

## User Documentation

Thi Van Anh DUONG Student ID: 90023112

Diploma of Information Technology, Curtin College

FOP1005: Fundamentals of Programming

Coordinator: Jason Brown

10 April 2022

### Student Declaration of Originality

<input checked="" type="checkbox"/>	This assignment is my own original work, and no part has been copied from another student's work or source, except where it is clearly attributed.
<input checked="" type="checkbox"/>	All facts/ideas/claims are from academic sources are paraphrased and cited/referenced correctly.
<input checked="" type="checkbox"/>	I have not previously submitted this work in any form for THIS or for any other unit; or published it elsewhere before.
<input checked="" type="checkbox"/>	No part of this work has been written for me by another person.
<input checked="" type="checkbox"/>	I recognise that should this declaration be found to be false, disciplinary action could be taken and penalties imposed in accordance with Curtin College policy.


### Electronic Submission:

<input checked="" type="checkbox"/>	I accept that it is my responsibility to check that the submitted file is the correct file, is readable and has not been corrupted.
<input checked="" type="checkbox"/>	I acknowledge that a submitted file that is unable to be read cannot be marked and will be treated as a non-submission.
<input checked="" type="checkbox"/>	I hold a copy of this work if the original is damaged, and will retain a copy of the Turnitin receipt as evidence of submission.

# Table of contents

10 April 2022

3	Part 1: Overview of The Program's Features
3	Add Filters (Genre, Release Year, Rating)
4	Printing Movies' Names
4	Graphing The Results
5	Part 2: User Guide
	How to use the program
15	Part 3: Discussion
	Option chosen to read csv file
	Add Filter Genre Code
	Add Filter Release Year Code
	Add Filter Rating Code (Deal with Not Rated)
	Explain Graph options
	....



## Part 1: Overview of The Program's Features

This program allows Disney to search through 8790 of Netflix movies from the year of 1954 to 2021 according to genres, production years, and ratings. In addition, they can also print out the specific names and description of each movie that satisfies the above filters with graphs in order to describe the number of movies that have been already filtered and have a clear picture of how what Netflix's viewer like watching. As a result, Disney can redesign their contents appropriately.

### 1. Add Filter (Genre, Release Year, Rating)

#### a. Genre

There are a total of 33 genres, as a Data Analyst working for Disney, I can enter the genre they want to investigate, the program will automatically process and filter out the movies that match the genre.

#### b. Release Year

In the collections of movies between 1954 to 2021, as a Data Analysts working for Disney, I can enter the range of year they want to investigate, the program will automatically process and filter out the movies that match the years range.

#### c. Rating

We have 7 types of ratings, ranging from G (General), PG (Parental Guidance Suggested), PG-13 (Parental Guidance Suggested for Children Under 13), M (Mature), MA (Mature Accompanied), R (Restricted), NR (Not Rated). As a Data Analysts work for Disney, I can enter the range of rating they want to investigate, the program will automatically process and filter out the movies that match the ratings range.

## **2. Printing Movies' Name**

This feature not only allows Data Analysts print names and descriptions of all movies that match the filter above, but also provide the total number of movies that suits these features. As a result, this will greatly facilitate Disney to get accurate and clear movie statistics in the research later on when it comes to comparing data in graphs.

## **3. Graphing The Results**

This feature will turn data from numbers into visual images, through which we can easily process data, statistics and offer solutions to Disney's strategy regarding to tailor their content accordingly. In this part, I provide 3 different graph to display data and for more information I will clarify in part 3, title 5.

## Part 2: User Guide

### How to use the program

In the zip file I gave you, it already has the filter program (program.py) and movies data (netflix.csv).

In order to run this program, we have to install VMware or Visual Code studio or any programming environment can run Python. Please make sure that your software can work with Python version 1.8.31 as below:

```
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$ sudo -V
Sudo version 1.8.31
Sudoers policy plugin version 1.8.31
Sudoers file grammar version 46
Sudoers I/O plugin version 1.8.31
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$
```

Once preparing completed, open your terminal and start working with the filter program by typing the command as below:

```
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$ python3 program.py
```

After it will display the menu as below:

```
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$ python3 program.py

~~~~~WELCOME TO MY NETFILX PROGRAM  9(^_^)6 ~~~~~

Enter the option - (A)dd filters, (P)rint names, (G)raph data or e(X)it:
```

Choose your option to start filtering and analyzing the whole bundles of Netflix movies. At this stage, you just only need to type the character A or a – for Add filters, P or p– for print names, G or g– for Graph data, and X or x – for exit if you don't want to continue filtering.

**If you type A or a for adding filter**, it will display the menu as below. Next, we have 3 filters for you to choose:

```
Enter the option - (A)dd filters, (P)rint names, (G)raph data or e(X)it: a
~~~~~NOW LET'S ADD A FILTER THAT YOU WANT 🎬~~~~~
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: █
```

