1. Loading your friend's data into a dictionary

```
# Create the years and durations lists
years = [2011,2012,2013,2014,2015,2016,2017,2018,2019,2020]
durations = [103, 101, 99, 100, 100, 95, 95, 96, 93, 90]

# Create a dictionary with the two lists
movie_dict = {'years':years,'durations':durations}

# Print the dictionary
print(movie_dict)
```

```
{'years': [2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019,
2020], 'durations': [103, 101, 99, 100, 100, 95, 95, 96, 93, 90]}
```

2. Creating a DataFrame from a dictionary

```
# Import pandas under its usual alias
import pandas as pd

# Create a DataFrame from the dictionary
durations_df = pd.DataFrame(movie_dict)

# Print the DataFrame
print(durations_df)
```

```
   years  durations
0   2011        103
1   2012        101
2   2013         99
3   2014        100
4   2015        100
5   2016         95
6   2017         95
7   2018         96
8   2019         93
9   2020         90
```

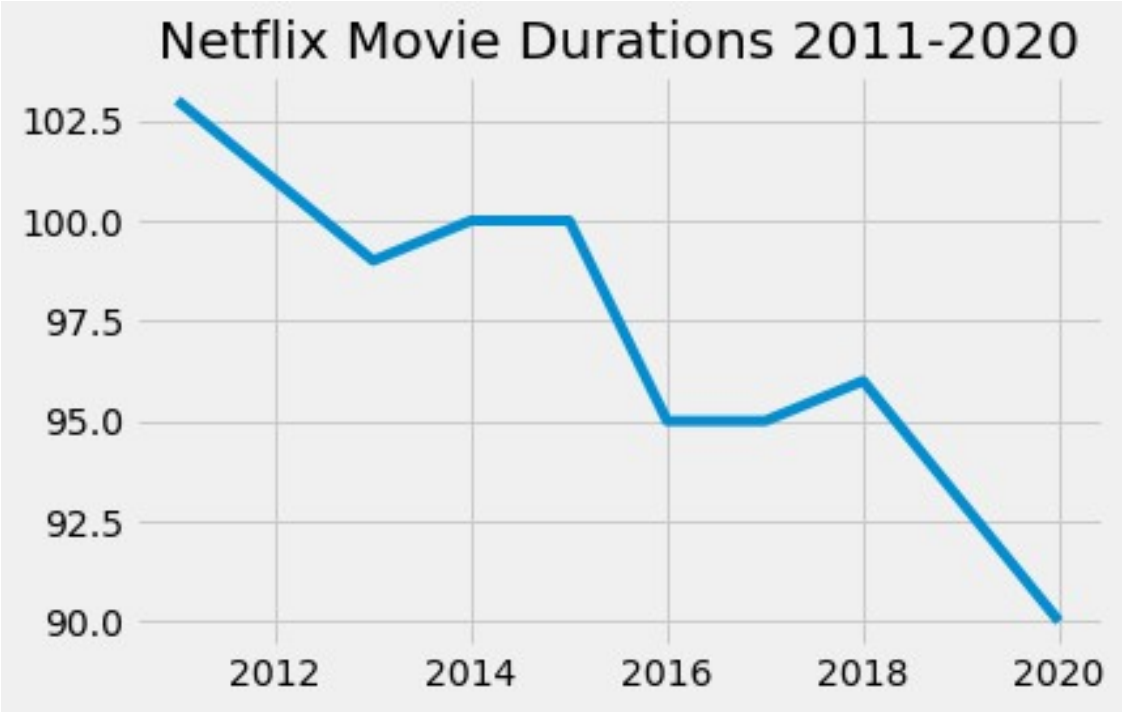
3. A visual inspection of our data

```
# Import matplotlib.pyplot under its usual alias and create a figure
import matplotlib.pyplot as plt
fig = plt.figure()

# Draw a line plot of release_years and durations
plt.plot(years, durations)


# Create a title
plt.title('Netflix Movie Durations 2011-2020')
```

```
# Show the plot
plt.show()
```



