



ELDORADO

Trilha Python

Aula 5

Foto de David Travis,
disponível na Unsplash.
Editada pelo autor.



Faísca

O que vimos

Na aula anterior...

Seleção de dados

Seleção é quando queremos escolher um subconjunto do dados.

```
[2]: lista_de_pessoas = pd.DataFrame(  
    [  
        ['Ana', 1.70, 21],  
        ['Pedro', 1.70, 22],  
        ['Julia', 1.60, 25],  
        ['João', 1.80, 26],  
        ['Paula', 1.80, 22],  
        ['Yasmim', 1.70, 24],  
        ['Luana', 1.75, 21],  
        ['Theo', 1.65, 21],  
    ],  
    columns=["Nome", "Altura", "Idade"])  
  
lista_de_pessoas
```

```
[2]:
```

	Nome	Altura	Idade
0	Ana	1.70	21
1	Pedro	1.70	22
2	Julia	1.60	25
3	João	1.80	26
4	Paula	1.80	22
5	Yasmim	1.70	24
6	Luana	1.75	21
7	Theo	1.65	21

```
[3]: lista_de_pessoas['Altura']
```

```
[3]:
```

0	1.70
1	1.70
2	1.60
3	1.80
4	1.80
5	1.70
6	1.75
7	1.65

Name: Altura, dtype: float64

```
[4]: lista_de_pessoas[["Nome", "Altura"]]
```

```
[4]:
```

	Nome	Altura
0	Ana	1.70
1	Pedro	1.70
2	Julia	1.60
3	João	1.80
4	Paula	1.80
5	Yasmim	1.70
6	Luana	1.75
7	Theo	1.65

```
[5]: lista_de_pessoas.iloc[2]
```

```
[5]:
```

Nome	Julia
Altura	1.6
Idade	25

Name: 2, dtype: object

```
[6]: lista_de_pessoas.iloc[[0,2]]
```

```
[6]:
```

	Nome	Altura	Idade
0	Ana	1.7	21
2	Julia	1.6	25

Filtragem de dados

Já filtragem é quando aplicamos algum(s) critério(s) para extrair apenas um subconjunto de dados.

No exemplo da biblioteca, eu poderia ter interesse apenas em ver as informações dos livros do gênero ficção científica, por exemplo.

```
[2]: lista_de_pessoas = pd.DataFrame(  
    [  
        ['Ana', 1.70, 21],  
        ['Pedro', 1.70, 22],  
        ['Julia', 1.60, 25],  
        ['João', 1.80, 26],  
        ['Paula', 1.80, 22],  
        ['Yasmim', 1.70, 24],  
        ['Luana', 1.75, 21],  
        ['Theo', 1.65, 21],  
    ],  
    columns=["Nome", "Altura", "Idade"])  
  
lista_de_pessoas
```

```
[2]:
```

	Nome	Altura	Idade
0	Ana	1.70	21
1	Pedro	1.70	22
2	Julia	1.60	25
3	João	1.80	26
4	Paula	1.80	22
5	Yasmim	1.70	24
6	Luana	1.75	21
7	Theo	1.65	21

```
[7]: lista_de_pessoas['Altura'] > 1.65
```

```
[7]: 0    True  
     1    True  
     2   False  
     3    True  
     4    True  
     5    True  
     6    True  
     7   False  
     Name: Altura, dtype: bool
```

```
[8]: lista_de_pessoas[lista_de_pessoas['Altura'] > 1.65]
```

```
[8]:
```

	Nome	Altura	Idade
0	Ana	1.70	21
1	Pedro	1.70	22
3	João	1.80	26
4	Paula	1.80	22
5	Yasmim	1.70	24
6	Luana	1.75	21

```
[9]: lista_de_pessoas[ (lista_de_pessoas['Altura'] > 1.65) & (lista_de_pessoas['Altura'] < 1.75) ]
```

```
[9]:
```

	Nome	Altura	Idade
0	Ana	1.7	21
1	Pedro	1.7	22
5	Yasmim	1.7	24

```
[10]: lista_de_pessoas[ (lista_de_pessoas['Altura'] == 1.60) | (lista_de_pessoas['Altura'] == 1.70) ]
```

```
[10]:
```

	Nome	Altura	Idade
0	Ana	1.7	21
1	Pedro	1.7	22
2	Julia	1.6	25
5	Yasmim	1.7	24

Ordenação

```
[14]: lista_de_pessoas = pd.DataFrame(  
    [  
        ['Ana', 1.70, 21],  
        ['Pedro', 1.70, 22],  
        ['Julia', 1.60, 25],  
        ['João', 1.80, 26],  
        ['Paula', 1.80, 22],  
        ['Yasmim', 1.70, 24],  
        ['Luana', 1.75, None],  
        ['Theo', 1.65, 21],  
    ],  
    columns=["Nome", "Altura", "Idade"])  
  
lista_de_pessoas
```

```
[14]:
```

	Nome	Altura	Idade
0	Ana	1.70	21.0
1	Pedro	1.70	22.0
2	Julia	1.60	25.0
3	João	1.80	26.0
4	Paula	1.80	22.0
5	Yasmim	1.70	24.0
6	Luana	1.75	NaN
7	Theo	1.65	21.0

Ordenar os dados de forma ascendente quanto descendente.

