

```
1 import pandas as pd

1 nome = ['Douglas', 'Daniela', 'Pedro', 'Maria', 'Eduardo', 'Ester']
2 idade = [45, 7, 65, 64, 42, 37]
3 altura = [1.85, 1.23, 1.75, 1.67, 1.82, 1.73]
4 peso = [70, 22, 87, 64, 96, 68]

1 df = pd.DataFrame(zip(nome, idade, altura, peso), columns=['nome', 'idade', 'altura', 'peso'])
2 display(df)
```



	nome	idade	altura	peso
0	Douglas	45	1.85	70
1	Daniela	7	1.23	22
2	Pedro	65	1.75	87
3	Maria	64	1.67	64
4	Eduardo	42	1.82	96
5	Ester	37	1.73	68

```
1 '''
2 Imagine que você deseja exibir uma subtabela como apenas nome e idade
3 dos elementos com 45 anos ou mais
4 '''
5
6 filtro_01 = df.loc[df['idade']>=45, ['nome', 'idade']]
7 display(filtro_01)
```

	nome	idade
0	Douglas	45
2	Pedro	65
3	Maria	64

```
1 '''
2 Agora vamos supor que nosso objetivo seja inserir uma nova coluna com
3 o sexo dos elementos
4 '''
5 sexo = ['M', 'F', 'M', 'F', 'M', 'F']
6 df['sexo'] = sexo
7 display(df)
```

	nome	idade	altura	peso	sexo
0	Douglas	45	1.85	70	M
1	Daniela	7	1.23	22	F
2	Pedro	65	1.75	87	M
3	Maria	64	1.67	64	F
4	Eduardo	42	1.82	96	M
5	Ester	37	1.73	68	F

```
1 '''
2 Agora vamos supor que nosso desejo é inserir um novo elemento (linha ou registro)
3
4 '''
5 df.loc[6] = ['Tobias',18,1.67,65,'M']
6 display(df)
```

	nome	idade	altura	peso	sexo
0	Douglas	45	1.85	70	M
1	Daniela	7	1.23	22	F
2	Pedro	65	1.75	87	M
3	Maria	64	1.67	64	F
4	Eduardo	42	1.82	96	M
5	Ester	37	1.73	68	F
6	Tobias	18	1.67	65	M

```
1 '''
2 Agora vamos mudar o nome das colunas peso (massa) e sexo (genero)
3 '''
4 df.rename(columns={'peso':'massa','sexo':'genero'},inplace=True)
```

```
1 display(df)
```

	nome	idade	altura	massa	genero
0	Douglas	45	1.85	70	M
1	Daniela	7	1.23	22	F
2	Pedro	65	1.75	87	M
3	Maria	64	1.67	64	F
4	Eduardo	42	1.82	96	M
5	Ester	37	1.73	68	F
6	Tobias	18	1.67	65	M

```
1 '''
2 Agora vamos exibir as medidas descritivas dos atributos numéricos do
3 nosso dataframe
4 '''
5 medidas = df.describe()
6 display(medidas)
```

	idade	altura	massa
count	7.000000	7.000000	7.000000
mean	39.714286	1.674286	67.428571
std	21.646511	0.207514	23.408383
min	7.000000	1.230000	22.000000
25%	27.500000	1.670000	64.500000
50%	42.000000	1.730000	68.000000
75%	54.500000	1.785000	78.500000
max	65.000000	1.850000	96.000000

```
1 '''
2 Agora vamos supor que nosso objetivo seja
3 fazer um filtro com todos os dados do elemento Douglas
4 '''
5 doug = df.loc[0]
6 print(doug)
```

```
nome      Douglas
idade      45
altura    1.85
massa      70
genero     M
Name: 0, dtype: object
```

```
1 '''
2 No bloco anterior o filtro gerou um dataframe como resposta
3 agora vamos refazer o filtro, mas armazená-lo em uma lista
4 '''
5 doug = list(df.loc[0])
6 print(doug)
```

```
['Douglas', 45, 1.85, 70, 'M']
```

```
1 '''
2 Imagine que você deseja criar uma lista com todas as idades
3 '''
4 idades = list(df['idade'])
5 print(idades)
```

```
[45, 7, 65, 64, 42, 37, 18]
```

```
1 '''
2 Desafio: Na tabela medidas temos a linha desvio padrão a partir dela
3 calcule uma linha de variância e a insira a nova linha como última da
4 tabela de medidas
5 '''
6 desvio = list(medidas.iloc[2])
7 print(desvio)
```

```
[21.646510771286643, 0.20751362548494198, 23.408382706893292]
```

```
1 '''
2 Correção do desafio
3 '''
4 desvio = list(medidas.iloc[2])
5 variancia = []
6 for i in range(len(desvio)):
7     variancia.append(desvio[i]**2)
8 print(variancia)
9 medidas.loc['variância'] = variancia
10 display(medidas)
```