*(1) Type G or g – Filter Genre*, it will display all of genres we have. Just typing *exactly* the genre you want in the list above to filter all of films in this genre.

```
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: g

All genres : ['Children & Family Movies', 'Action & Adventure', 'Classic Movies', 'Classic & Cult TV', 'Anime', 'Kids' TV', 'TV Shows', 'British TV Shows', '
Reality TV', 'Dramas', 'Documentaries', 'International Movies', 'International TV Shows', 'Movies', 'Docuseries', 'Crime TV Shows', 'Comedies', 'Music & Music
als', 'Stand-Up Comedy', 'Romantic Movies', 'Cult Movies', 'TV Dramas', 'TV Comedies', 'TV Action & Adventure', 'Romantic TV Shows', 'Horror Movies', 'Indepen
dent Movies', 'Thrillers', 'Spanish-Language TV Shows', 'TV Sci-Fi & Fantasy', 'Sci-Fi & Fantasy', 'TV Horror', 'Sports Movies']

Choose genre you want: █
```

Otherwise, if you mistype, this message will appear and you have to type again until it is true:

```

Choose genre you want: Anim
Please enter a valid genre !
  _
Choose genre you want: Anime
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: █

```

(2) *Type Y or y – Filter Year.* Choose your minimum and maximum year in the range of 2019 and 2020 as below:

```

Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: y
*~~~~~*~~~~~*~~~~~*~~~~~*~~~~~*~~~~~*
Choose a year between 1954 and 2021:

Please choose a minimun year: 2019
Please choose a maximun year: 2020

```

Make sure that you don't choose the wrong years or give the program the year that don't have in netflix.csv file. If it happened, the error message will appear and you have to give the value until it valid.

```

*~~~~~*~~~~~*~~~~~*~~~~~*~~~~~*~~~~~*
Choose a year between 1954 and 2021:

Please choose a minimun year: 1921
Error! This year is not available.

Please choose a minimum year: 2019
Please choose a maximun year: 2034
Error! This year is not available.

Please choose a maximum year: 2018
Error! This year is not available.

Please choose a maximum year: 2020

Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: █

```

(3) *Type R or r – Filter Rating.* The program will provide you a list of rating as I mentioned in the Overview part. Choose your low range and high range according to the sequence of the list provided.

```
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a
~~~~~NOW LET'S ADD A FILTER THAT YOU WANT 🍷~~~~~
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating:  r
Rating list:  ['G', 'PG', 'PG-13', 'M', 'MA', 'R', 'NR']
Choose a low rating: pg
Choose a high rating: m
```

Make sure that you don't choose the low range > high range or give the program the rating that don't have in netflix.csv file, just type exactly what is in the rating list, both uppercase and lowercase are accepted. If it happened, the error message will appear and you have to give the value until it valid.

```
~~~~~NOW LET'S ADD A FILTER THAT YOU WANT 🍷~~~~~
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating:  r
Rating list:  ['G', 'PG', 'PG-13', 'M', 'MA', 'R', 'NR']
Choose a low rating: M
Choose a high rating: G
Error ! Please enter again !
Choose a high rating: pg
Error ! Please enter again !
Choose a high rating: ma
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: █
```

**If you type P or p for printing movies name,** you can choose your filter first and then print or you can print initially. This menu will provide you the total of movies, just make it easier to recognize whether or not your movies is filtered.



*Printing the movies before adding filters, total movies is **8790***

```

Terminal - ccadmin@CCUbu... Terminal - ccadmin@CCUbu...
Terminal - ccadmin@CCUbu54bit: ~/FOP/Assignment1
File Edit View Terminal Tabs Help
Kate : Slipped a fatal poison on her final job a ruthless assassin working in Tokyo has less than 24 hours to find out who ordered the hit and exact revenge
.
Movies:
Malcolm & Marie : As a filmmaker and his girlfriend return home from his movie premiere smoldering tensions and painful revelations push them toward a roman
tic reckoning.
Movies:
Operation Varsity Blues: The College Admissions Scandal : Reenactments drive this documentary investigating the mastermind behind a scam to get the kids of
rich and famous families into top US universities.
Movies:
Outside the Wire : In the near future a drone pilot sent into a war zone finds himself paired with a top-secret android officer on a mission to stop a nucle
ar attack.
Movies:
SAS: Rise of the Black Swan : A special forces operative traveling from London to Paris with his girlfriend takes action when armed ruthless mercenaries sei
ze control of their train.
Movies:
Strip Down Rise Up : In an effort to reclaim their bodies and lives a group of women explore the intersections of movement and meaning in a powerful pole da
ncing program.
Movies:
Sweet Girl : He lost the love of his life to a pharmaceutical company's greed. Now his daughter is without a mother and he's without justice. For now.
Movies:
The Seventh Day : An inexperienced priest teams up with a hardened exorcist to stop the demonic possession of a young boy. But darkness lies where they leas
t expect it.
Movies:
The Vault : A genius engineer and his crew of amateur thieves plot a scheme to seize a legendary lost treasure hidden in a fortress beneath the Bank of Spai
n.
Movies:
The White Tiger : The ambitious driver for a rich Indian family uses his wit and cunning to escape from poverty and become an entrepreneur. Based on the bes
tselling novel.
Movies:
The Woman in the Window : Confined to her home by agoraphobia a psychologist becomes obsessed with her new neighbors – and solving a brutal crime she witnes
ses from her window.
Total movies: 8790
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it:

```

*Adding multiple filter and the print, total movies is **134***

```

Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a
-----NOW LET'S ADD A FILTER THAT YOU WANT 0_0-----
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: g
All genres : ['Children & Family Movies', 'Action & Adventure', 'Classic Movies', 'Classic & Cult TV', 'Anime', 'Kids' TV', 'TV Shows', 'British TV Shows', '
Reality TV', 'Dramas', 'Documentaries', 'International Movies', 'International TV Shows', 'Movies', 'Docuseries', 'Crime TV Shows', 'Comedies', 'Music & Music
als', 'Stand-Up Comedy', 'Romantic Movies', 'Cult Movies', 'TV Dramas', 'TV Comedies', 'TV Action & Adventure', 'Romantic TV Shows', 'Horror Movies', 'Indepen
dent Movies', 'Thrillers', 'Spanish-Language TV Shows', 'TV Sci-Fi & Fantasy', 'Sci-Fi & Fantasy', 'TV Horror', 'Sports Movies']
Choose genre you want: Anime
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a
-----NOW LET'S ADD A FILTER THAT YOU WANT 0_0-----
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: 2019
We don't have this filter ! (T__T)
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating or e(X)it: 2020
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a
-----NOW LET'S ADD A FILTER THAT YOU WANT 0_0-----
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: r
Rating list: ['G', 'PG', 'PG-13', 'M', 'MA', 'R', 'NR']
Choose a low rating: m
Choose a high rating: r

```

```

Movies:
RESIDENT EVIL: Infinite Darkness : Years after the horrors of Raccoon City Leon and Claire find themselves consumed by a dark conspiracy when a viral attack ravages the White House.

Movies:
The Way of the Househusband : "After disappearing from the underworld the legendary yakuza Tatsuo ""the Immortal Dragon "" resurfaces – as a fiercely devoted stay-at-home husband."

Movies:
Thus Spoke Kishibe Rohan : A popular manga creator becomes enmeshed in paranormal events while conducting research: Stand User Kishibe Rohan visits Italy goes bankrupt and more.

Movies:
Trese : In Manila where dark supernatural forces pervade the criminal underworld it's up to Alexandra Trese to keep the peace – but there's a storm brewing.

Movies:
Yasuke : A peaceful boatman once known as the Black Samurai is pulled back into conflict when he takes a little girl with mysterious powers under his wing.

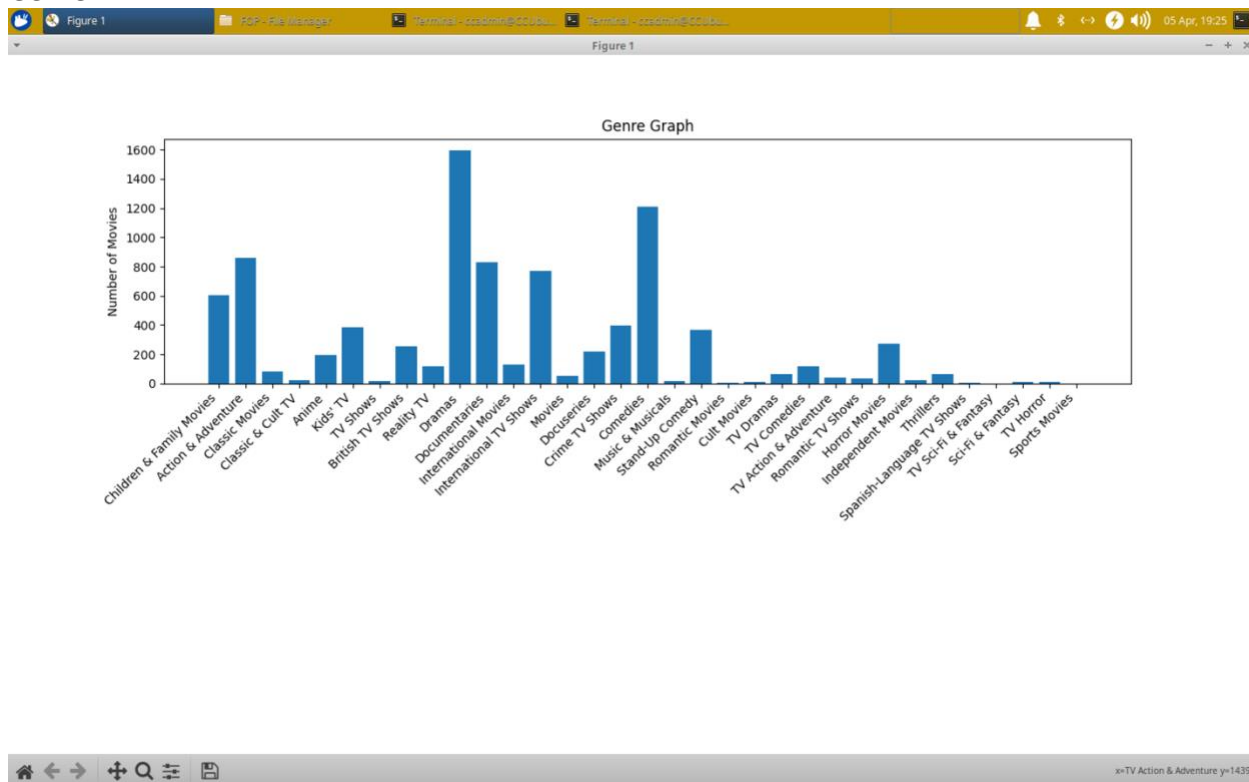
Total movies: 134

```

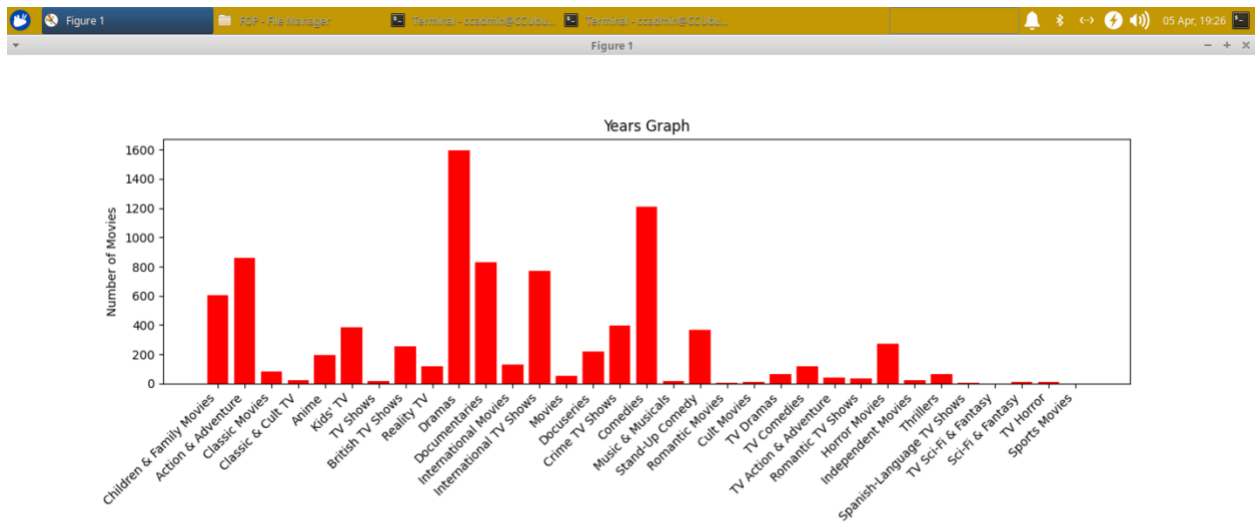
**If you type G or g for graphing data.**

Firstly, I will show you how raw data before filtering displayed in the graph.

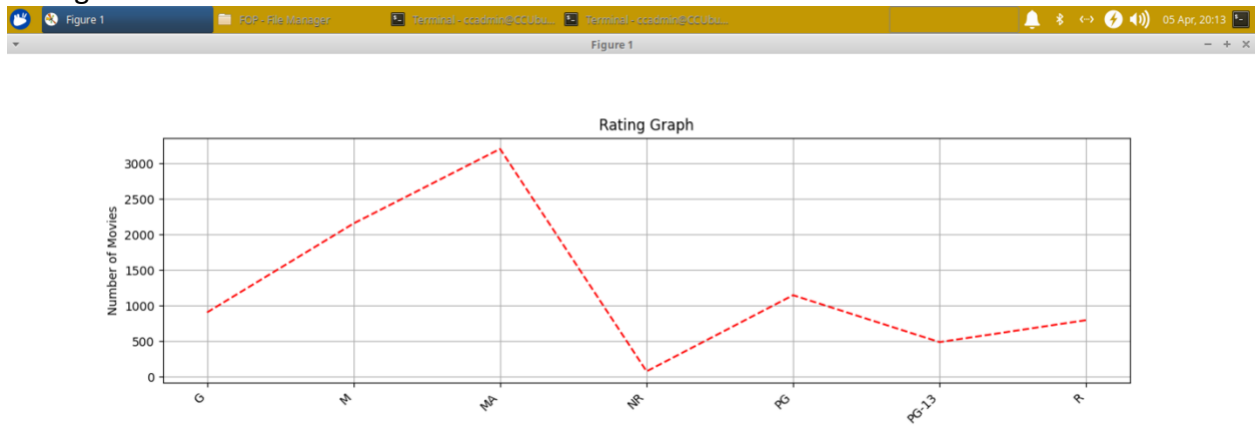
Genre:



Release Year:



Rating:



Now let's add some filters and graph it!

```
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$ python3 program.py
-----WELCOME TO MY NETFILX PROGRAM  9(^_^)f -----

Enter the option - (A)dd filters, (P)rint names, (G)raph data or e(X)it: a

-----NOW LET'S ADD A FILTER THAT YOU WANT  0_0 -----

Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: g

All genres : ['Children & Family Movies', 'Action & Adventure', 'Classic Movies', 'Classic & Cult TV', 'Anime', 'Kids' TV', 'TV Shows', 'British TV Shows', 'Reality TV', 'Dramas', 'Documentaries', 'International Movies', 'International TV Shows', 'Movies', 'Docuseries', 'Crime TV Shows', 'Comedies', 'Music & Music als', 'Stand-Up Comedy', 'Romantic Movies', 'Cult Movies', 'TV Dramas', 'TV Comedies', 'TV Action & Adventure', 'Romantic TV Shows', 'Horror Movies', 'Independent Movies', 'Thrillers', 'Spanish-Language TV Shows', 'TV Sci-Fi & Fantasy', 'Sci-Fi & Fantasy', 'TV Horror', 'Sports Movies']

Choose genre you want: Dramas

Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a

-----NOW LET'S ADD A FILTER THAT YOU WANT  0_0 -----

Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: y

*-----*-----*-----*-----*-----*
Choose a year between 1954 and 2021:

Please choose a minimum year: 1985
Please choose a maximum year: 2001

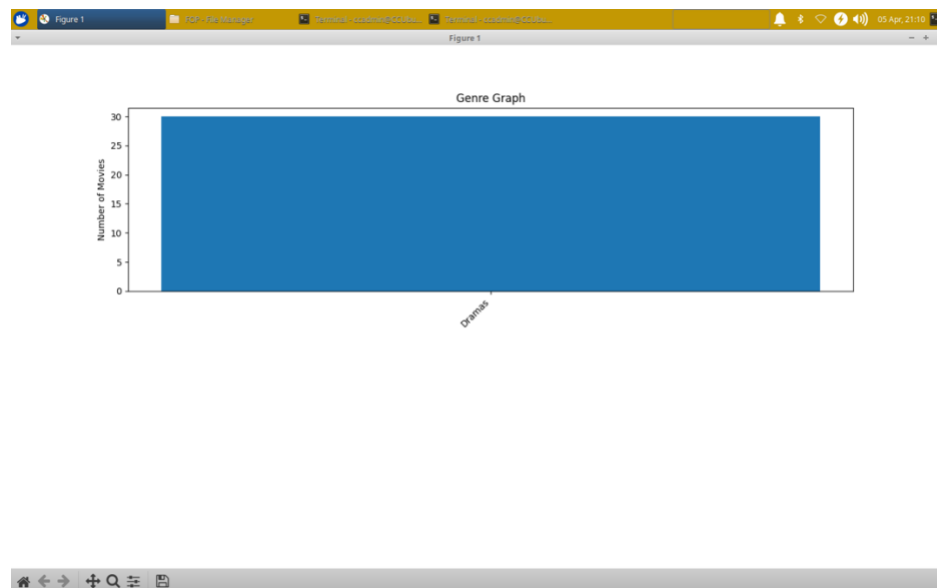
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a

-----NOW LET'S ADD A FILTER THAT YOU WANT  0_0 -----

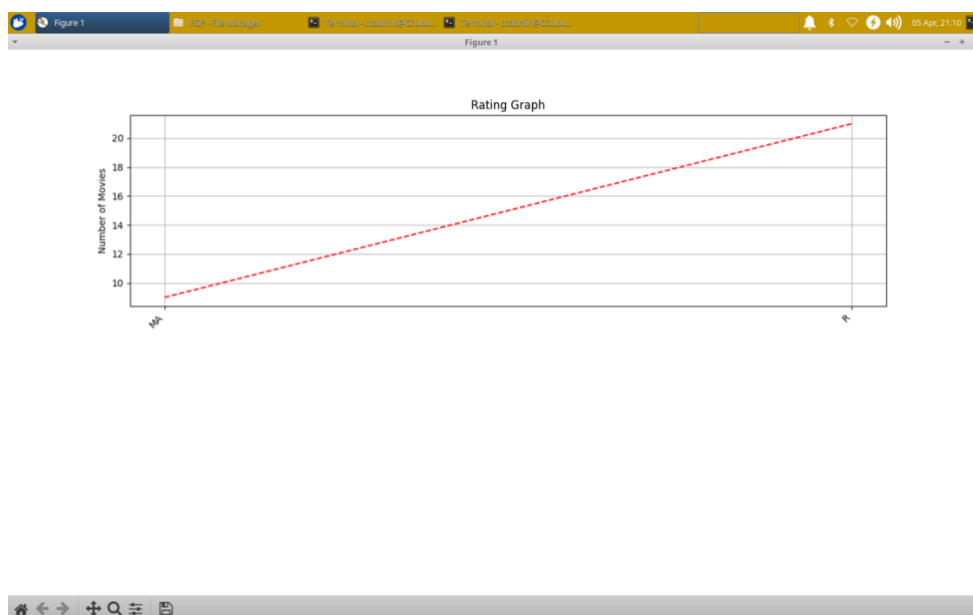
Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: r

Rating list: ['G', 'PG', 'PG-13', 'M', 'MA', 'R', 'NR']
Choose a low rating: ma
Choose a high rating: r
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: g
Choose a graph type: (G)enre, Rating (Y)ear, (R)ating: █
```

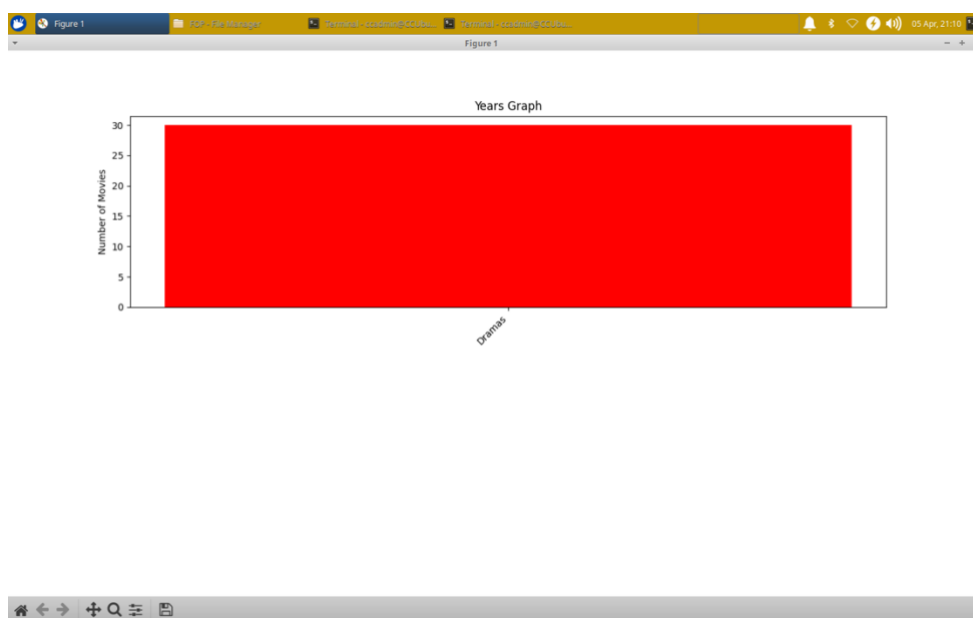
Graph with Genre:



Graph with rating:



Graph with year:



If you type X or x, they program will automatically exit:

```
Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: g
Choose a graph type: (G)enre, Rating (Y)ear, (R)ating: y

Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: g
Choose a graph type: (G)enre, Rating (Y)ear, (R)ating: r

Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: x
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$
```

If you type something that does not include in our menu, the error message will appear.

And then, you have to re-enter again until the input is valid.

```
ccadmin@CCUbuntu64bit:~/FOP/Assignment1$ python3 program.py

~~~~~WELCOME TO MY NETFILX PROGRAM  9(^_^)8 ~~~~~

Enter the option - (A)dd filters, (P)rint names, (G)raph data or e(X)it: kjfgksjd
Please choose a valid option ! (╯_╯)

Enter the option: (A)dd filters, (P)rint names, (G)raph data or e(X)it: a

~~~~~NOW LET'S ADD A FILTER THAT YOU WANT 0_0) ~~~~~

Please choose a filter type - (G)enre, Release (Y)ear, (R)ating: 
```

## Part 3: Discussion

### 1. Option chosen to read csv file

```
# Reading The File

netfile = open('netflix.csv','r')
data = netfile.readline() # This line is for read the first line and ignore it
data = netfile.readline() # Start working at this line

rawData = [] # Variable stores all movies information

while data:
    splitline = data.strip().split(',')
    rawData.append(splitline)
    data = netfile.readline()

netfile.close()
```

- Read the csv file: I used `readline()` 2 times simply because I want to skip the first line of the file including title, type, country, release year, rating, genre and description which contains no necessary data. I just only work with data the second `readline()` code.

	A	B	C	D	E	F	G	H
1	title	type	country	release year	rating	duration	genre	description
2	White Christmas	Movie	United States	1954	G	120 min	Children & Family Movies	Two war buddies fall for two sisters and follow the girls to a resort owned by their former commanding officer who is in danger of losing the place.

- rawData*: is the list that contains all of movies information from `netflix.csv` file. I don't choose an array for my *rawData* because array must have unchangeable elements, while my *rawData* need to update later.
- The while loop is to check whether or not the file I work with is empty or not, if it contains data I will be happy to work with, otherwise my program will stop there.

## 2. Add Genre Code

\*\*\*Finding all types of genres: use **findList(a)** – a (position of genre in the file: 6) to in the 8790 rows of data.

- *Result:* variable is an empty list, containing these types of genre.

I scan all through the length of my *rawData* and add each genre that I find in netflix.csv to my *result = []*, if it already had, I will ignore and don't add to *result*, if not I will continue my loop until reaching all genres.

```
def findList(a):
    result = []
    for i in range(len(rawData)):
        if (rawData[i][a] not in result):
            result.append(rawData[i][a])
    return(result)
```

Funtion: *addFilterGenre(ogenre)* – *ogenre* (input genre from user)

*Result:* variable here is an empty list, storing all information of movies after adding filter genre.

I run through length of my *rawData* and check the condition if my input genre is the genre in my *rawData*, I will add the film having this genre to result variable and return it.

```
def addFilterGenre(ogenre):
    result = []
    for i in range(len(rawData)):
        if rawData[i][6] == ogenre:
            result.append(rawData[i])
    return(result)
```



For adding genre filter, I just simply call my function *findList(a)* to create a list of *genre*.

**\*\*Typing my genre option (ogenre) & check condition for my ogenre**

In order to check whether or not my *ogenre* is a valid value in the list of *genre*: create new variable called *checkgenre* and set it is False (data type: Boolean) by default. Next, I scan through length of genre list to check my *ogenre* input. If *ogenre* value is in the *genre* list, *checkgenre* will return True. By contrast, if *checkgenre* is still False, I will let the user type genre until I get the valid value.