4. Loading the rest of the data from a CSV

```
# Read in the CSV as a DataFrame
netflix_df = pd.read_csv("datasets/netflix_data.csv")

# Print the first five rows of the DataFrame
print(netflix_df.head(5))
```

```
  show_id     type  title           director  \
0      s1  TV Show     3%                NaN
1      s2    Movie   7:19  Jorge Michel Grau
2      s3    Movie  23:59       Gilbert Chan
3      s4    Movie      9        Shane Acker
4      s5    Movie     21     Robert Luketic

                                                cast        country  \
0  João Miguel, Bianca Comparato, Michel Gomes, R...         Brazil
1  Demián Bichir, Héctor Bonilla, Oscar Serrano, ...         Mexico
2  Tedd Chan, Stella Chung, Henley Hii, Lawrence ...      Singapore
3  Elijah Wood, John C. Reilly, Jennifer Connelly...  United States
4  Jim Sturgess, Kevin Spacey, Kate Bosworth, Aar...  United States

          date_added  release_year  duration  \
0    August 14, 2020          2020         4
1  December 23, 2016          2016        93
2  December 20, 2018          2011        78
```

```
3  November 16, 2017          2009        80
4    January 1, 2020          2008       123

                                          description             genre

0  In a future where the elite inhabit an island ...  International TV

1  After a devastating earthquake hits Mexico Cit...            Dramas

2  When an army recruit is found dead, his fellow...     Horror Movies

3  In a postapocalyptic world, rag-doll robots hi...            Action

4  A brilliant group of students become card-coun...            Dramas
```

5. Filtering for movies!

```
# Subset the DataFrame for type "Movie"
netflix_df = pd.read_csv("datasets/netflix_data.csv")
netflix_df_movies_only = netflix_df[netflix_df['type'] == "Movie"]

# # Select only the columns of interest
netflix_movies_col_subset = netflix_df_movies_only.loc[:, ["title",
"country", "genre", "release_year","duration"]]

# # Print the first five rows of the new DataFrame
print(netflix_movies_col_subset.head(5))
```

```
   title        country          genre  release_year  duration
1   7:19         Mexico         Dramas          2016        93
2  23:59      Singapore  Horror Movies          2011        78
3      9  United States         Action          2009        80
4     21  United States         Dramas          2008       123
6    122          Egypt  Horror Movies          2019        95
```