[468.57142857142867, 0.04306190476190476, 547.952380952381]

	idade	altura	massa
count	7.000000	7.000000	7.000000
mean	39.714286	1.674286	67.428571
std	21.646511	0.207514	23.408383
min	7.000000	1.230000	22.000000
25%	27.500000	1.670000	64.500000
50%	42.000000	1.730000	68.000000
75%	54.500000	1.785000	78.500000
max	65.000000	1.850000	96.000000
variância	468.571429	0.043062	547.952381

```
1 '''
2 Agora você deverá converter idade do df em uma lista
3 chamada idades, altura do df em uma lista chamada alturas, e
4 massa do df em uma lista chamada massas, após isso,
5 você deverá calcular a mediana de cada lista, usando a biblioteca numpy
6 e depois inserir cada mediana em uma lista chamada mediana e por último
7 inserir a lista mediana no df medidas na última posição
8 '''
9
10 idades = list(df['idade'])
11 alturas = list(df['altura'])
12 massas = list(df['massa'])
13 import numpy as np
14 medi = np.median(idades)
15 meda = np.median(alturas)
16 medm = np.median(massas)
17 mediana = [medi, meda, medm]
18 medidas.loc['mediana'] = mediana
19 display(medidas)
```

	idade	altura	massa
count	7.000000	7.000000	7.000000
mean	39.714286	1.674286	67.428571
std	21.646511	0.207514	23.408383
min	7.000000	1.230000	22.000000
25%	27.500000	1.670000	64.500000
50%	42.000000	1.730000	68.000000
75%	54.500000	1.785000	78.500000
max	65.000000	1.850000	96.000000
variância	468.571429	0.043062	547.952381
mediana	42.000000	1.730000	68.000000

```

1 '''
2 Tarefa para 23/04/2024:
3
4 Criar as medidas:
5 1 - AIQ
6 2 - LI
7 3 - LS
8
9 e inserí-las nessa ordem no df medidas
10 '''

    '\nTarefa para 23/04/2024:\n\nCriar as medidas:\n1 - AIQ\n2 - LI\n3 - LS\n\ne inserí-las nessa ordem no df
    medidas\n'

1 # transformando a linha 25% (q1) da tabela medidas em uma lista
2 q1 = list(medidas.loc['25%'])
3 print(q1)

[27.5, 1.67, 64.5]

1 # transformando a linha 25% (q3) da tabela medidas em uma lista
2 q1 = list(medidas.iloc[4])
3 print(q1)

[27.5, 1.67, 64.5]

1 # transformando a linha 75% (q3) da tabela medidas em uma lista
2 q3 = list(medidas.loc['75%'])
3 print(q3)

[54.5, 1.7850000000000001, 78.5]

1 # transformando a linha 75% (q3) da tabela medidas em uma lista
2 q3 = list(medidas.iloc[6])
3 print(q3)

[54.5, 1.7850000000000001, 78.5]

1 aiq = []
2 for i in range(len(q1)):
3     aiq.append(q3[i]-q1[i])
4 print(aiq)

[27.0, 0.11500000000000002, 14.0]

1 # Inserindo a lista aiq como linha na tabela de medidas
2 medidas.loc['AIQ'] = aiq
3 display(medidas)

```

```

    idade  altura  massa
count  7.000000  7.000000  7.000000

1 li = []
2 for i in range(len(q1)):
3     li.append(q1[i]-1.5*aiq[i])
    min      7.000000  1.230000  22.000000
1 print(li)

[-13.0, 1.4974999999999996, 43.5]
-----
1 ls = []
2 for i in range(len(q3)):
3     ls.append(q3[i]+1.5*aiq[i])
4 print(ls)

[95.0, 1.9575000000000005, 99.5]
-----
1 medidas.loc['li'] = li
2 medidas.loc['ls'] = ls
3 display(medidas)
```

	idade	altura	massa
count	7.000000	7.000000	7.000000
mean	39.714286	1.674286	67.428571
std	21.646511	0.207514	23.408383
min	7.000000	1.230000	22.000000
25%	27.500000	1.670000	64.500000
50%	42.000000	1.730000	68.000000
75%	54.500000	1.785000	78.500000
max	65.000000	1.850000	96.000000