nome": "Rato", "stock": 200, "localizacao": "A2"},
  {"id": "P003", "nome": "Monitor", "stock": 50, "localizacao": "B1"},
  {"id": "P004", "nome": "Webcam", "stock": 80, "localizacao": "B2"}
]
```

**Questões:** Escreva o código Python/Pandas para:

1. Criar o ficheiro stock\_produtos.json com os dados acima.
2. Extrair os dados JSON para um DataFrame.
3. Carregar o DataFrame para um novo ficheiro CSV chamado stock\_limpo.csv.

### Proposta de Resolução:



```
import pandas as pd
import json

# 1. Criar o ficheiro stock_produtos.json
dados_json = [
    {"id": "P001", "nome": "Teclado", "stock": 150, "localizacao": "A1"},
    {"id": "P002", "nome": "Rato", "stock": 200, "localizacao": "A2"},
    {"id": "P003", "nome": "Monitor", "stock": 50, "localizacao": "B1"},
    {"id": "P004", "nome": "Webcam", "stock": 80, "localizacao": "B2"}
]

with open("stock_produtos.json", "w") as f:
    json.dump(dados_json, f, indent=4)

# 2. Ler JSON para DataFrame
df_json = pd.read_json("stock_produtos.json")

# 3. Guardar para CSV
df_json.to_csv("stock_limpo.csv", index=False)

print("Ficheiro stock_limpo.csv criado com sucesso!")
print(df_json)
```

### Exercício 3: ETL Completo com Limpeza e Agregação

**Objetivo:** Realizar um processo ETL completo: Extrair dados de vendas (CSV), limpar dados em falta, agregar o valor total das vendas e carregar o resultado para um novo CSV.

**Cenário:** A tabela de vendas contém valores em falta (NaN) e a coluna de valor unitário está em formato de texto.

Dados de Entrada (Simulados - vendas\_brutas.csv):

ID_Venda	Produto	Cliente_ID	Valor_Unitario	Quantidade
1	Teclado	101	25.00	2
2	Rato	102	15.50	1
3	Monitor	101	150.00	NaN
4	Webcam	103	30.00	5
5	Teclado	102	25.00	3
6	Rato	101	15.50	NaN
7	Monitor	103	150.00	1

**Questões:** Escreva o código Python/Pandas para:

1. Extrair os dados.
2. Transformação (Limpeza): Substituir os valores NaN na coluna Quantidade pela média da coluna.
3. Transformação (Cálculo): Criar uma nova coluna Valor\_Total (Valor\_Unitario \* Quantidade).
4. Transformação (Agregação): Calcular o Total\_Gasto por cada Cliente\_ID.
5. Carregar o resultado final (Cliente\_ID e Total\_Gasto) para um ficheiro CSV chamado total\_gasto\_clientes.csv.

### Proposta de Resolução:



```
import pandas as pd
import numpy as np

# 1. Criar vendas_brutas.csv (simulação)
dados = """ID_Venda,Produto,Cliente_ID,Valor_Unitario,Quantidade
1,Teclado,101,25.00,2
2,Rato,102,15.50,1
3,Monitor,101,150.00,NaN
4,Webcam,103,30.00,5
5,Teclado,102,25.00,3
6,Rato,101,15.50,NaN
7,Monitor,103,150.00,1
"""

with open("vendas_brutas.csv", "w") as f:
    f.write(dados)

# 2. Extrair
df = pd.read_csv("vendas_brutas.csv")

# 3. Limpeza: substituir NaN da coluna Quantidade pela média
media_qtd = df["Quantidade"].astype(float).mean()
df["Quantidade"] = df["Quantidade"].fillna(media_qtd)

# 4. Criar coluna Valor_Total
df["Valor_Total"] = df["Valor_Unitario"] * df["Quantidade"]

# 5. Agrupar gasto por Cliente_ID
resultado = df.groupby("Cliente_ID")["Valor_Total"].sum().reset_index()
resultado.rename(columns={"Valor_Total": "Total_Gasto"}, inplace=True)

# 6. Carregar para CSV
resultado.to_csv("total_gasto_clientes.csv", index=False)

print("\nResultado final:")
print(resultado)
```