



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

# **Projeto e Implementação de Interface Gráfica com Python**

## **Módulo 02 - Manipulação de Dataframe**

Eduardo Jabbur Machado

[ejabbur@unifei.edu.br](mailto:ejabbur@unifei.edu.br)



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

# Manipulação de Dataframe

**Objetivo:** Apresentar a estrutura de dados DataFrame em Python e como manipular dados tabulares usando a biblioteca Pandas.

**Principais Conceitos:** Um DataFrame é uma tabela bidimensional (como uma planilha) onde as linhas e colunas possuem rótulos. Ele é muito usado para armazenar dados em Python e é uma estrutura central do Pandas.



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

# Instalando a biblioteca Pandas

## Instalando o Pandas:

Caso não tenha ainda o Pandas instalado, execute o comando abaixo no terminal ou prompt de comando:

IDE: ***pip install pandas***

Google Colab: ***!pip install pandas***

## Importando o Pandas:

***import pandas as pd***



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

## Criando um dataframe

Um DataFrame pode ser criado a partir de várias fontes de dados, como listas, dicionários, arquivos CSV, etc.

**Criando um DataFrame a partir de um dicionário:**

```
import pandas as pd
dados = {
    "Nome": ["Ana", "Carlos", "Lucas", "Maria"],
    "Idade": [23, 34, 25, 31],
    "Cidade": ["São Paulo", "Rio de Janeiro", "Curitiba", "Belo Horizonte"]
}
df = pd.DataFrame(dados)
print(df)
```



## Visualizando dados do dataframe

Após criar um DataFrame, você pode usar vários métodos para visualizar os dados.

- ❑ `head()`: Exibe as primeiras 5 linhas do DataFrame (padrão).

```
print(df.head())
```

- ❑ `tail()`: Exibe as últimas 5 linhas do DataFrame.

```
print(df.tail())
```

- ❑ `shape`: Retorna a forma (número de linhas e colunas).

```
print(df.shape)
```



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

## Acessando Colunas e Linhas

# Acessando uma coluna

```
print(df["Nome"])
```

# Acessando uma coluna com atributo

```
print(df.Nome)
```

# Acessando a linha 0

```
print(df.iloc[0])
```



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

## Filtrando dados no dataframe

Você pode filtrar dados usando condições. Por exemplo, para filtrar pessoas com idade maior que 30:

# Filtrando dados

```
maiores_que_30 = df[df["Idade"] > 30]
```

```
print(maiores_que_30)
```



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

# Modificando dados no dataframe

Você pode alterar os dados de uma coluna ou linha específica

**# Alterando uma célula específica:**

```
df.at[1, "Idade"] = 35 # Altera a idade de Carlos para 35
```

```
print(df)
```

**# Alterando uma coluna específica (inteira):**

```
df["Cidade"] = ["São Paulo", "Rio de Janeiro", "Curitiba",  
"Campinas"]
```

```
print(df)
```





**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

# Adicionando e Removendo Colunas

**# Adicionando uma coluna:**

```
df["Profissão"] = ["Engenheira", "Médico", "Designer", "Arquiteta"]
```

```
print(df)
```

**# Removendo uma coluna**

```
df = df.drop("Profissão", axis=1)
```

```
print(df)
```



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

## Operações com dados

**# Média das idades**

```
print(df["Idade"].mean())
```

**# Soma das idades**

```
print(df["Idade"].sum())
```

**# Criando uma nova coluna com a idade dobrada**

```
df["Idade x2"] = df["Idade"] * 2
```

```
print(df)
```



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

## Ordenando dados

Você pode ordenar o DataFrame com base em uma coluna:

```
# Ordenando pela coluna "Idade"
```

```
df_sorted = df.sort_values(by="Idade", ascending=False)
```

```
print(df_sorted)
```



## Lendo e Gravando arquivos

Após criar um DataFrame, você pode usar vários métodos para visualizar os dados.

- ❑ Lendo um arquivo CSV:

```
df = pd.read_csv("dados.csv")
```

```
print(df)
```

- ❑ Escrevendo um Dataframe em um arquivo CSV:

```
df.to_csv("dados_modificados.csv", index=False)
```



**UNIFEI**

Universidade Federal de Itajubá

Projeto e Implementação de Interface  
Gráfica com Python

*Manipulação de Dataframe*

## Conclusão

- ❑ DataFrame é uma estrutura poderosa e flexível para trabalhar com dados tabulares em Python.
- ❑ O Pandas fornece diversas funcionalidades para filtrar, modificar, e analisar dados de forma eficiente.
- ❑ Algumas das operações básicas que vimos incluem: criação, acesso, modificação, filtragem, ordenação e leitura/gravação de arquivos.