

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

```
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

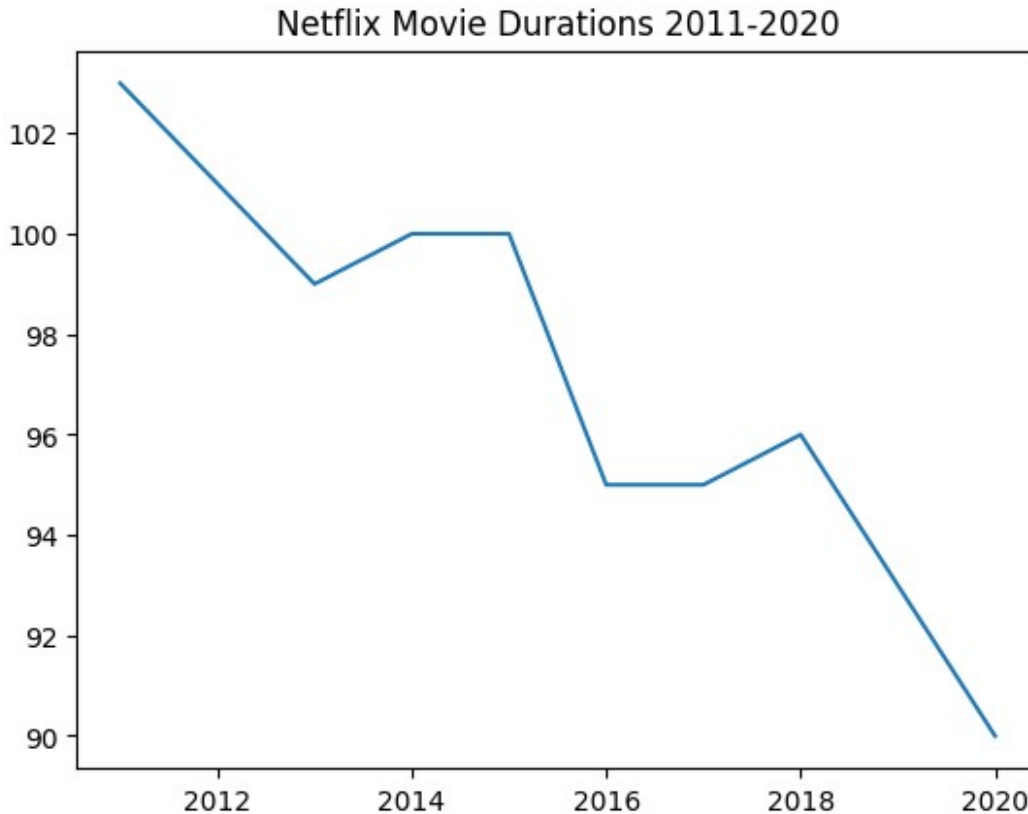
```
plt.plot(durations_df['years'], durations_df["durations"])
```

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[:5])
```

	show_id	...	genre
0	s1	...	International TV
1	s2	...	Dramas
2	s3	...	Horror Movies
3	s4	...	Action
4	s5	...	Dramas

[5 rows x 11 columns]

5. Filtering for movies!

Subset the DataFrame for type "Movie"

```
netflix_df_movies_only = netflix_df[netflix_df['type'] == 'Movie']
```

Select only the columns of interest

```
netflix_movies_col_subset = netflix_df_movies_only[['title',  
'country', 'genre', 'release_year', 'duration']]
```

```
# Print the first five rows of the new DataFrame
print(netflix_movies_col_subset[: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
```

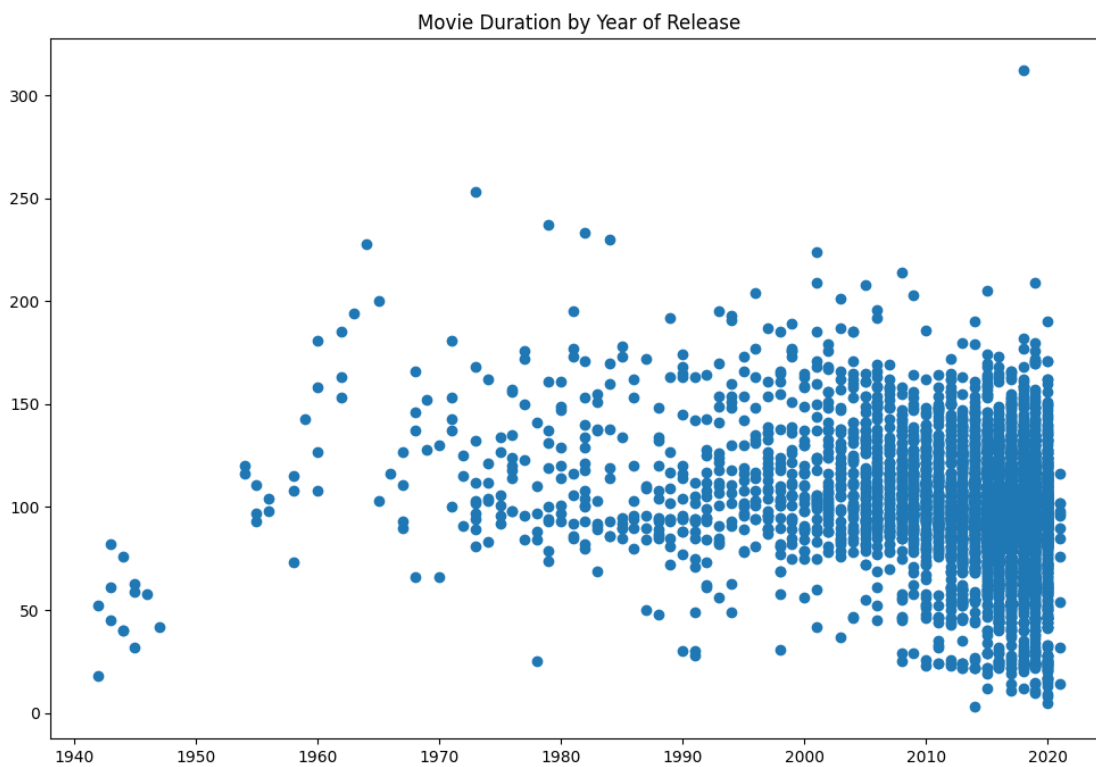
```
plt.scatter(netflix_movies_col_subset.release_year,  
netflix_movies_col_subset.duration)
```

```
# 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[:20])
```

