V2_selective_subsets

August 17, 2025

1 Activity: Selective Subsets

1.1 Introduction

In this activity you will practice selecting subsets of data from a DataFrame using Pandas. This activity will cover the following topics: - Creating masks - Negating masks - Masks with slicing - Null value masks

Question 1 Create a `DataFrame` called `df` from the given CSV file `movie_data.csv`, and then create a mask called `before_millennium` to select all movies that were released before 2000.

[7]:
```
import pandas as pd
df = pd.read_csv("movie_data.csv")
before_millennium = df["Year Released"] < 2000
print(df[before_millennium].head())
```

```
                     Title  Year Released  Rating  Box Office ($M)
0  The Shawshank Redemption           1994     9.3             58.3
1              The Godfather          1972     9.2            246.1
3               Pulp Fiction          1994     8.9            213.9
4          Schindler's List          1993     8.9            321.3
5                Fight Club          1999     8.8            100.9
```

[ ]:
```
# Question 1 Grading Checks

assert isinstance(df, pd.DataFrame), 'Did you create a DataFrame called df?'
```

Question 2 Using the `before_millennium` mask from Question 1, assign the titles of every movie that was released after 2000 to a `Series` called `newer_titles`.

[8]:
```
newer_titles = df.loc[~before_millennium, "Title"]
print(newer_titles.head())
```

```
2                                    The Dark Knight
7                                          Inception
9     The Lord of the Rings: The Fellowship of the Ring
11                                          Gladiator
```

```
14                The Lord of the Rings: The Two Towers
Name: Title, dtype: object
```

[ ]:
```
# Question 2 Grading Checks

assert isinstance(newer_titles, pd.Series), 'Did you create a Series called␣
↪newer_titles?'
```

Question 3 Create a mask to select movies with a `Rating` of `8.9` and a `Box Office ($M)` value higher than `1000.0`. Assign the resulting `Series` to a variable called `popular_pg_movies`.

[15]:
```
import pandas as pd
df = pd.read_csv("movie_data.csv")
popular_pg_movies = (df['Rating'] == 8.9) & (df['Box Office ($M)'] > 1000.0)
print(popular_pg_movies)
```

```
0     False
1     False
2     False
3     False
4     False
      …
70    False
71    False
72    False
73    False
74    False
Length: 75, dtype: bool
```

[ ]:
```
# Question 3 Grading Checks

assert isinstance(popular_pg_movies, pd.Series), 'Did you create a Series␣
↪called popular_pg_movies?'
```

Question 4 Create a mask to select movies with a null value for `Box Office ($M)` or `Rating`. Assign the resulting `Series` to a variable called `missing_info`.

[4]:
```
import pandas as pd
df = pd.read_csv("movie_data.csv")
missing_info = df["Box Office ($M)"].isnull() | df["Rating"].isnull()
print(missing_info)
```

```
0     False
1     False
2     False
3     False
```

```
4     False
      …
70    False
71    False
72    False
73    False
74    False
Length: 75, dtype: bool
```

[6]:
```
# Question 4 Grading Checks

assert isinstance(missing_info, pd.Series), 'Did you create a Series called␣
↪missing_info?'
```