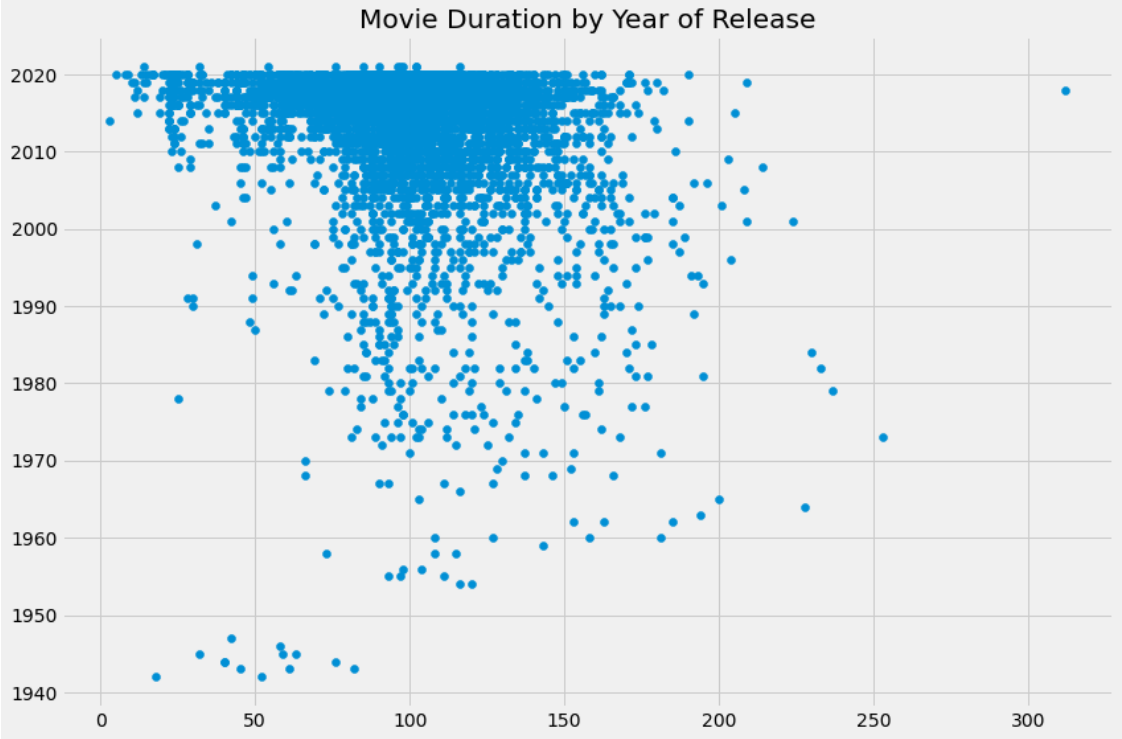
6. Creating a scatter plot

```
# Create a figure and increase the figure size
fig = plt.figure(figsize=(12,8))

# Create a scatter plot of duration versus year
plt.scatter(netflix_movies_col_subset.loc[:,'duration'],netflix_movies
_col_subset.loc[:,'release_year'])

# Create a title
plt.title("Movie Duration by Year of Release")

# Show the plot
plt.show()
```



7. Digging deeper

```
# Filter for durations shorter than 60 minutes
short_movies =
netflix_movies_col_subset[netflix_movies_col_subset['duration'] < 60]

# Print the first 20 rows of short_movies
print(short_movies.head(20))
```

```
                                                  title         country
\
35                                            #Rucker50   United States

55                  100 Things to do Before High School   United States

67   13TH: A Conversation with Oprah Winfrey & Ava ...             NaN

101                                   3 Seconds Divorce          Canada

146                                      A 3 Minute Hug          Mexico

162  A Christmas Special: Miraculous: Tales of Lady...          France

171                           A Family Reunion Christmas   United States

177                    A Go! Go! Cory Carson Christmas   United States

178                    A Go! Go! Cory Carson Halloween             NaN
```

```
179                  A Go! Go! Cory Carson Summer Camp             NaN
181             A Grand Night In: The Story of Aardman  United Kingdom
200                              A Love Song for Latasha   United States
220                            A Russell Peters Christmas          Canada
233                               A StoryBots Christmas   United States
237                              A Tale of Two Kitchens   United States
242                            A Trash Truck Christmas             NaN
247                            A Very Murray Christmas   United States
285                               Abominable Christmas   United States
295                                 Across Grace Alley   United States
305                Adam Devine: Best Time of Our Lives   United States


            genre  release_year  duration
35   Documentaries          2016        56
55   Uncategorized          2014        44
67   Uncategorized          2017        37
101  Documentaries          2018        53
146  Documentaries          2019        28
162  Uncategorized          2016        22
171  Uncategorized          2019        29
177       Children          2020        22
178       Children          2020        22
179       Children          2020        21
181  Documentaries          2015        59
200  Documentaries          2020        20
220       Stand-Up          2011        44
233       Children          2017        26
237  Documentaries          2019        30
242       Children          2020        28
247       Comedies          2015        57
285       Children          2012        44
295         Dramas          2013        24
305       Stand-Up          2019        59
```

8. Marking non-feature films

```
# Define an empty list
colors = []
```

```
# Iterate over rows of netflix_movies_col_subset
for x, y in netflix_movies_col_subset.iterrows():
    if y['genre'] == 'Children':
        colors.append("red")
    elif y['genre'] == 'Documentaries':
        colors.append('blue')
    elif y['genre'] == 'Stand-Up':
        colors.append('green')
    else:
        colors.append('black')

# Inspect the first 10 values in your list
colors[:10]
```

```
['black',
 'black',
 'black',
 'black',
 'black',
 'black',
 'black',
 'black',
 'black',
 'blue']
```

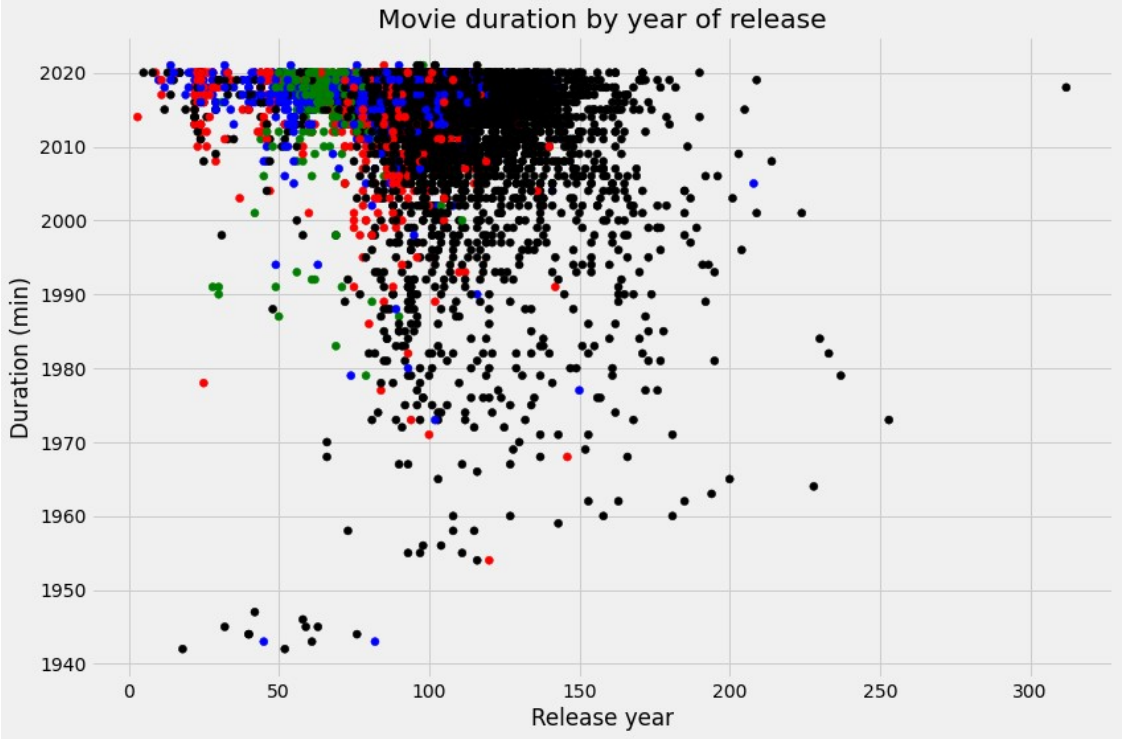
9. Plotting with color!

```
# Set the figure style and initalize a new figure
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(12,8))

# Create a scatter plot of duration versus release_year
plt.scatter(netflix_movies_col_subset.loc[:,'duration'],
netflix_movies_col_subset.loc[:,'release_year'], c = colors)

# Create a title and axis labels
plt.title("Movie duration by year of release")
plt.xlabel('Release year')
plt.ylabel('Duration (min)')

# Show the plot
plt.show()
```


Movie duration by year of release
Duration (min)
2020
2010
2000
1990
1980
1970
1960
1950
1940
0
50
100
150
200
250
300
Release year