Then I just simply call the *addFilterGenre* function and update my *rawData*.

```
# Add Filter Genre

if filt == "G":
    genre = findList(6)
    print('All genres : ', genre)
    print()
    ogenre = input("Choose genre you want: ")
    print()
    checkgenre = False
    for g in range(len(genre)):
        if ogenre == genre[g]:
            checkgenre = True
    if checkgenre == False:
        print("Please enter a valid genre !")
        print(" ❌ ")
        print()
        ogenre = input("Choose genre you want: ")
        print()
    else:
        rawData = addFilterGenre(ogenre)
```

### 3. Add Release Year Code

\*\*Checking minimum and maximum value from user: the minimum and maximum input must be in the range of 1954 and 2021 and the minimum input cannot larger than maximum input.

```
def minvalid(minYear):
    while minYear < 1954 or minYear > 2021:
        print("Error! This year is not available.")
        print()
        minYear = int(input("Please choose a minimum year: "))
    return(minYear)

def maxvalid(minYear,maxYear):
    while maxYear > 2021 or maxYear < 1954 or maxYear < minYear:
        print("Error! This year is not available.")
        print()
        maxYear = int(input("Please choose a maximum year: "))
    return(maxYear)
```

For adding filter Release Year, similarly, I create *result* variable - is an empty list, storing all information of movies after adding filter release year.

Scanning through my *rawData*, I will compare the year in my *rawData* whether or not it satisfies the condition as the code below. If it does, the movie will be added in my empty list.

```
def addFilterReleaseYear(minYear,maxYear):
    result = []
    for i in range(len(rawData)):
        if int(rawData[i][3]) >= minYear and int(rawData[i][3]) <= maxYear:
            result.append(rawData[i])
    return(result)
```

At this stage, I just need to call the aforementioned function and update my rawData.

```
# Add Filter Release Year

elif filt == "Y":
    print("*****")
    print('Choose a year between 1954 and 2021: ')
    print()
    minYear = int(input("Please choose a minimun year: "))
    minYear = minvalid(minYear)

    maxYear = int(input("Please choose a maximun year: "))
    maxYear = maxvalid(minYear,maxYear)
    print()

    rawData = addFilterReleaseYear(minYear,maxYear)
```

#### 4. Add Rating Code

Do the same things regarding to empty list *result* and add the data to *result*.

**\*\*Note:** the difference is checking the condition of low range and high range. Because the list of rating is a string, so I cannot set the condition like I did in the release year filter. My Idea is that I use a **intermediate variable** (called: position). This variable represent to the value lies between low range and high range. Then I just need to compare the position of my low range, high range and variable position as below to add the appropriate film.

```
def addFilterRating(low_position, high_position):
    result = []
    for i in range(len(rawData)):
        position = -1
        for j in range(len(rating)):
            if rawData[i][4] == rating[j]:
                position = j
        if position != -1 and position <= high_position and position >= low_position:
            result.append(rawData[i])
    return(result)
```

\*\*\*Looking at the code below you will wonder, why I need to set position = -1? and why is it equal to -1 but not another number?

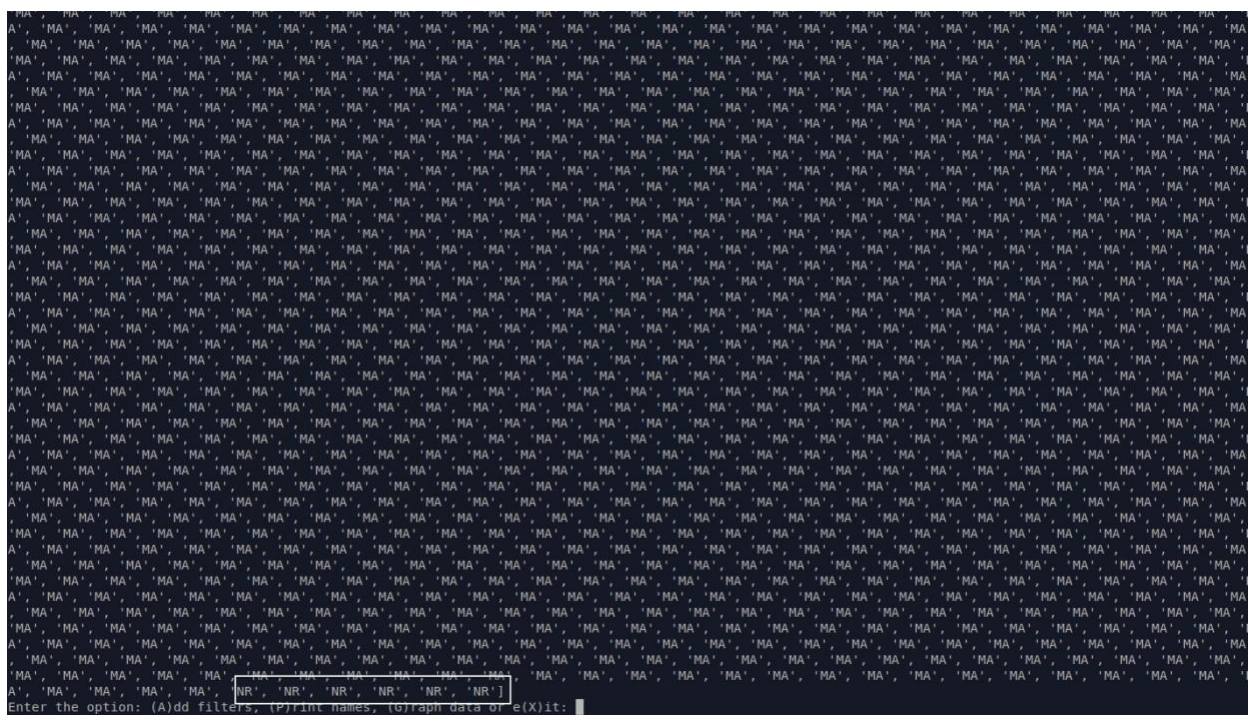
Here is my rating list:

```
[ 'G', 'PG', 'PG-13', 'M', 'MA', 'R' ]
```