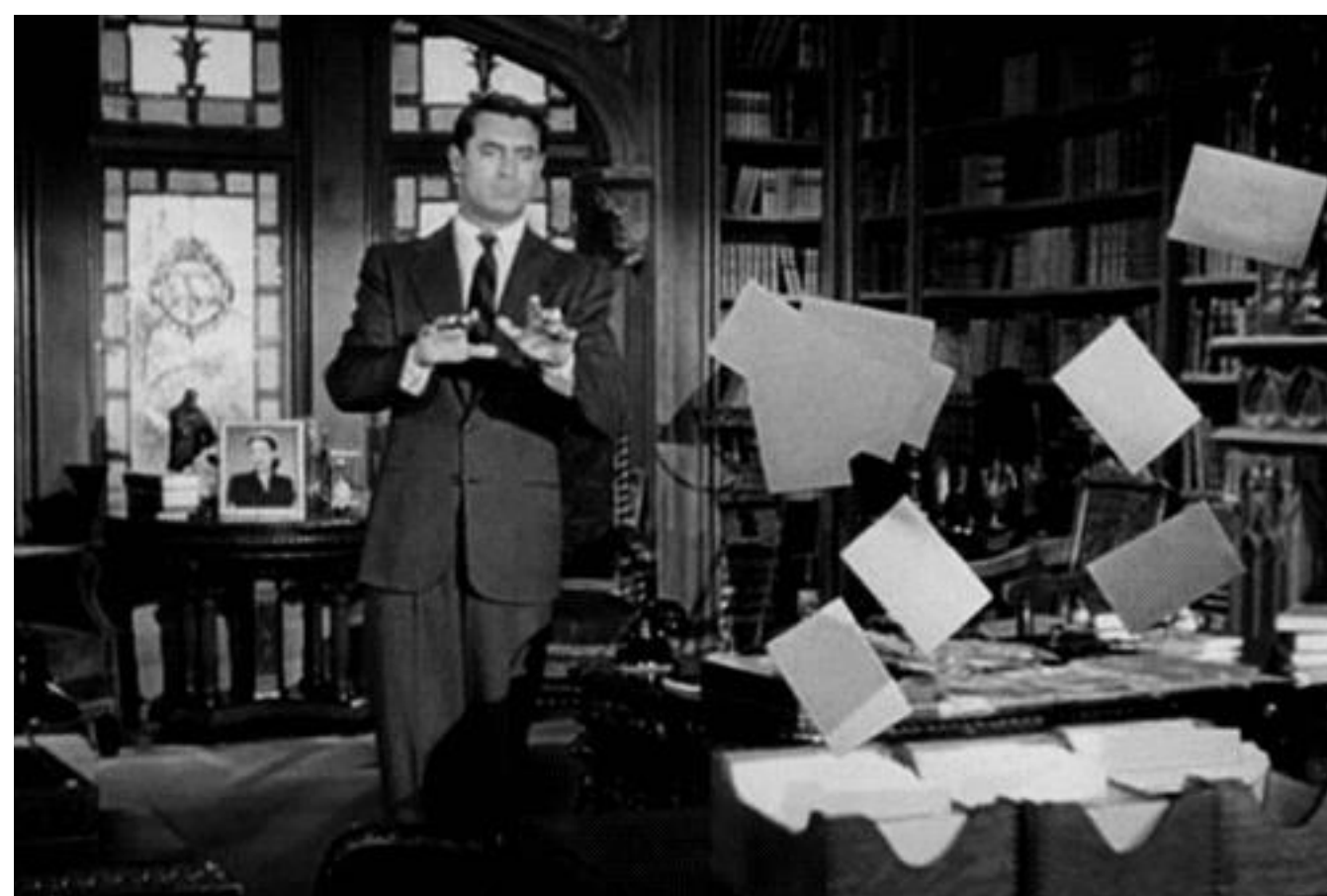
```
[15]: lista_de_pessoas.sort_values(by=['Idade'], ascending=False, na_position="first")
```

```
[15]:
```

	Nome	Altura	Idade
6	Luana	1.75	NaN
3	João	1.80	26.0
2	Julia	1.60	25.0
5	Yasmim	1.70	24.0
1	Pedro	1.70	22.0
4	Paula	1.80	22.0
0	Ana	1.70	21.0
7	Theo	1.65	21.0

Ordenação

Outro parâmetro interessante na_position, que coloca todos os valores nulos em 1º.



```
[14]: lista_de_pessoas = pd.DataFrame(  
    [  
        ['Ana', 1.70, 21],  
        ['Pedro', 1.70, 22],  
        ['Julia', 1.60, 25],  
        ['João', 1.80, 26],  
        ['Paula', 1.80, 22],  
        ['Yasmim', 1.70, 24],  
        ['Luana', 1.75, None],  
        ['Theo', 1.65, 21],  
    ],  
    columns=["Nome", "Altura", "Idade"])  
  
lista_de_pessoas
```

```
[14]:
```

	Nome	Altura	Idade
0	Ana	1.70	21.0
1	Pedro	1.70	22.0
2	Julia	1.60	25.0
3	João	1.80	26.0
4	Paula	1.80	22.0
5	Yasmim	1.70	24.0
6	Luana	1.75	NaN
7	Theo	1.65	21.0

```
[15]: lista_de_pessoas.sort_values(by=['Idade'], ascending=False, na_position="first")
```

```
[15]:
```

	Nome	Altura	Idade
6	Luana	1.75	NaN
3	João	1.80	26.0
2	Julia	1.60	25.0
5	Yasmim	1.70	24.0
1	Pedro	1.70	22.0
4	Paula	1.80	22.0
0	Ana	1.70	21.0
7	Theo	1.65	21.0



Foto de Gaelle Marcel,
disponível na Unsplash.
Editada pelo autor.

Mão Na Massa

Exercícios 50Xp

- ✦ Exercício 09
- ✦ Use 3 dataframes criados na aula anterior, mostre apenas o filme/série/anime com maior pontuação.
- ✦ Leia o arquivo do dataset em json.
- ✦ Selecione apenas as 3 primeiras colunas do dataset
- ✦ Selecione apenas as 10 primeiras linhas do dataset



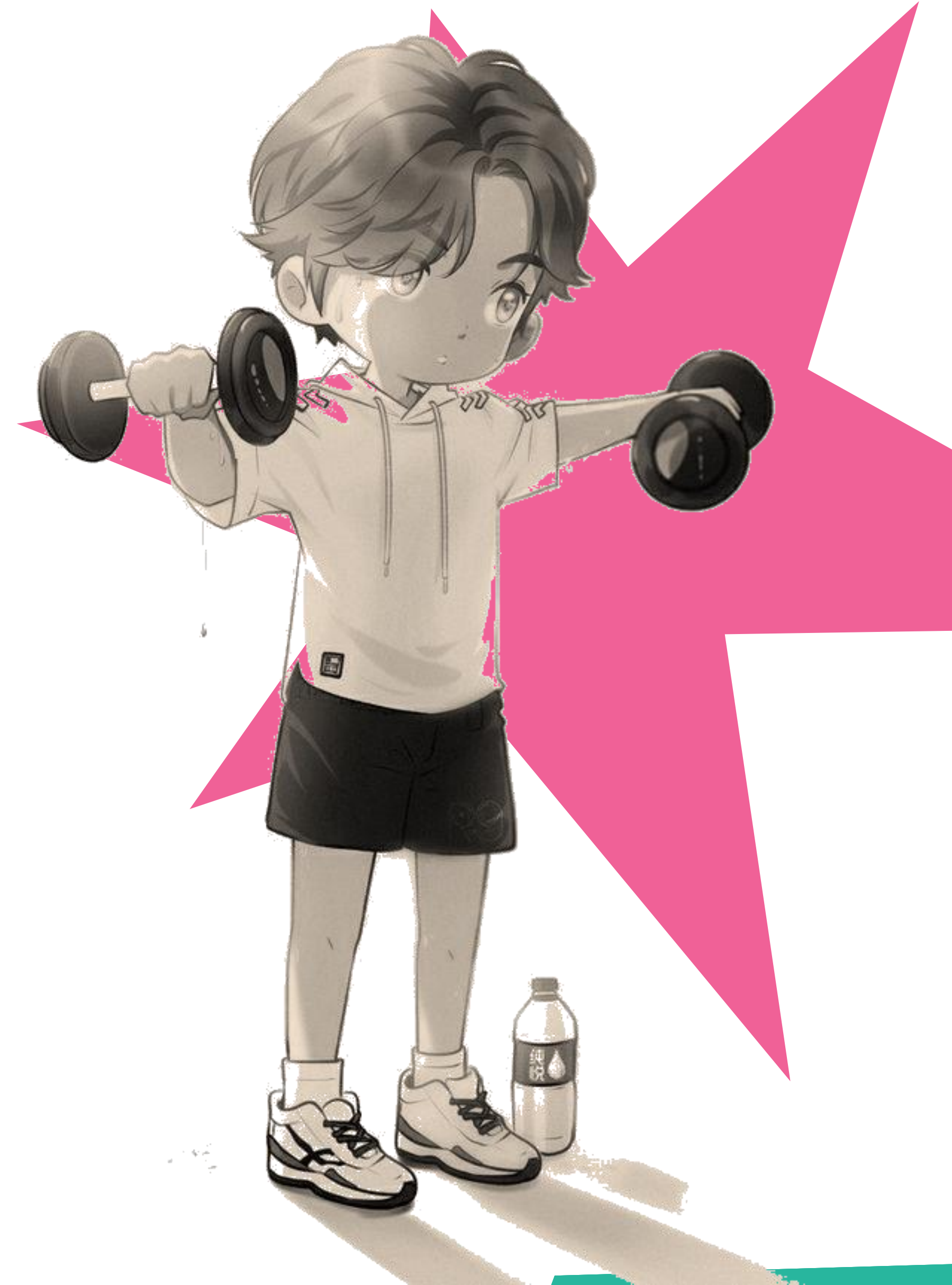
Exercícios 50Xp

- ✦ Exercício 09
- ✦ Filtre o dataframe pegando apenas as linhas com valores maiores que essa média.
- ✦ Escolha uma coluna com valores textuais e filtre o dataframe pelo 1º valor
- ✦ Escolha uma coluna, como ano, titulo, etc e ordene seu dataframe tanto de forma crescente quanto decrescente



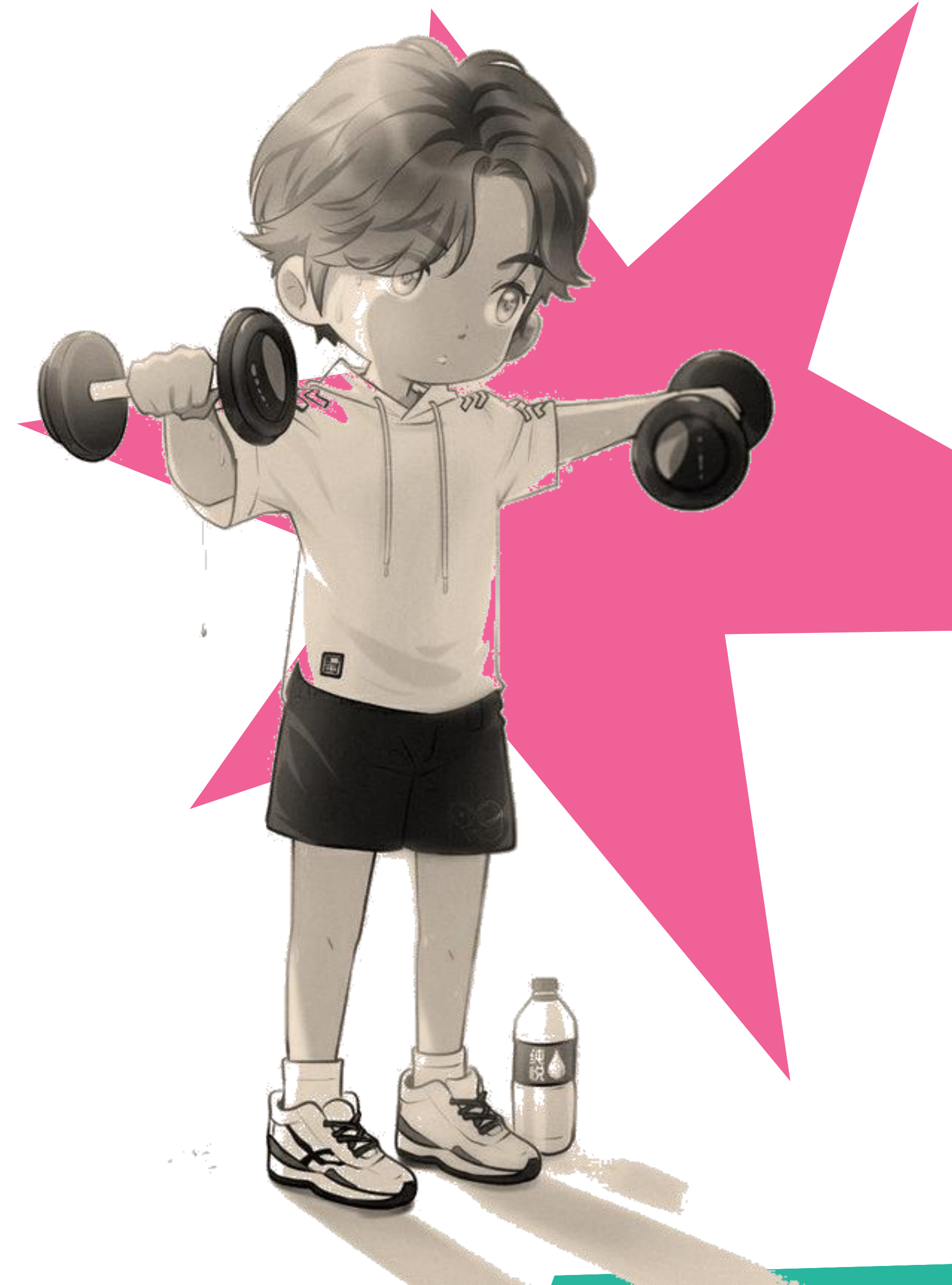
Exercícios 50Xp

- ✦ Exercício 10 Coop - Responda as Perguntas
- ✦ O que é uma **chave primária** em um banco de dados? Por que é importante que cada tabela tenha uma chave primária?
- ✦ Em um dataframe do Pandas, como você pode identificar uma coluna que atua como uma chave primária?



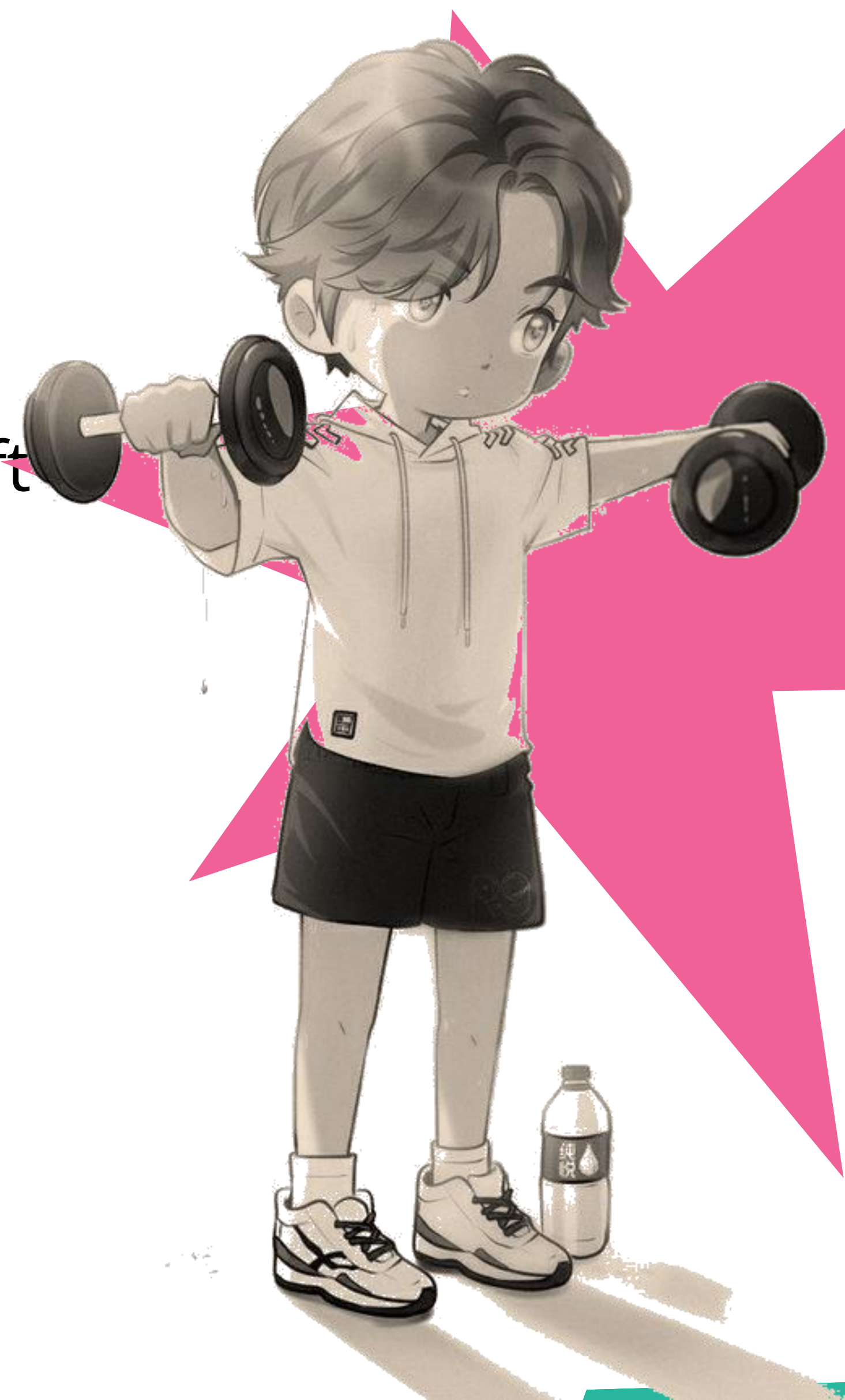
Exercícios 50Xp

- ✦ Exercício 10 Coop - Responda as Perguntas
- ✦ O que são chaves **estrangeiras** e como elas se relacionam com as chaves **primárias**?
- ✦ Explique como a relação entre duas tabelas com chaves primárias e estrangeiras pode ser representada com dataframes no Pandas.
- ✦ Qual a diferença entre os métodos `.merge()` e `.join()` no Pandas?



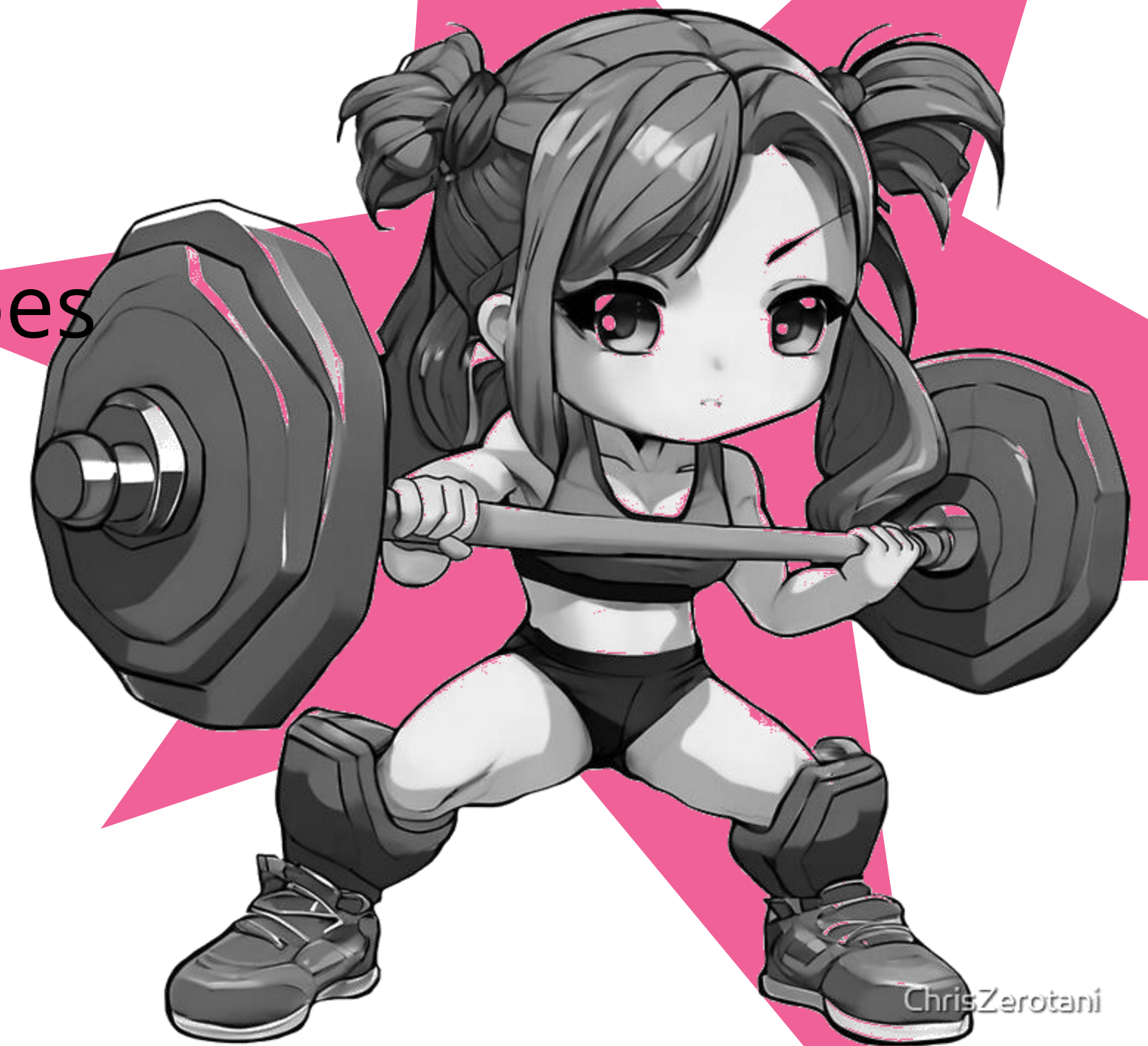
Exercícios 50Xp

- ✦ Exercício 10 Coop - Responda as Perguntas
- ✦ Quando você usaria um inner join ao invés de um left join? Explique com exemplos.
- ✦ Em que situação um merge resultaria em valores faltantes? Como podemos lidar com esses valores?
- ✦ Quais são as maneiras de tratar dados faltantes no Pandas? Explique os métodos `.dropna()` e `.fillna()`.



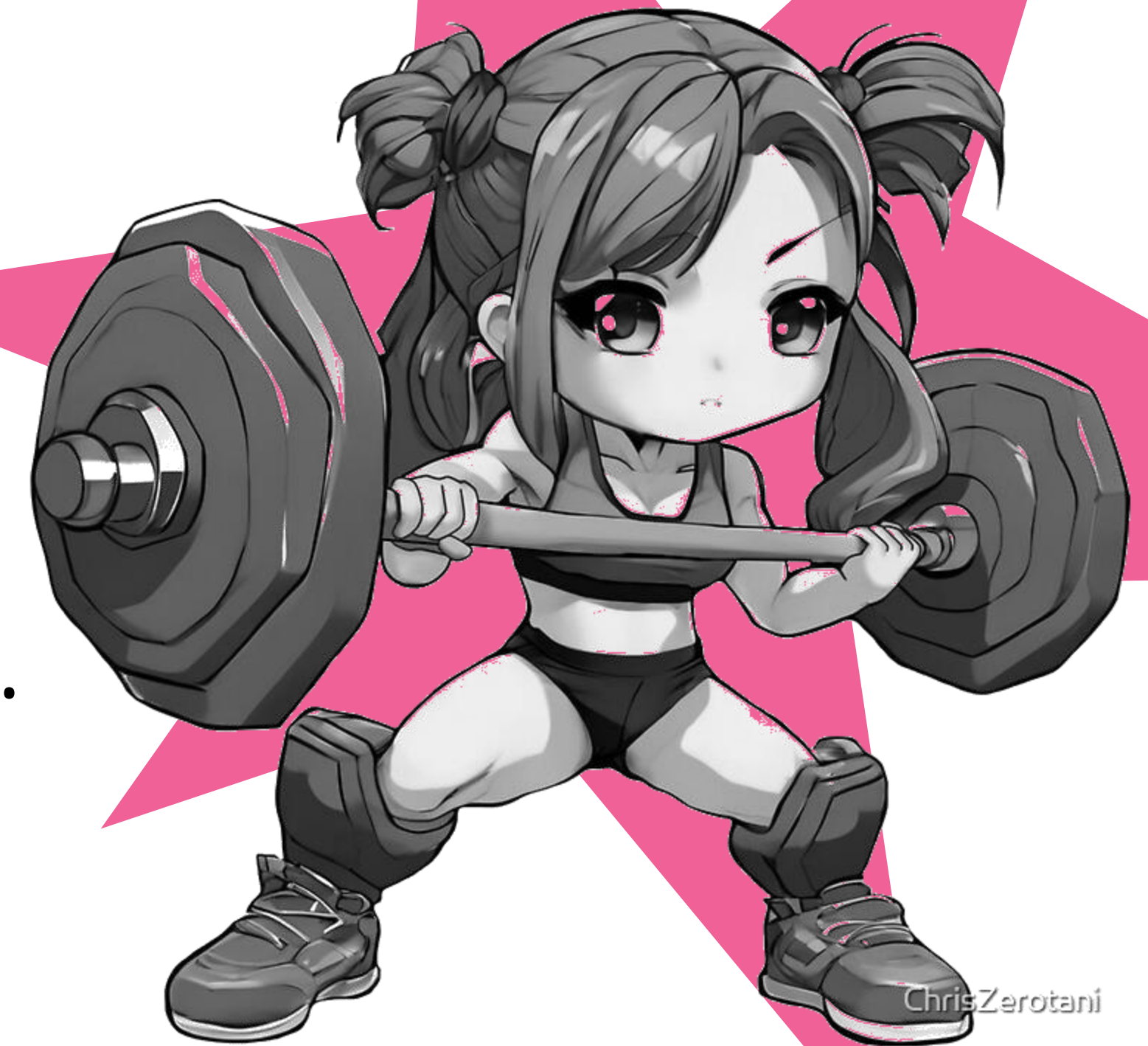
Exercícios 50Xp

- ✦ Exercício 11 – Mini Boss - Coop
- ✦ Crie e exiba um dataframe **musicas** com informações sobre suas músicas favoritas (5 músicas):
- ✦ **id_musica** – identificador da música.
- ✦ **nome_musica** – nome da música.
- ✦ **artista** – nome do artista ou banda
- ✦ **genero** – gênero musical (Pop, Rock, Jazz, etc.)
- ✦ **duracao** – duração da música em minutos.



Exercícios 50Xp

- ✦ Exercício 11 – Mini Boss
- ✦ Crie e exiba um dataframe **reproducoes** dessas musicas:
- ✦ **id_reproducao** – identificador único da reprodução.
- ✦ **id_musica** – identificador da música reproduzida.
- ✦ **data_reproducao** – data da reprodução.
- ✦ **duracao_reproducao** – duração da reprodução (caso o usuário não ouça a música completa).



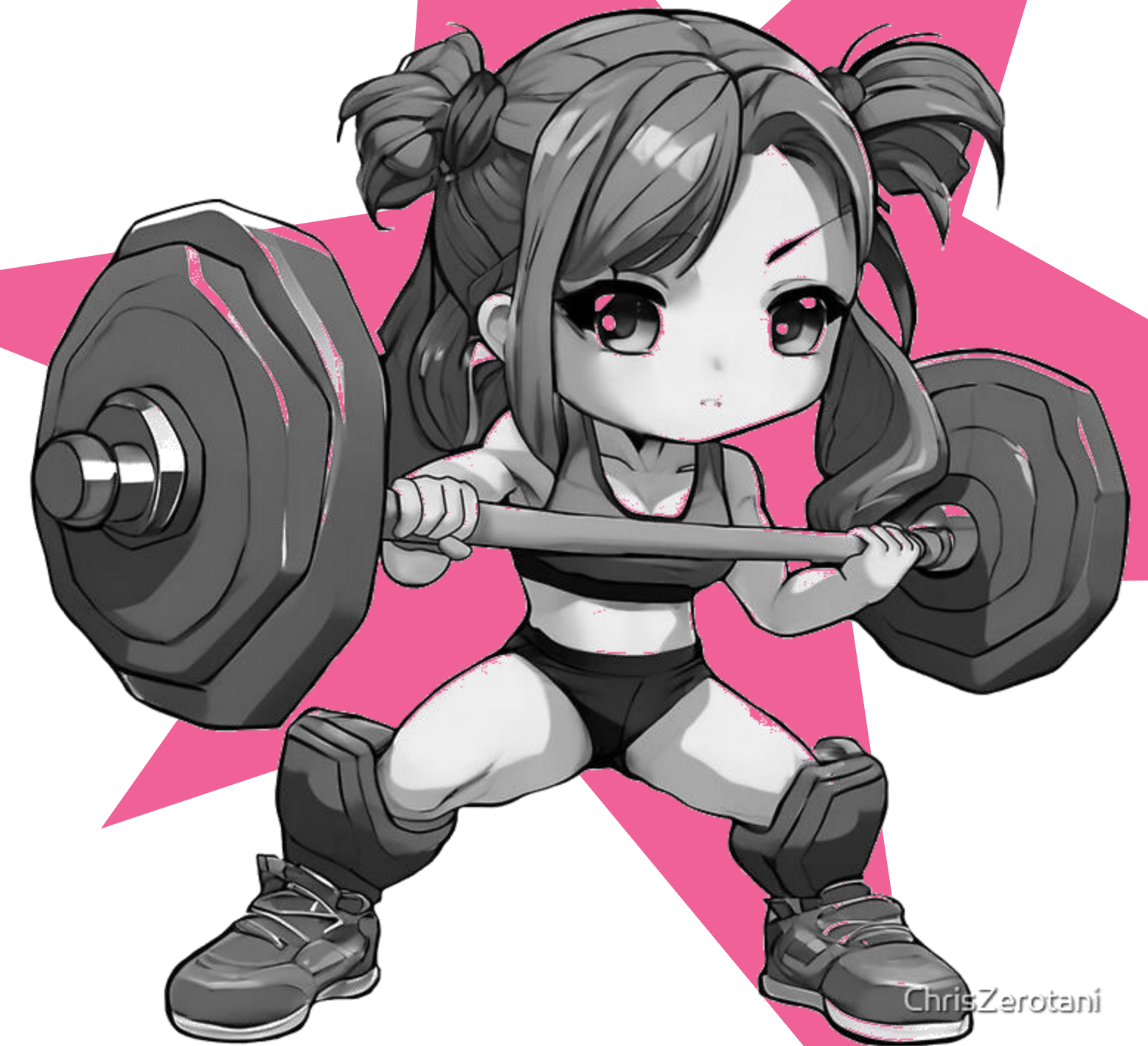
Exercícios 50Xp

- ✦ Exercício 11 – Mini Boss
- ✦ Simule 5 reproduções de diferentes músicas, associando o `id_musica` com a data e a duração reproduzida.
- ✦ Realize um merge entre os dataframes `musicas` e `reproducoes`, combinando-os pela coluna `id_musica`.
- ✦ Exiba o nome da música, o artista e a duração



Exercícios 50Xp

- ✦ Exercício 11 – Mini Boss
- ✦ Atualize o dataframe de reproducoes, removendo um registro para simular que nem todas as músicas foram reproduzidas.
- ✦ Realize o merge novamente e identifique os valores faltantes.
- ✦ Preencha os valores faltantes com "Não Reproduzida"..



Exercícios 50Xp

- ✦ Exercício 11 – Mini Boss
- ✦ Análise os dados
- ✦ Qual o gênero de música mais reproduzido? Use agrupamento para calcular isso.
- ✦ Qual o total de minutos de música reproduzidos?
- ✦ Crie uma nova coluna chamada avaliacao e atribua uma classificação de 1 a 5 para cada música com base em quanto você gosta dela.



Até a próxima aula...

02/10/23