	title	...	duration
35	#Rucker50	...	56
55	100 Things to do Before High School	...	44
67	13TH: A Conversation with Oprah Winfrey & Ava	37
101	3 Seconds Divorce	...	53
146	A 3 Minute Hug	...	28
162	A Christmas Special: Miraculous: Tales of Lady...	...	22
171	A Family Reunion Christmas	...	29
177	A Go! Go! Cory Carson Christmas	...	22
178	A Go! Go! Cory Carson Halloween	...	22
179	A Go! Go! Cory Carson Summer Camp	...	21
181	A Grand Night In: The Story of Aardman	...	59
200	A Love Song for Latasha	...	20
220	A Russell Peters Christmas	...	44
233	A StoryBots Christmas	...	26
237	A Tale of Two Kitchens	...	30
242	A Trash Truck Christmas	...	28
247	A Very Murray Christmas	...	57
285	Abominable Christmas	...	44
295	Across Grace Alley	...	24
305	Adam Devine: Best Time of Our Lives	...	59

[20 rows x 5 columns]

8. Marking non-feature films

```
colors = []

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

# Inspect the first 10 values in your list
print(colors[:11])
```

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

9. Plotting with color!

```
# Set the figure style and initialize a new figure
```

```
fig = plt.figure(figsize=(12,8))
```

```
# Create a scatter plot of duration versus release_year
```

```
plt.scatter(netflix_movies_col_subset.release_year,  
netflix_movies_col_subset.duration, c=colors)
```

```
# Create a title and axis labels
```

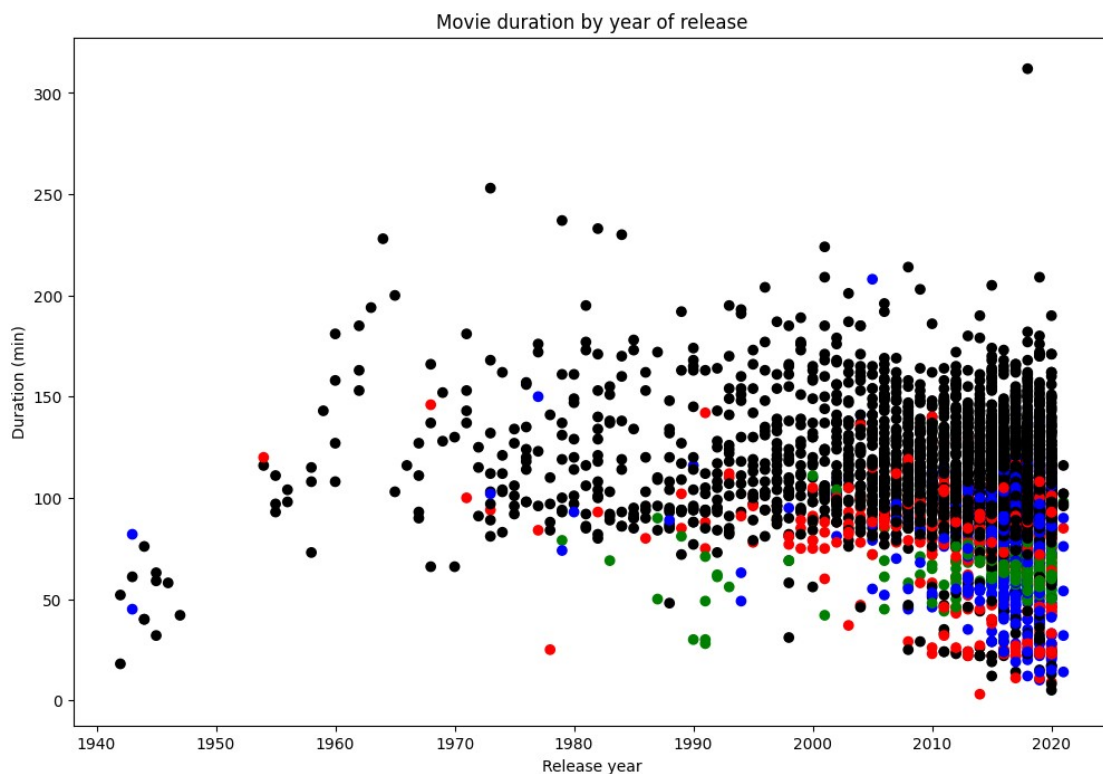
```
plt.title("Movie duration by year of release")
```

```
plt.xlabel("Release year")
```

```
plt.ylabel("Duration (min)")
```

```
# Show the plot
```

```
plt.show()
```



10. What next?

```
# Are we certain that movies are getting shorter?
```

```
are_movies_getting_shorter = "NO"
```