Index =            0        1        2            3        4        5







This is simply because position value is in for loop, every single time we run the code, the position value keeps changing until  $j = 6$ , so in this case 'NR' is still valid regardless of our condition.

My solution there here is setting an initial value for position and this value has to be out of range  $[0;6]$  which is the index of rating list. Therefore, when the program run for loop again, there will be no probability for other invalid indexes are true.

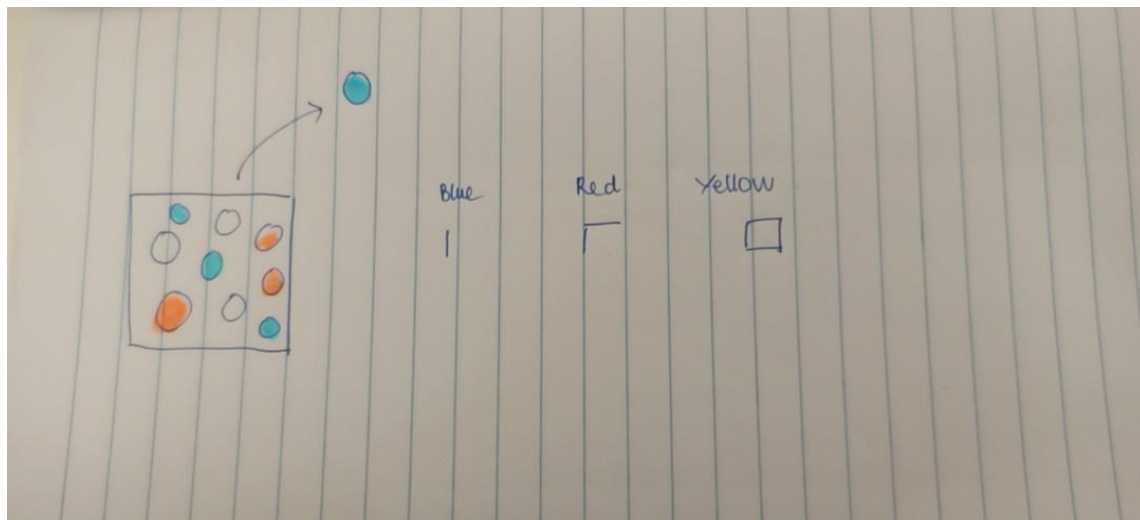
\*\*\* Coping with NR (Not Rated): I decided to ignore all movies that have this rating, Because we don't know whether or not it is suitable for viewers, especially children under 13 or not.

## 5. Dealing with number of movies in graphing:

\*\* In plot part, I choose the following method to count how many movies of a genre and rating.

To sort it out, I try to visualize genre as balls (for different genres I assume as red ball, blue ball, yellow ball, etc ...). My math problem is: counting how many red ball, blue ball, yellow ball have in the box.

⇒ I will write red ball, blue ball, yellow ball category in the blackboard and start picking ball from the box, if it is blue ball, I will add 1 to it, and continue till the end. Like this:



Similarly, I create a list of zero elements (*count*) having the length equal to length of genre & rating list – *like writing categories in the blackboard*

Then, I will scan through these two lists (*like my box containing of balls*) , and count as the code below:

```
def countFilms(listFilter,a):
    count = np.zeros(len(listFilter))
    for i in range(len(rowData)):
        for j in range(len(listFilter)):
            if listFilter[j] == rowData[i][a]:
                count[j] += 1
    return(count)
```

## 6. Explain Graph Option

*Genre Filter*: Bar chart (x label: genre, y label: number of movies)

*Release Year filter*: Bar chart (x label: genre, y label: number of movies)

⇒ In my perspective, bar chart is perfectly suitable for these two categories, simply because whether we have only 1 genre or multiple genres, the bar graph will show

specifically the quantity each category has, how much or how little. In addition, looking at the high and low shape of the bar graph, we can also compare the two genres from that to have a more general view.

*Rating Filter:* Line Graph (x label: rating, y label: number of movies)

⇒ Actually with Rating data, bar graph is also suitable, but I go for line graph because the rating only has 6 types, looking at the line graph will be less confusing, in addition, the straight lines represent high and low data, up and down, folded. This will make it easier for us to compare data when looking at the bar chart.