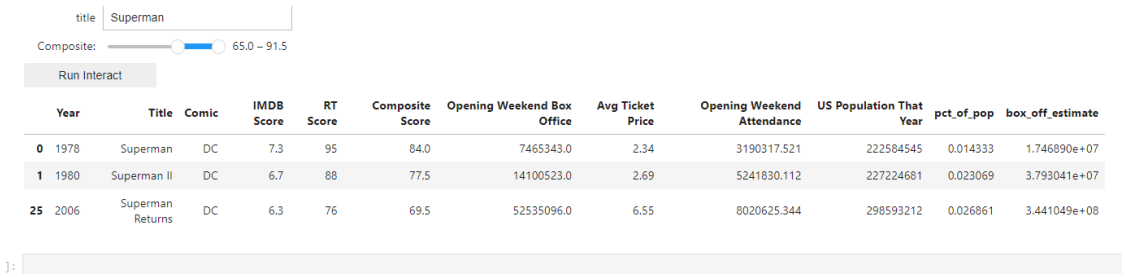
Practice-Superhero-Movies

October 7, 2024

1 Name: Ramya Chowdary Patchala

1.1 PRACTICE - Superhero Movies

- UI to search for a title
  - use the `.str` property on the Series to access the string methods!
- Select a range for the Composite score based on the movie.
  - use the FloatRangeSlider widget https://ipywidgets.readthedocs.io/en/latest/examples/Widget%20List
- Output the Results using `display()`



	Year	Title	Comic	IMDB Score	RT Score	Composite Score	Opening Weekend Box Office	Avg Ticket Price	Opening Weekend Attendance	US Population That Year	pct_of_pop	box_off_estimate
0	1978	Superman	DC	7.3	95	84.0	7465343.0	2.34	3190317.521	222584545	0.014333	1.746890e+07
1	1980	Superman II	DC	6.7	88	77.5	14100523.0	2.69	5241830.112	227224681	0.023069	3.793041e+07
25	2006	Superman Returns	DC	6.3	76	69.5	52535096.0	6.55	8020625.344	298593212	0.026861	3.441049e+08

Here is the interface example

[27]:
```
from ipywidgets import interact_manual, widgets, VBox
from IPython.display import display
import pandas as pd
import numpy as np
```

1.1.1 Show 10 random rows of the dataset

[28]:
```
# Loading Superhero dataset
superhero_df = pd.read_csv('superhero2.csv')

# Getting 10 random rows of dataset
superhero_df.sample(10)
```

[28]:
```
    Year                                Title   Comic  IMDB Score  \
1   1980                           Superman II      DC         6.7
24  2005                        Fantastic Four  Marvel         5.7
43  2012                The Dark Knight Rises      DC         9.1
```

```
27  2007  Fantastic Four: Rise of the Silver Surfer  Marvel         5.7
6   1989                                     Batman      DC         7.6
4   1986                           Howard the Duck  Marvel         4.3
39  2011                             Green Lantern      DC         5.9
18  2004                             Blade: Trinity  Marvel         5.8
23  2005                                    Elektra  Marvel         4.8
22  2005                              Batman Begins      DC         8.3

    RT Score  Composite Score  Opening Weekend  Box Office  Avg Ticket Price  \
1         88             77.5                  14100523.0              2.69
24        27             42.0                  56061504.0              6.41
43        86             88.5                 160887295.0              7.92
27        37             47.0                  58051684.0              6.88
6         71             73.5                  40489746.0              3.97
4         16             29.5                   5070136.0              3.71
39        27             43.0                  53174303.0              7.93
18        26             42.0                  16061271.0              6.21
23        10             29.0                  12804793.0              6.41
22        85             84.0                  48745440.0              6.41

    Opening Weekend Attendance  US Population That Year  pct_of_pop  \
1                 5.241830e+06                227224681    0.023069
24                8.745944e+06                295753151    0.029572
43                2.031405e+07                314055984    0.064683
27                8.437745e+06                301579895    0.027978
6                 1.019893e+07                246819230    0.041321
4                 1.366613e+06                240132887    0.005691
39                6.705461e+06                311591917    0.021520
18                2.586356e+06                293045739    0.008826
23                1.997628e+06                295753151    0.006754
22                7.604593e+06                295753151    0.025713

    box_off_estimate
1       3.793041e+07
24      3.593542e+08
43      1.274227e+09
27      3.993956e+08
6       1.607443e+08
4       1.881020e+07
39      4.216722e+08
18      9.974049e+07
23      8.207872e+07
22      3.124583e+08
```

1.1.2 Search for Superman movies

[57]:
```
# Function to filter movies
def search_movies(title):
    movies = superhero_df[superhero_df['Title'].str.contains(title, case=False)]
    return movies

# Searching for superman movies
display(search_movies('Superman'))
```

```
    Year                            Title Comic  IMDB Score  RT Score  \
0   1978                         Superman    DC         7.3        95
1   1980                      Superman II    DC         6.7        88
2   1983                     Superman III    DC         4.9        24
5   1987  Superman IV: The Quest for Peace    DC         3.6        10
25  2006                  Superman Returns    DC         6.3        76

    Composite Score  Opening Weekend  Box Office  Avg Ticket Price  \
0              84.0                   7465343.0              2.34
1              77.5                  14100523.0              2.69
2              36.5                  13352357.0              3.15
5              23.0                   5683122.0              3.91
25             69.5                  52535096.0              6.55

    Opening Weekend Attendance  US Population That Year  pct_of_pop  \
0                  3190317.521                222584545    0.014333
1                  5241830.112                227224681    0.023069
2                  4238843.492                233791994    0.018131
5                  1453483.887                242288918    0.005999
25                 8020625.344                298593212    0.026861

    box_off_estimate
0       1.746890e+07
1       3.793041e+07
2       4.205992e+07
5       2.222101e+07
25      3.441049e+08
```

[58]:
```
# Searching for Iron Man movies
display(search_movies('Iron Man'))
```

```
    Year       Title   Comic  IMDB Score  RT Score  Composite Score  \
32  2008    Iron Man  Marvel         7.9        94             86.5
36  2010  Iron Man 2  Marvel         7.1        74             72.5

    Opening Weekend  Box Office  Avg Ticket Price  Opening Weekend Attendance  \
32                   98618668.0              7.18                 13735190.53
36                  128122480.0              7.89                 16238590.62
```

```
    US Population That Year  pct_of_pop  box_off_estimate
32                 304374846    0.045126      7.080820e+08
36                 308745538    0.052595      1.010886e+09
```

Build the range slider of the composite score. Determine tha max and min values

Here is a start min_comp = sh['Composite Score'].min() max_comp = sh['Composite Score'].max() print(min_comp, max_comp)

[59]:
```
from ipywidgets import FloatRangeSlider

# Minimum composite score
min_comp = superhero_df['Composite Score'].min()

# Maximum composite score
max_comp = superhero_df['Composite Score'].max()
print(min_comp, max_comp)

# Creating the FloatRangeSlider for Composite Score
composite_slider = FloatRangeSlider(
    value=[min_comp, max_comp],  # Default range
    min=min_comp,                # Minimum value
    max=max_comp,                # Maximum value
    step=0.1,                    # Step size
    description='Composite Score:', # Label for the slider
    continuous_update=False      # Update only when slider is released
)

# Display the slider
display(composite_slider)
```

```
19.5 91.5
```

```
FloatRangeSlider(value=(19.5, 91.5), continuous_update=False,␣
↪description='Composite Score:', max=91.5, min=19…
```

1.2 Complete working code

[55]:
```
# Creating a text input for movie title search
title_input = widgets.Text(
    value='',
    placeholder='Type a movie title',
    description='Title:',
    disabled=False
)

# Function to filter by composite score range
```

```
def filter_by_score(title, composite_slider):
    filtered_movies = search_movies(title)
    score_filtered = filtered_movies[
        (filtered_movies['Composite Score'] >= composite_slider[0]) &
        (filtered_movies['Composite Score'] <= composite_slider[1])
    ]
    return score_filtered
```

[60]:
```
# Interactive function to combine title search and score filtering
interact_manual(
    filter_by_score,
    title=title_input,
    composite_slider=composite_slider
)
```

```
interactive(children=(Text(value='Superman', continuous_update=False,␣
 ↪description='Title:', placeholder='Type …
```

[60]:
```
<function __main__.filter_by_score(title, composite_slider)>
```