

```
In [1]: import pandas as pd
import seaborn as sns
```

Questão 4

O Python vem se tornando, cada vez mais, uma linguagem muito utilizada por cientistas de dados devido sua flexibilidade, facilidade em manipular, analisar e visualizar dados. Uma parte importante e fundamental do kit de ferramentas do cientista de dados é a visualização de dados. Embora seja muito fácil criar visualizações, é bem mais difícil produzir boas que as tornem de fácil interpretação.

Escreva um programa em Python que gere um gráfico de linha, onde anos no eixo x, e gdp no eixo y; years = [1950, 1960, 1970, 1980, 1990, 2000, 2010] gdp = [300.2, 543.3, 1075.9, 2862.5, 5979.6, 10289.7, 14958.3]

```
In [2]: quest_4_data = {"years": [1950, 1960, 1970, 1980, 1990, 2000, 2010],
                      "gdp": [300.2, 543.3, 1075.9, 2862.5, 5979.6, 10289.7, 14958.3]}
quest_4_df = pd.DataFrame(quest_4_data)
```

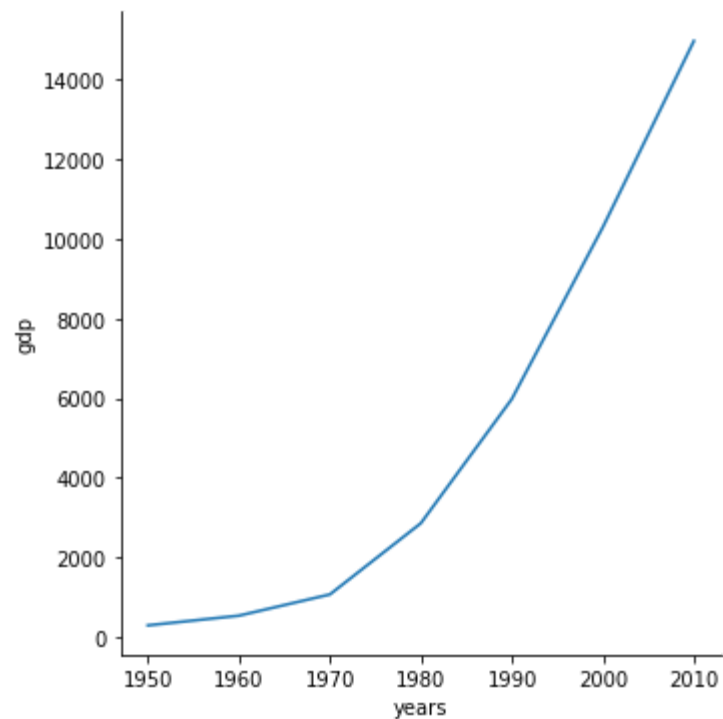
```
In [3]: quest_4_df.head()
```

Out[3]:

	years	gdp
0	1950	300.2
1	1960	543.3
2	1970	1075.9
3	1980	2862.5
4	1990	5979.6

```
In [5]: sns.relplot(x='years', y='gdp', kind='line', data = quest_4_df)
```

Out[5]: <seaborn.axisgrid.FacetGrid at 0x1f9b80084c0>



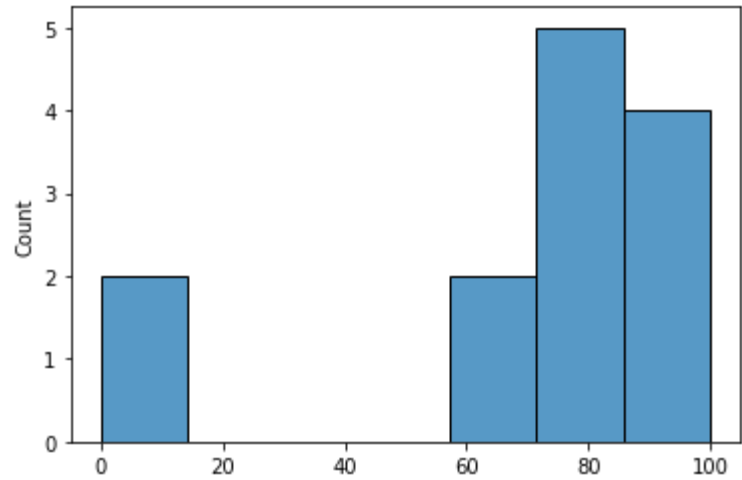
Questão 5

Gráficos de histogramas de valores numéricos carregados é uma boa escolha para explorar visualmente como os valores são distribuídos. Escreva um programa em Python que gere um histograma dos seguintes dados:

grades = [83,95,91,87,70,0,85,82,100,67,73,77,0]

```
In [9]: grades = [83,95,91,87,70,0,85,82,100,67,73,77,0]
sns.histplot(x = grades)
```

Out[9]: <AxesSubplot:ylabel='Count'>



Questão 6

Um gráfico de dispersão é a escolha certa para visualizar o relacionamento entre dois pares de conjuntos de dados. Dados os pares s de dados abaixo, escreva um programa em Python para gerar um gráfico de dispersão.

friends = [70, 65, 72, 63, 71, 64, 60, 64, 67]

minutes = [175, 170, 205, 120, 220, 130, 105, 145, 190]

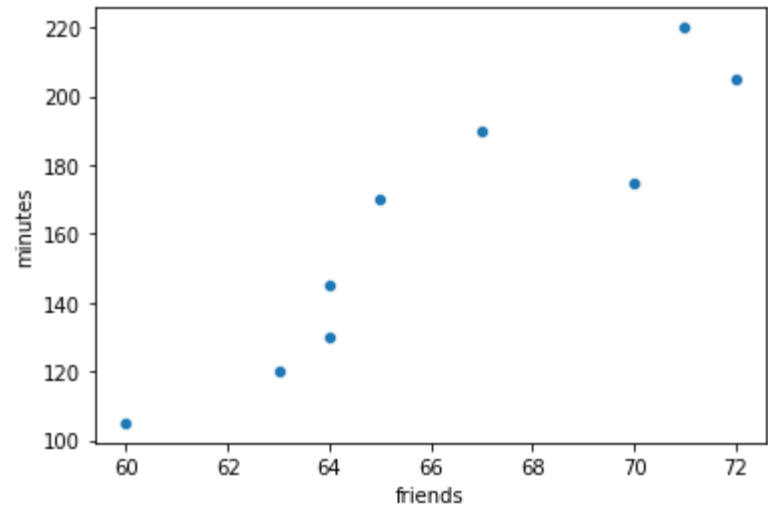
```
In [12]: quest_6_data = {"friends": [ 70, 65, 72, 63, 71, 64, 60, 64, 67],
                        "minutes": [175, 170, 205, 120, 220, 130, 105, 145, 190]}
quest_6_df = pd.DataFrame(quest_6_data)
quest_6_df.head()
```

Out[12]:

	friends	minutes
0	70	175
1	65	170
2	72	205
3	63	120
4	71	220

```
In [13]: sns.scatterplot(x = 'friends', y='minutes', data = quest_6_df)
```

Out[13]: <AxesSubplot:xlabel='friends', ylabel='minutes'>



In []: