

```
In [1]: #Importanto as bibliotecas
import pyarrow as pa
import pandas as pd
from pyarrow import feather
import numpy as np
import pyarrow.parquet
```

```
In [2]: #Abra o arquivo parquet usando a função parquet_dataset:
dataset = pa.parquet.ParquetDataset('C:\\Users\\maced\\Desktop\\Confitec\\OriginaisNetflix - Python.parquet')
```

```
In [3]: #Ler o dataset
table = dataset.read()
```

```
In [4]: #Converta a tabela PyArrow para um DataFrame do pandas:
df = table.to_pandas()
```

```
In [5]: # A partir deste ponto, você pode manipular os dados no DataFrame df de acordo com suas necessidades.
# Salvar as mudanças no DataFrame em um novo arquivo parquet:
table = pa.Table.from_pandas(df)
```

```
In [6]: with pa.parquet.ParquetWriter('C:\\Users\\maced\\Desktop\\Confitec\\OriginaisNetflix - Python.parquet', table.schema) as writer:
        writer.write_table(table)
```

```
In [7]: # Para transformar os campos "Premiere" e "dt_inclusao" de string para datetime em um DataFrame do pandas,
# você pode usar a função pd.to_datetime().
# Essa função converte uma série de valores em formato de data/hora para o tipo datetime.

df['Premiere'] = pd.to_datetime(df['Premiere'])
```

```
In [8]: # Assumindo que você tenha um DataFrame chamado df com as colunas "Premiere" e "dt_inclusao" como strings,
# você pode converter esses campos para datetime da seguinte maneira:

# Convertendo a coluna "dt_inclusao" para datetime
df['dt_inclusao'] = pd.to_datetime(df['dt_inclusao'])

#Agora, as colunas "Premiere" e "dt_inclusao" do DataFrame df estão no formato datetime.
```

```
In [9]: print(df.keys())

Index(['Title', 'Genre', 'GenreLabels', 'Premiere', 'Seasons', 'SeasonsParsed',
      'EpisodesParsed', 'Length', 'MinLength', 'MaxLength', 'Status',
      'Active', 'Table', 'Language', 'dt_inclusao'],
      dtype='object')
```

```
In [10]: # Para ordenar os dados por ativos e gênero de forma decrescente, primeiro você precisa converter a coluna "ativos" em um
# tipo numérico, onde 1 representa ativo e 0 representa inativo. Em seguida, você pode ordenar o DataFrame com a
# função sort_values() do pandas.

# Assumindo que você tenha um DataFrame chamado df com as colunas "ativos" e "gênero",
#você pode ordenar os dados da seguinte maneira:

# Converter a coluna "ativos" para numérico
df['Active'] = pd.to_numeric(df['Active'])
```

```
In [11]: # Ordenar o DataFrame por ativos e gênero de forma decrescente
df = df.sort_values(['Active', 'Genre'], ascending=[False, False])
```

```
In [12]: #Neste exemplo, a função pd.to_numeric() é usada para converter a coluna "ativos" de uma string para um tipo numérico.
#Em seguida, a função sort_values() é usada para ordenar o DataFrame em ordem decrescente por ativos e gênero.
```

```
#A opção ascending=[False, False] é usada para ordenar a coluna "ativos" em ordem decrescente
#(com os valores 1 aparecendo primeiro) e a coluna "gênero" em ordem alfabética decrescente.
# Com este código, o DataFrame df estará ordenado por ativos e gênero de forma decrescente,
# com os ativos com valor 1 aparecendo primeiro.
```

```
In [13]: # Para remover linhas duplicadas e trocar o resultado das linhas que tiverem a coluna "Seasons" de "TBA"
# para "a ser anunciado", você pode usar as funções drop_duplicates() e replace() do pandas.
```

```
# Remover linhas duplicadas
df = df.drop_duplicates()
```

```
# Substituir "TBA" por "a ser anunciado" na coluna "Seasons"
df['Seasons'] = df['Seasons'].replace('TBA', 'a ser anunciado')
```

```
In [14]: #Neste exemplo, a função drop_duplicates() é usada para remover linhas duplicadas do DataFrame df.
#Em seguida, a função replace() é usada para substituir todos os valores "TBA" na coluna "Seasons"por "a ser anunciado".
#Desta forma o DataFrame df terá as linhas duplicadas removidas e os valores "TBA" na coluna "Seasons"
#substituídos por "a ser anunciado".
```

```
print(df)
```

```

Title
176 Dance & Sing with True chrildrens musical short
177 Super Monsters Monster Party chrildrens musical short
178 True Tunes chrildrens musical short
131 Word Party childrens-animation
133 Ask the StoryBots childrens-animation
..
318 Chasing Cameron Celebrity
12 A Series of Unfortunate Events Black-comedy mystery
254 Abstract: The Art of Design Biography
263 The Day I Met El Chapo Biography
200 Blazing Transfer Students Action comedy

GenreLabels Premiere Seasons \
176 chrildrens,musical,short 2018-05-18 1 season, 11 episodes
177 chrildrens,musical,short 2018-09-14 1 season, 4 episodes
178 chrildrens,musical,short 2019-07-12 a ser anunciado
131 childrens-animation 2016-07-08 3 seasons, 40 episodes
133 childrens-animation 2016-08-12 2 seasons, 14 episodes
..
318 celebrity 2016-12-27 1 season, 10 episodes
12 black-comedy,mystery 2017-01-13 3 seasons, 25 episodes
254 biography 2017-02-10 1 season, 8 episodes
263 biography 2017-10-20 1 season, 3 episodes
200 action,comedy 2017-11-10 1 season, 8 episodes

SeasonsParsed EpisodesParsed Length MinLength MaxLength Status \
176 1 11 2 min. 2 2 Pending
177 1 4 2 min. 2 2 Pending
178 0 0 TBA 0 0 Pending
131 3 40 13 min. 13 13 Renewed
133 2 14 20-28 min. 20 28 Renewed
..
318 1 10 26-33 min. 26 33 Ended
12 3 25 36-64 min. 36 64 Ended
254 1 8 41-48 min. 41 48 Ended
263 1 3 48-61 min. 48 61 Ended
200 1 8 23-28 min. 23 28 Ended

Active Table Language dt_inclusao
176 1 Children English 2021-03-16 21:20:24.167000-03:00
177 1 Children English 2021-03-16 21:20:24.167000-03:00
178 1 Children English 2021-03-16 21:20:24.167000-03:00
131 1 Children English 2021-03-16 21:20:24.167000-03:00
133 1 Children English 2021-03-16 21:20:24.167000-03:00
..
318 0 Reality English 2021-03-16 21:20:24.167000-03:00
12 0 Drama English 2021-03-16 21:20:24.167000-03:00
254 0 DocuSeries English 2021-03-16 21:20:24.167000-03:00
263 0 DocuSeries English 2021-03-16 21:20:24.167000-03:00
200 0 ForeignLanguage Japanese 2021-03-16 21:20:24.167000-03:00
```

[358 rows x 15 columns]

In []: