Calculate the average rating for free apps

Calculate the average rating for non-free apps

Calculate the average rating of Gaming and Non-Gaming apps

Calculate the average rating of free Gaming apps

Compute the average rating of the apps whose genre is either "Social Networking" or "Games."

Compute the average rating of the non-free apps whose genre is either "Social Networking" or "Games."

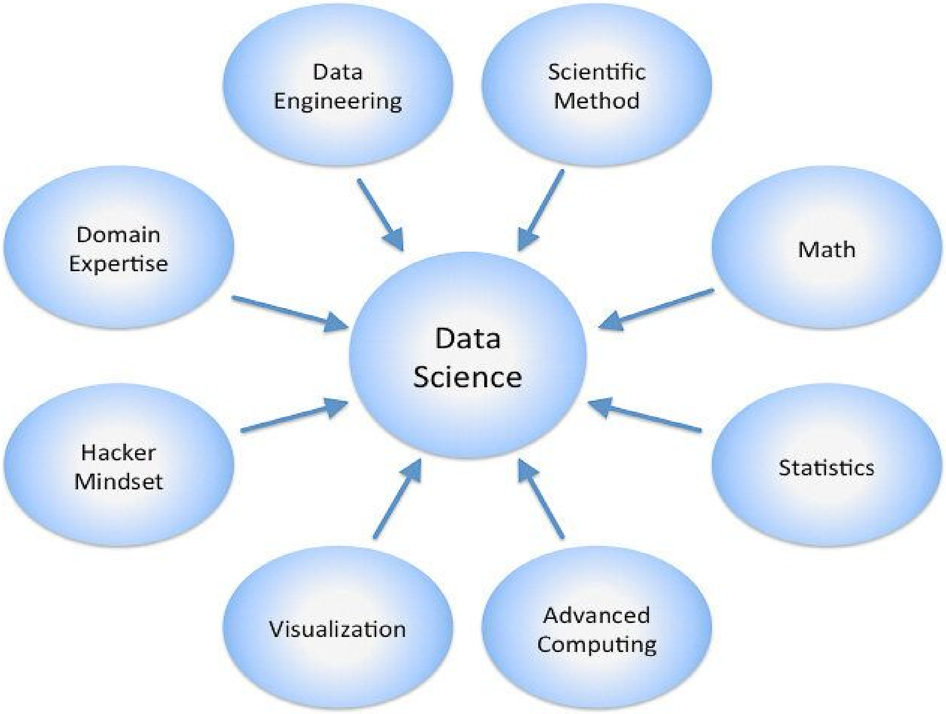
Compute the average rating of the apps that have a price greater than $9.

Categorise all apps by labelling each app as "free"(=0), "affordable" (<20), "expensive" (<50) or "very expensive" (>50). Add a label column to the data.

Data Science: Data Science is a multi- disciplinary field that uses scientific methods, processes, algorithms to produce knowledge & insights from structured & unstructured data

OR

Data science is the analytics that comprises advanced understanding of statistics, programming language along with all the required skills for data analytics and machine learning. This helps in performing predictive analysis, sentiment analysis, text mining to extract complex patterns from the structured/unstructured dataset. Unstructured dataset like from audio, video or image processing. A data scientist perform all the basis analysis along with applying advance algorithms to get the outcome which is much more reliable and satisfactory



What is Python?

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, that can be used for a wide variety of applications. It includes data structures, dynamic typing, dynamic binding, and many

more features that make it as useful for complex application development

Why Python Became Popular?

Python is used in a variety of purposes, ranging from web development to data science to DevOps. Data science and machine learning becoming more common in many types of companies, and Python becoming a common choice for that purpose. This is particularly visible in the growth of the pandas package, which is the fastest-growing Python-related tag on the site. As for which industries are using Python, we found that it is more visited in a few industries, such as electronics, manufacturing, software, government, and especially universities.

INSTALLATION

Platforms to Run Python

IDLE, PyCharm, PyScripter, PyDev, Anaconda, Jupyter etc…
Anaconda Anaconda is a platform/navigator to run python.

Why should we use Anaconda for Python?

Anaconda is popular because it brings many of the tools used in data science and machine learning. Anaconda contains,popular python libraries that can be used in data science . It also comes with the jupyter notebook and Ipython distribution. So, it saves you from importing numerous libraries separately

Link for installation of Anaconda Software: https://www.anaconda.com/distribution/

Advantages

- Object oriented programming language.
- Simple to use.
- Best for Machine learning and Data Sciences.
- Huge collection of packages and libraries

Jupyter Notebook Environment

- Jupyter is a web - application
- Jupyter's name is a reference to the three core programming languages supported by Jupyter, which are Julia, Python and R.

Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses Include:data cleaning and transformation,numericalsimulation,statistical modeling, data visualization, machine learning, and much more

Advantages

- Best for data exploration, data preparation, data validation, Productionalization
- It is a client-server application used for running notebook documents in the browser. Notebook documents are documents able to contain bothcodeand text such as paragraphs, equations and so on

Working with Jupyter Notebook

- Open Anaconda Navigator
- Open Jupyter Notebook by clicking on “Launch” option below the Jupyter Notebook application

Opening of a new Python Notebook

After launching the Jupyter Notebook

- Create a folder on the Desktop by entering Desktop folder
- Click on New (at the right side) >> Folder
- Give a name(Data Analysis for Mobile App Uses) to that folder
- Enter that folder New >> Python 3

Introduction to Numpy Python

NumPy, which stands for Numerical **Python**, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using **NumPy**, mathematical and logical operations on arrays can be performed. This **tutorial** explains the basics of **NumPy** such as its architecture and environment.

What are NumPy Arrays?

NumPy is a Python package that stands for 'Numerical Python'. It is the core library for scientific computing, which contains a powerful n-dimensional array object.

Where is NumPy used?

Python NumPy arrays provide tools for integrating C, C++, etc. It is also useful in linear algebra, random number capability etc. NumPy array can also be used as an efficient multi-dimensional container for generic data. Now, let me tell you what exactly is a Python NumPy array.

Python NumPy Array: Numpy array is a powerful N-dimensional array object which is in the form of rows and columns. We can initialize NumPy arrays from nested Python lists and access it elements. In order to perform these NumPy operations, the next question which will come inyour mind is:

How do I install NumPy?

To install Python NumPy, go to your command prompt and type “pip install numpy”. Once the installation is completed, go to your IDE (For example: PyCharm) and simply **import it by typing: “import numpy as np”**

How do I start NumPy?

Let us see how it is implemented in PyCharm:

Single-dimensional Numpy Array:

```
import numpy as np
a=np.array([1,2,3])
print(a)
```

Output – [1 2 3]

Multi-dimensional Array:

```
a=np.array([(1,2,3),(4,5,6)])
print(a)
```

O/P – [[1 2 3]

[4 5 6]]

Introduction to Pandas Python

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. The name Pandas is derived from the word Panel Data – an Econometrics from Multidimensional data.

Key Features of Pandas

Fast and efficient DataFrame object with default and customized indexing.
Tools for loading data into in-memory data objects from different file formats.
Data alignment and integrated handling of missing data.
Reshaping and pivoting of date sets.
Label-based slicing, indexing and subsetting of large data sets.
Columns from a data structure can be deleted or inserted.
Group by data for aggregation and transformations.
High performance merging and joining of data.
Time Series functionality.

Importing Pandas
import `pandas as pd`

Pandas Series

The Series is a one-dimensional array that can store various data types, including mix data types. The row labels in a Series are called the index. Any list, tuple and dictionary can be converted in to Series using 'series' method as shown below,

```
 >>> import pandas as pd
>>> # converting tuple to Series
>>> h = ('AA', '2012-02-01', 100, 10.2)
>>> s = pd.Series(h)
>>> type(s)
>>> print(s)
0 AA 1 2012-02-01 2 100 3 10.2 dtype: object
>>> # converting dict to Series
>>> d = {'name' : 'IBM', 'date' : '2010-09-08', 'shares' : 100,
'price' : 10.2}
>>> ds = pd.Series(d)
>>> type(ds)
>>> print(ds) date 2010-09-08 name IBM price 10.2 shares 100 dtype:
object
```

DataFrame

DataFrame is the widely used data structure of pandas. Note that, Series are used to work with one dimensional array, whereas DataFrame can be used with two dimensional arrays. DataFrame has two different index i.e. column-index and row-index. The most common way to create a DataFrame is by using the dictionary of equal-length list as shown below. Further, all the spreadsheets and text files are read as DataFrame, therefore it is a very important data structure of pandas.

```
>>> data = { 'name' : ['AA', 'IBM', 'GOOG'], ... 'date' :
['2001-12-01', '2012-02-10', '2010-04-09'], ... 'shares' : [100, 30,
90], ... 'price' : [12.3, 10.3, 32.2] ... }
>>> df = pd.DataFrame(data)
>>> type(df)
>>> df date name price shares 0 2001-12-01 AA 12.3 100 1 2012-02-10
IBM 10.3 30 2 2010-04-09 GOOG 32.2 90
```

Introduction to Data Visualization in Python

Data visualization is the technique to present the data in a pictorial or graphical format. It enables stakeholders and decision makers to analyze data visually. The data in a graphical format allows them to identify new trends and patterns easily.

.1.Scatter plot

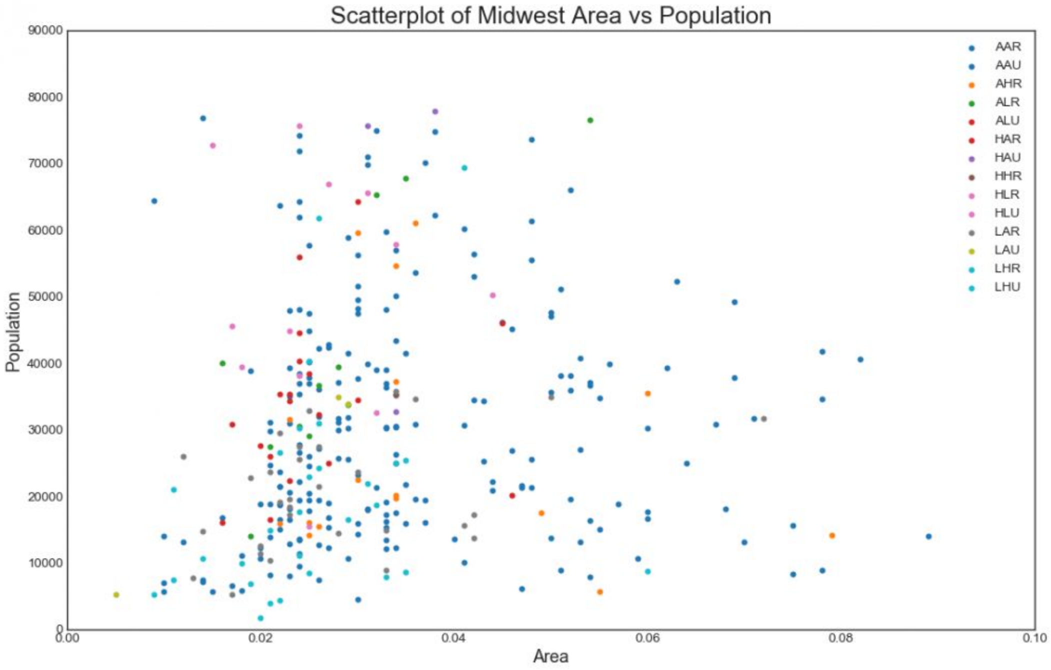
2.Bar plot

3.Histogram

4.Pie Chart

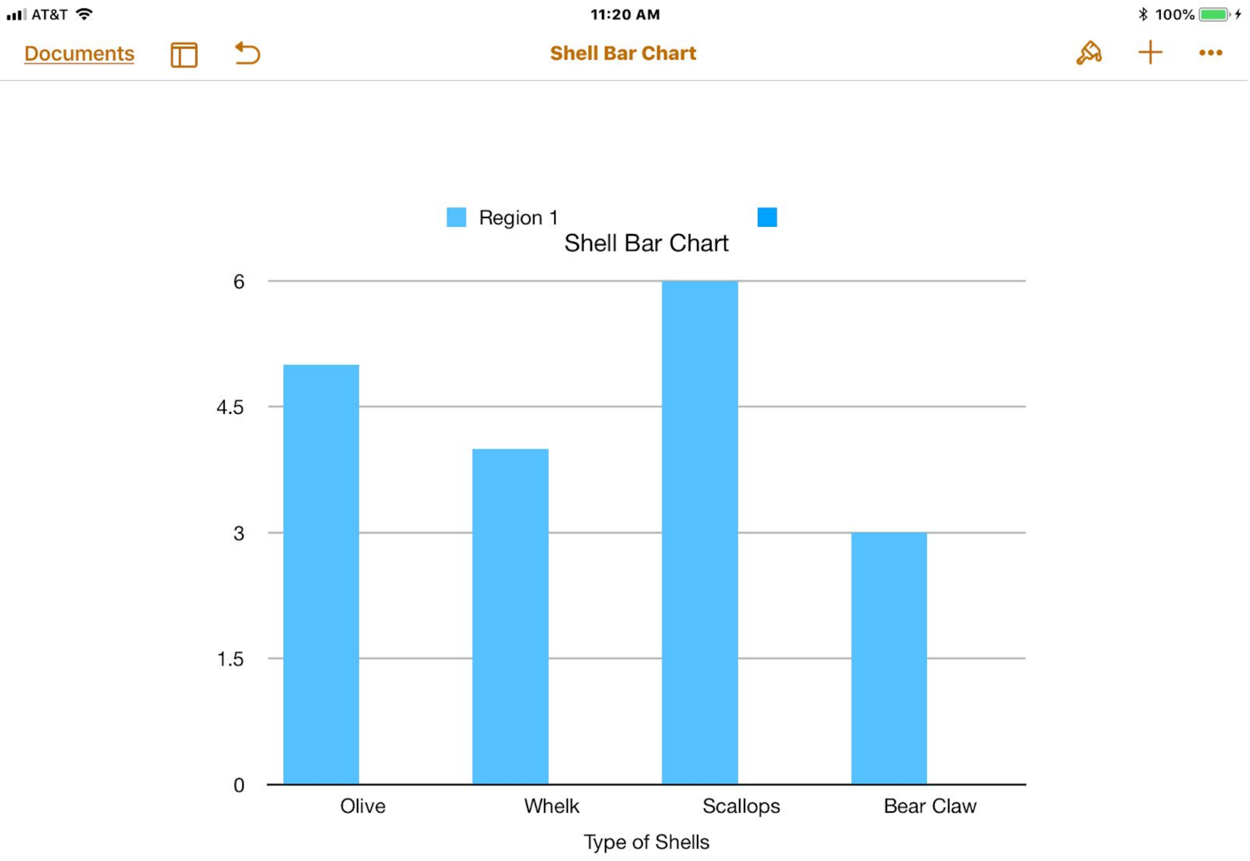
1.Scatter plot

Scatterplot is a classic and fundamental plot used to study the relationship between two variables. If you have multiple groups in your data you may want to visualise each group in a different color.

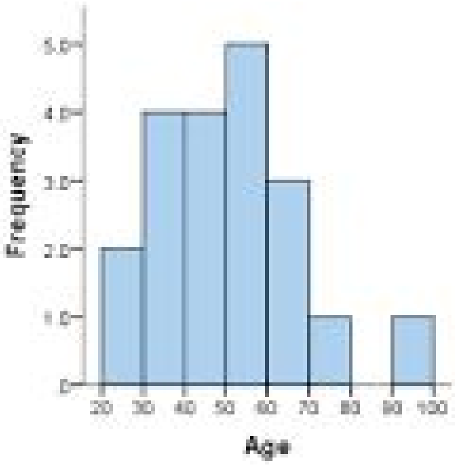


2.Bar plot

A **bar plot** is a **plot** that presents categorical data with rectangular **bars** with lengths proportional to the values that they represent. A **bar plot** shows comparisons among discrete categories. One axis of the **plot** shows the specific categories being compared, and the other axis represents a measured value.

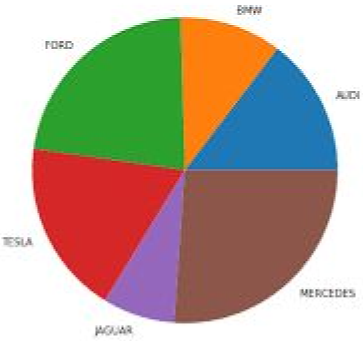


3.Histogram



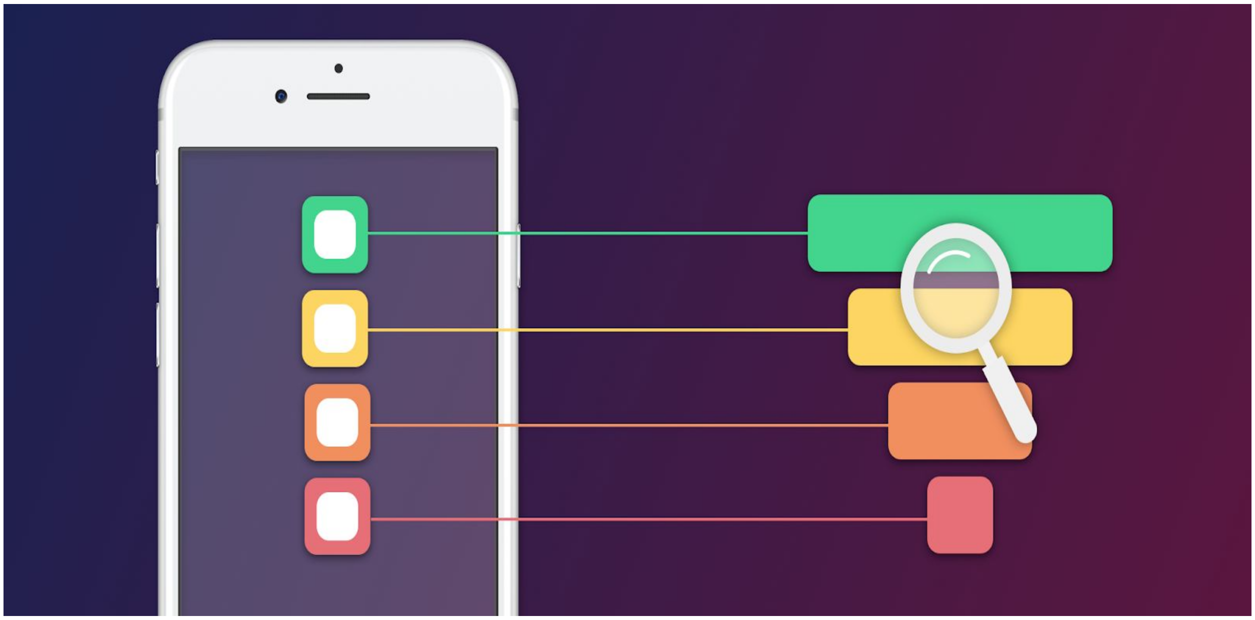
A **histogram** is a graphical display of data using bars of different heights. In a **histogram**, each bar groups numbers into ranges. Taller bars show that more data falls in that range. A **histogram** displays the shape and spread of continuous sample data.

4.Pie Chart



Plot a **pie chart** in **Python** using Matplotlib. A **Pie Chart** is a circular statistical **plot** that can display only one series of data. The area of the **chart** is the total percentage of the given data. The area of slices of the **pie** represents the percentage of the parts of the data. The slices of **pie** are called wedges.

Data Analysis for Mobile Apps:



Key Points do Analysis of mobile Apps

Social Networking:

That is why **social networks** for **apps** are so important. **Social media** helps people establish better relationships with their family and friends, and now the **networking** sites also show their significance for **apps**. Mobile devices are gaining more and more space amongst people when compared to other ways of internet access

Free apps:

An app is computer software, or a program, most commonly a small, specific one used for mobile devices. The term app originally referred to any mobile or desktop application, but as more app stores have emerged to sell mobile apps to smartphone and tablet users, the term has evolved to refer to small programs that can be downloaded and installed all at once.
There are thousands of apps designed to run on today's smartphones and tablets. Some apps can be downloaded for free, while others must be purchased from an app store.

Paid apps:
Paid apps - applications that need money for setup, in-**app** purchases or subscription to use them. Also, there are free aplications, that need no purchases. They have another model of monetization for example, in **application** advertising and so on

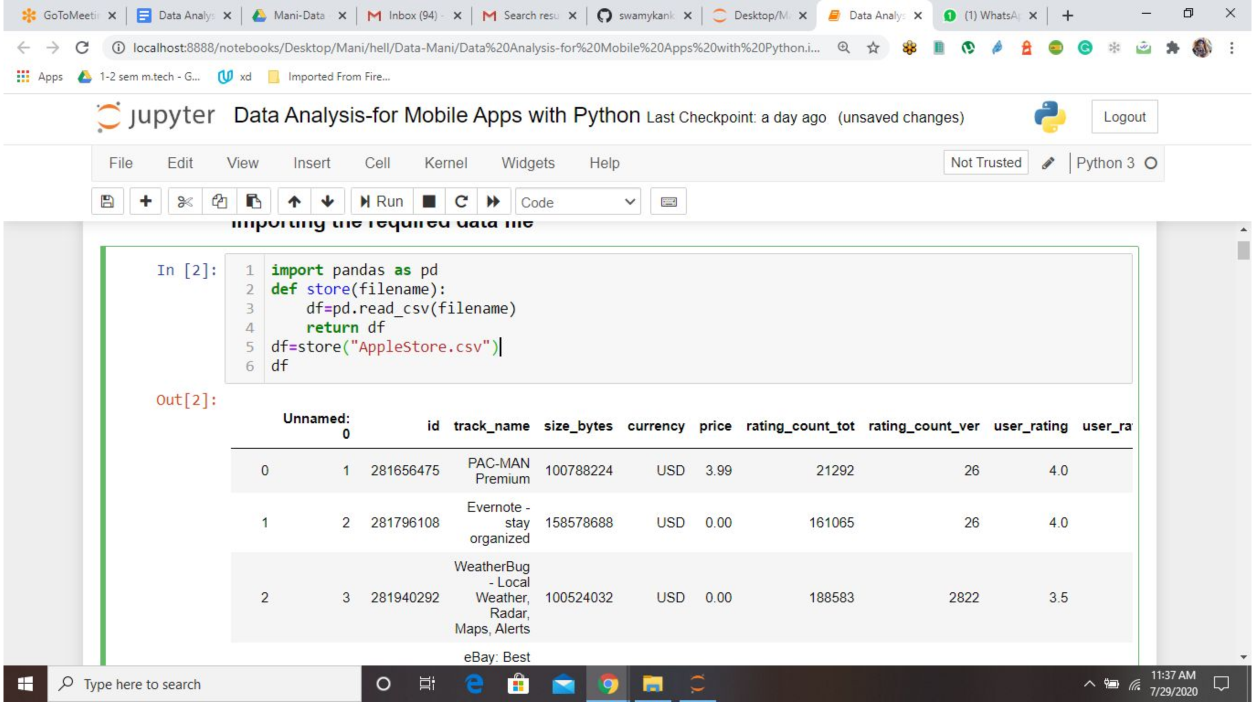
Gaming Apps:

Gaming is the running of specialized applications known as electronic **games** or video **games** on **game** consoles like X-box and Playstation or on personal computers (in which case the activity is known as online **gaming**). ... In its most sophisticated form, a **gaming** interface can constitute a form of virtual reality.

Moving on the code

# Data Analysis-for Mobile Apps with Python

Importing the required data file


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter
Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
Type here to search
11:37 AM
7/29/2020

In [2]:

```
import pandas as pd
def store(filename):
    df=pd.read_csv(filename)
    return df
df=store("AppleStore.csv")
df
```

Out[2]:

	Unnamed: 0	id	track_name	size_bytes	currency	price	rating_count_tot	rating_count_ver	user_rating	user_ra
0	1	281656475	PAC-MAN Premium	100788224	USD	3.99	21292	26	4.0	
1	2	281796108	Evernote - stay organized	158578688	USD	0.00	161065	26	4.0	
2	3	281940292	WeatherBug - Local Weather, Radar, Maps, Alerts	100524032	USD	0.00	188583	2822	3.5	
			eBay: Best							

Problem 1: Calculate the average rating for free apps

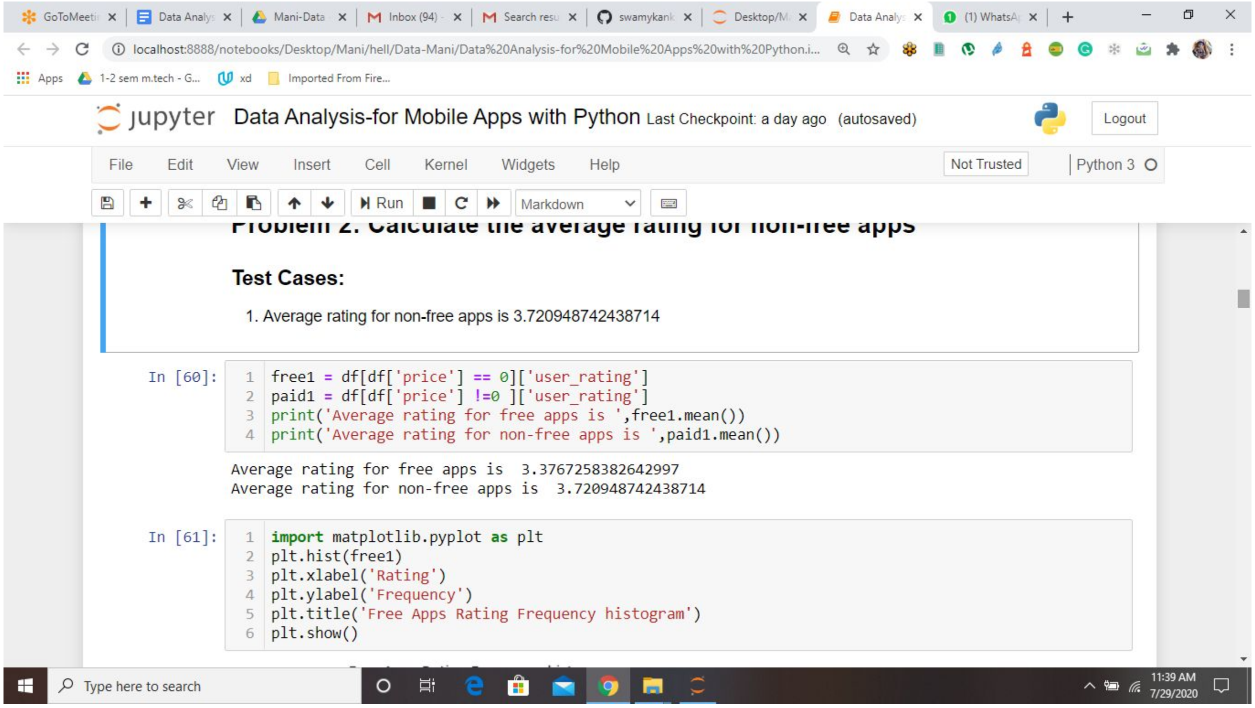
Test Cases:

1. **Average rating for free apps is 3.3767258382642997**

Problem 2: Calculate the average rating for non-free apps

Test Cases:

1. **Average rating for non-free apps is 3.720948742438714**



Problem 2: Calculate the average rating for non-free apps

Test Cases:

1. Average rating for non-free apps is 3.720948742438714

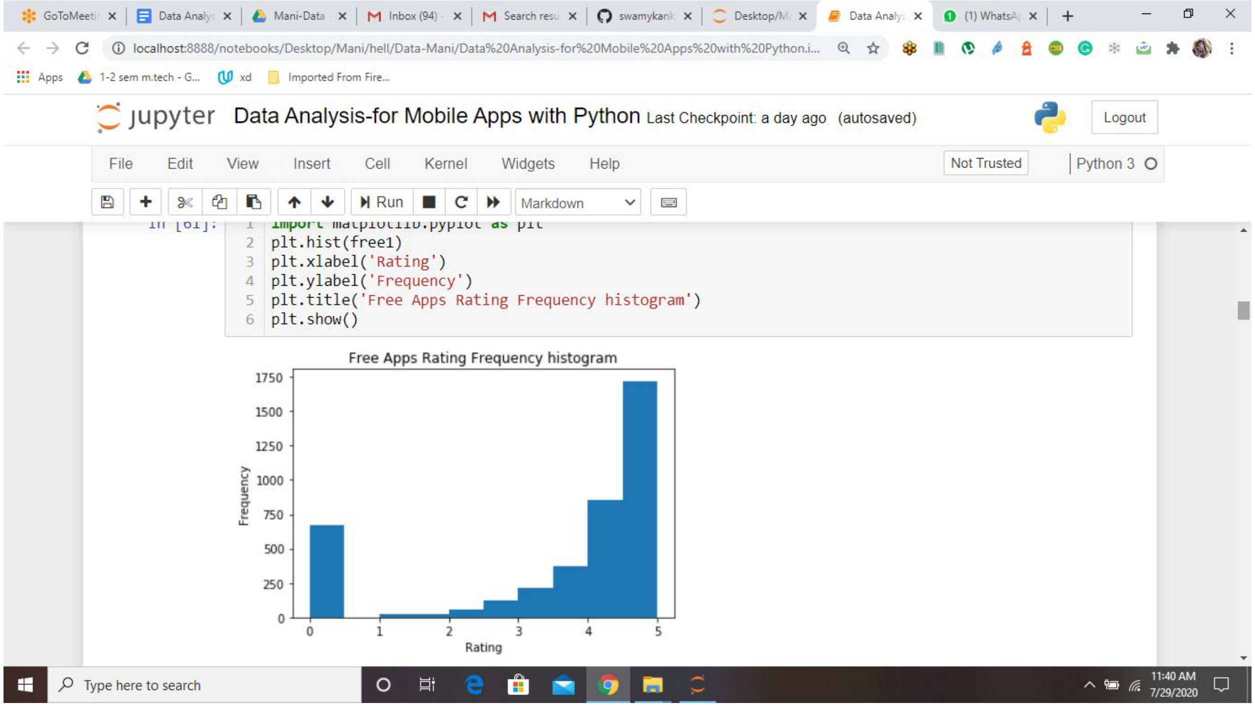
In [60]:

```python
free1 = df[df['price'] == 0]['user_rating']
paid1 = df[df['price'] !=0 ]['user_rating']
print('Average rating for free apps is ',free1.mean())
print('Average rating for non-free apps is ',paid1.mean())
```

```
Average rating for free apps is  3.3767258382642997
Average rating for non-free apps is  3.720948742438714
```

In [61]:

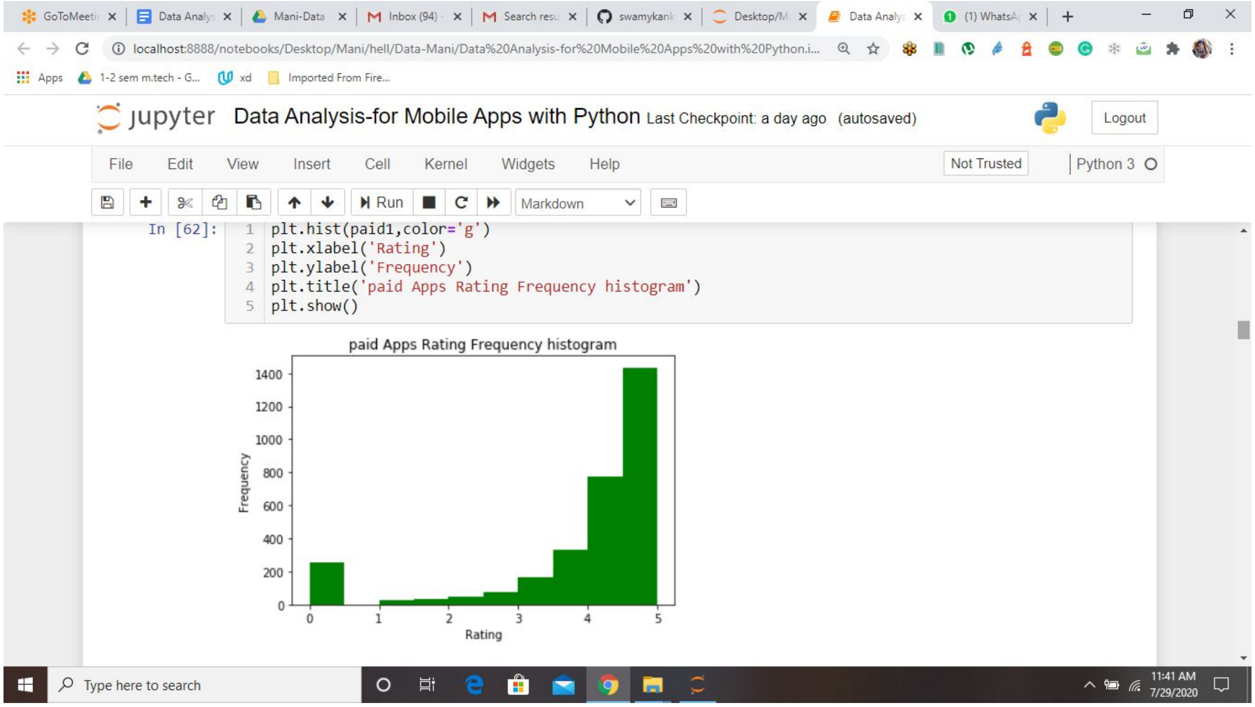
```python
import matplotlib.pyplot as plt
plt.hist(free1)
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.title('Free Apps Rating Frequency histogram')
plt.show()
```



jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown

```
In [61]: import matplotlib.pyplot as plt
         plt.hist(free1)
         plt.xlabel('Rating')
         plt.ylabel('Frequency')
         plt.title('Free Apps Rating Frequency histogram')
         plt.show()
```

Free Apps Rating Frequency histogram
1750
1500
1250
1000
750
500
250
0
Frequency
0
1
2
3
4
5
Rating
Type here to search
11:40 AM
7/29/2020



jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown

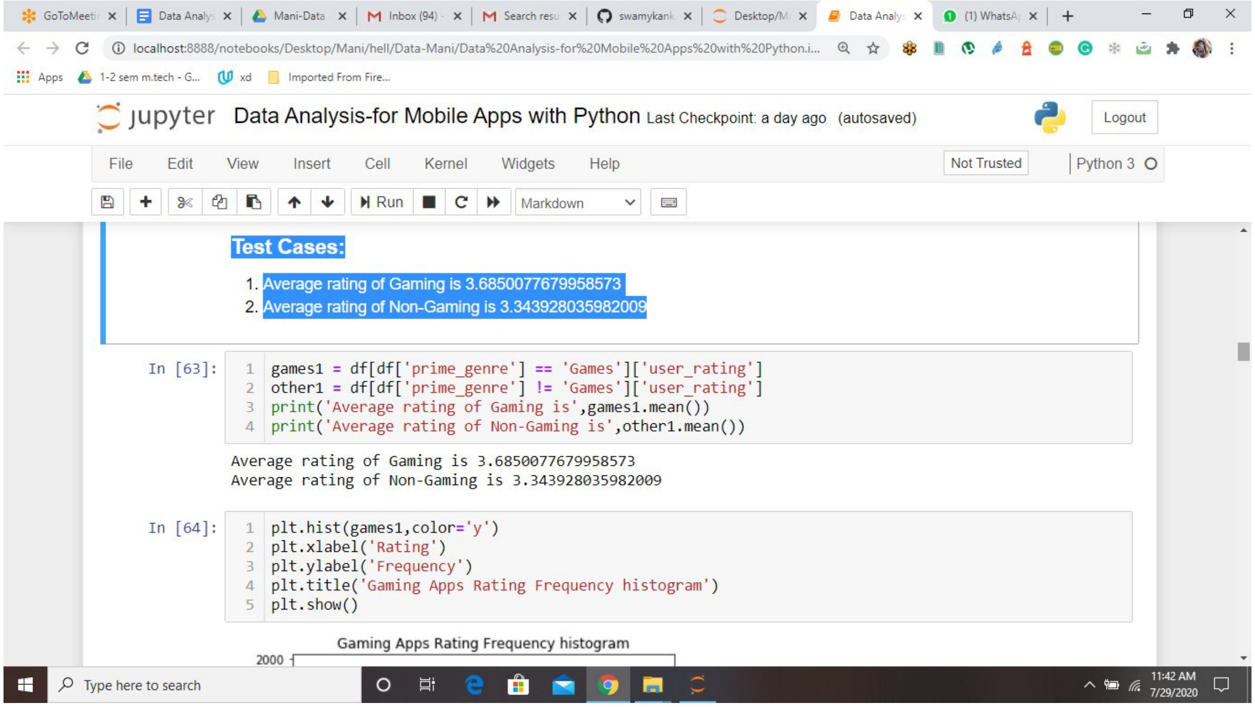
```
In [62]: plt.hist(paid1,color='g')
         plt.xlabel('Rating')
         plt.ylabel('Frequency')
         plt.title('paid Apps Rating Frequency histogram')
         plt.show()
```

paid Apps Rating Frequency histogram
1400
1200
1000
800
600
400
200
0
Frequency
0
1
2
3
4
5
Rating
Type here to search
11:41 AM
7/29/2020

Problem 3: Calculate the average rating of Gaming and Non-Gaming apps

Test Cases:

1. Average rating of Gaming is 3.6850077679958573
2. Average rating of Non-Gaming is 3.343928035982009



jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Logout
Run
Markdown
Type here to search
11:42 AM
7/29/2020

Test Cases:

1. Average rating of Gaming is 3.6850077679958573
2. Average rating of Non-Gaming is 3.343928035982009

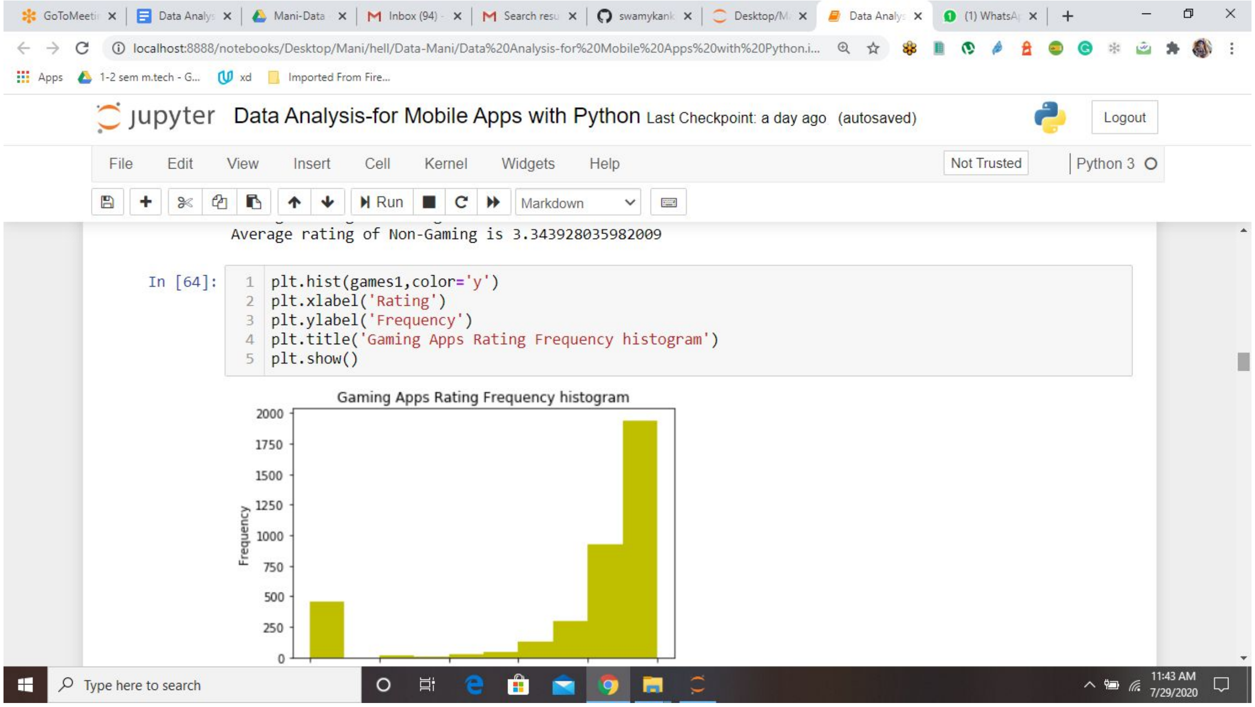
In [63]:

```
games1 = df[df['prime_genre'] == 'Games']['user_rating']
other1 = df[df['prime_genre'] != 'Games']['user_rating']
print('Average rating of Gaming is',games1.mean())
print('Average rating of Non-Gaming is',other1.mean())
```

```
Average rating of Gaming is 3.6850077679958573
Average rating of Non-Gaming is 3.343928035982009
```

In [64]:

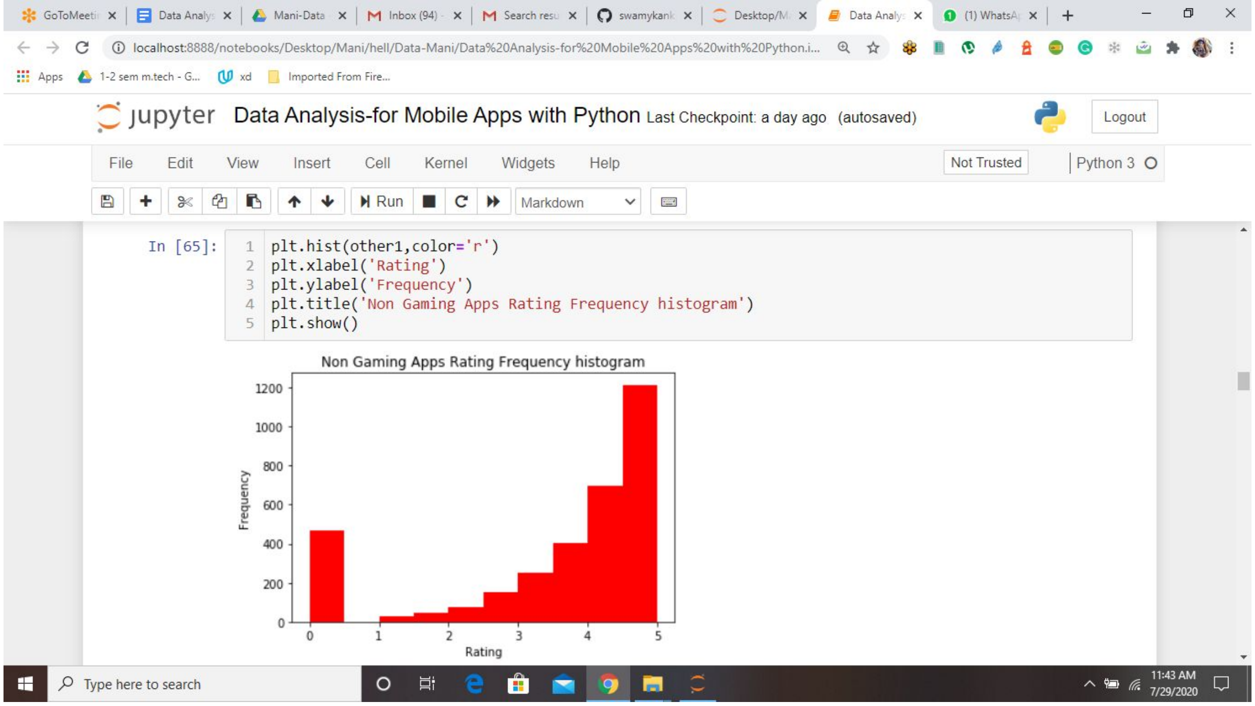
```
plt.hist(games1,color='y')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.title('Gaming Apps Rating Frequency histogram')
plt.show()
```



```
Average rating of Non-Gaming is 3.343928035982009
```

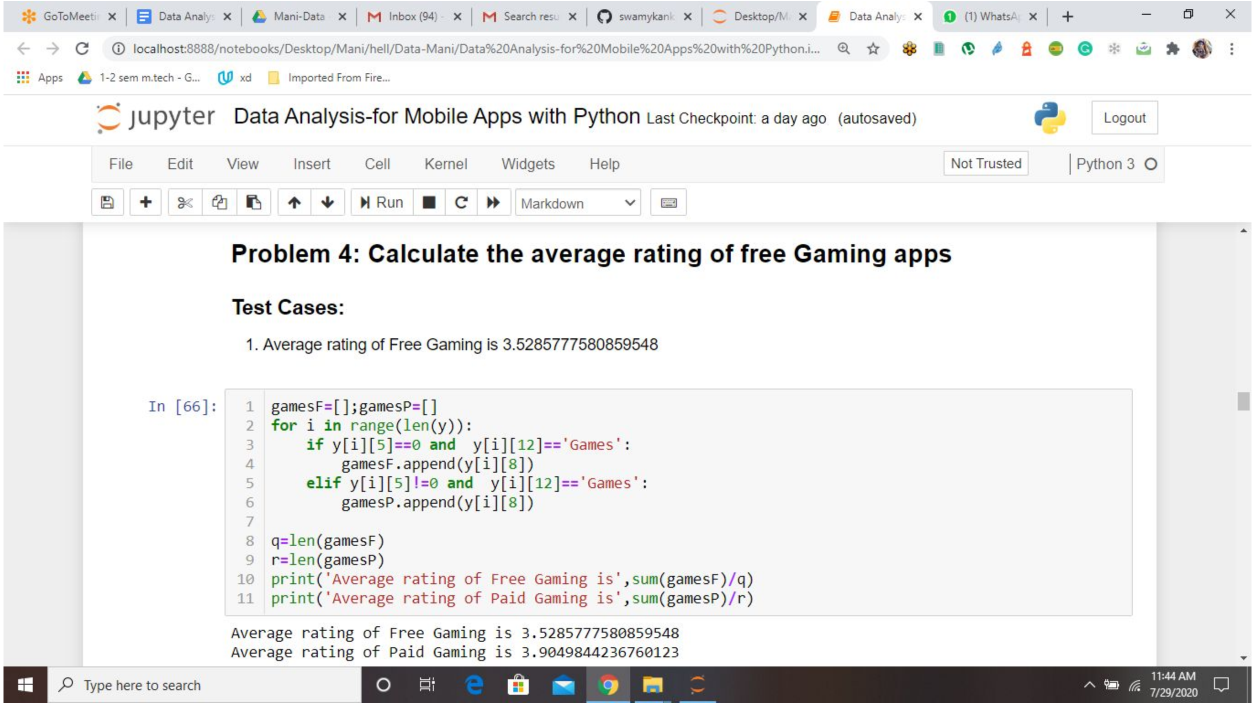
In [64]:

```python
plt.hist(games1,color='y')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.title('Gaming Apps Rating Frequency histogram')
plt.show()
```



In [65]:

```python
plt.hist(other1,color='r')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.title('Non Gaming Apps Rating Frequency histogram')
plt.show()
```


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:44 AM
7/29/2020

Problem 4: Calculate the average rating of free Gaming apps

Test Cases:

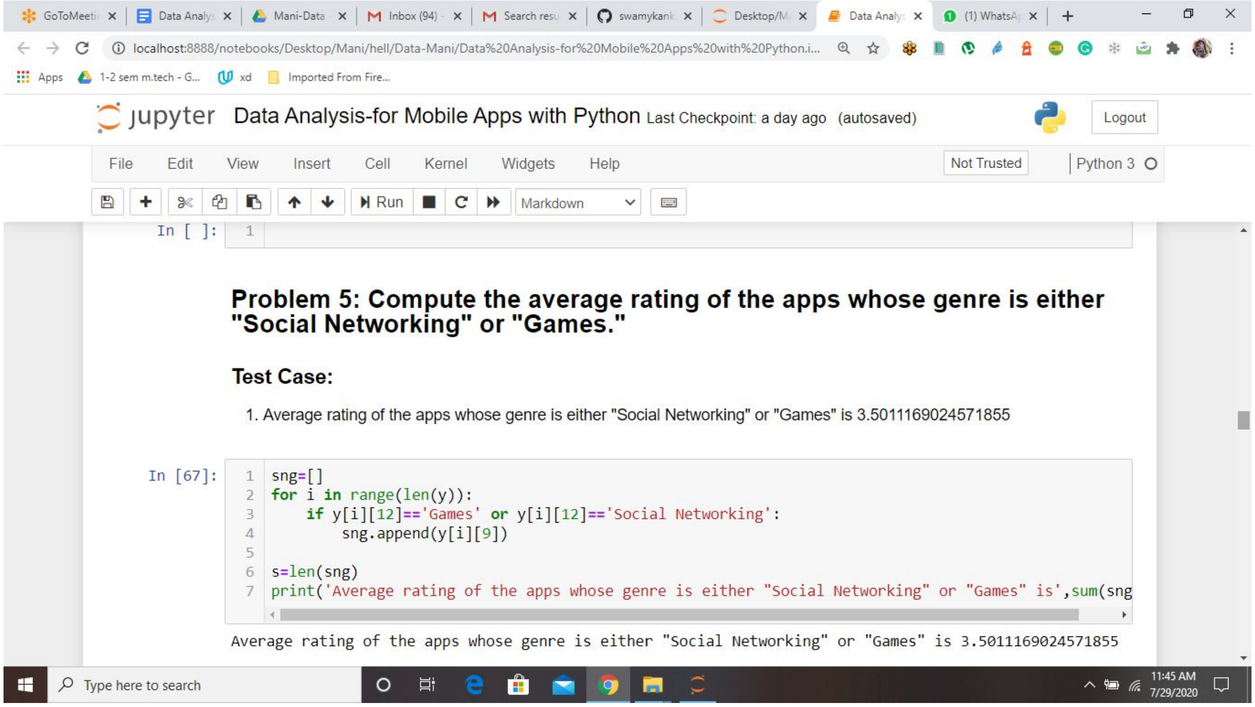
1. Average rating of Free Gaming is 3.5285777580859548

In [66]:

```python
gamesF=[];gamesP=[]
for i in range(len(y)):
    if y[i][5]==0 and  y[i][12]=='Games':
        gamesF.append(y[i][8])
    elif y[i][5]!=0 and  y[i][12]=='Games':
        gamesP.append(y[i][8])

q=len(gamesF)
r=len(gamesP)
print('Average rating of Free Gaming is',sum(gamesF)/q)
print('Average rating of Paid Gaming is',sum(gamesP)/r)
```

```
Average rating of Free Gaming is 3.5285777580859548
Average rating of Paid Gaming is 3.9049844236760123
```


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:45 AM
7/29/2020

In [ ]:

Problem 5: Compute the average rating of the apps whose genre is either "Social Networking" or "Games."

Test Case:

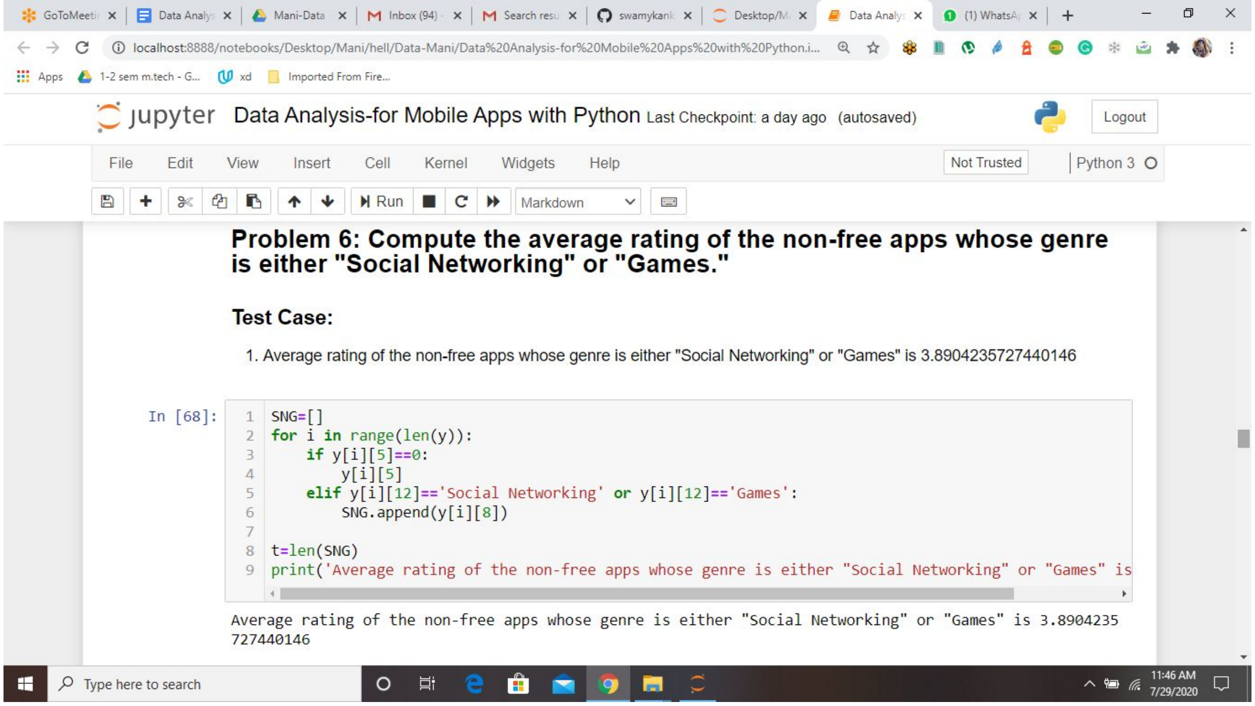
1. Average rating of the apps whose genre is either "Social Networking" or "Games" is 3.5011169024571855

In [67]:

```python
sng=[]
for i in range(len(y)):
    if y[i][12]=='Games' or y[i][12]=='Social Networking':
        sng.append(y[i][9])

s=len(sng)
print('Average rating of the apps whose genre is either "Social Networking" or "Games" is',sum(sng
```

```
Average rating of the apps whose genre is either "Social Networking" or "Games" is 3.5011169024571855
```


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:46 AM
7/29/2020

Problem 6: Compute the average rating of the non-free apps whose genre is either "Social Networking" or "Games."

Test Case:

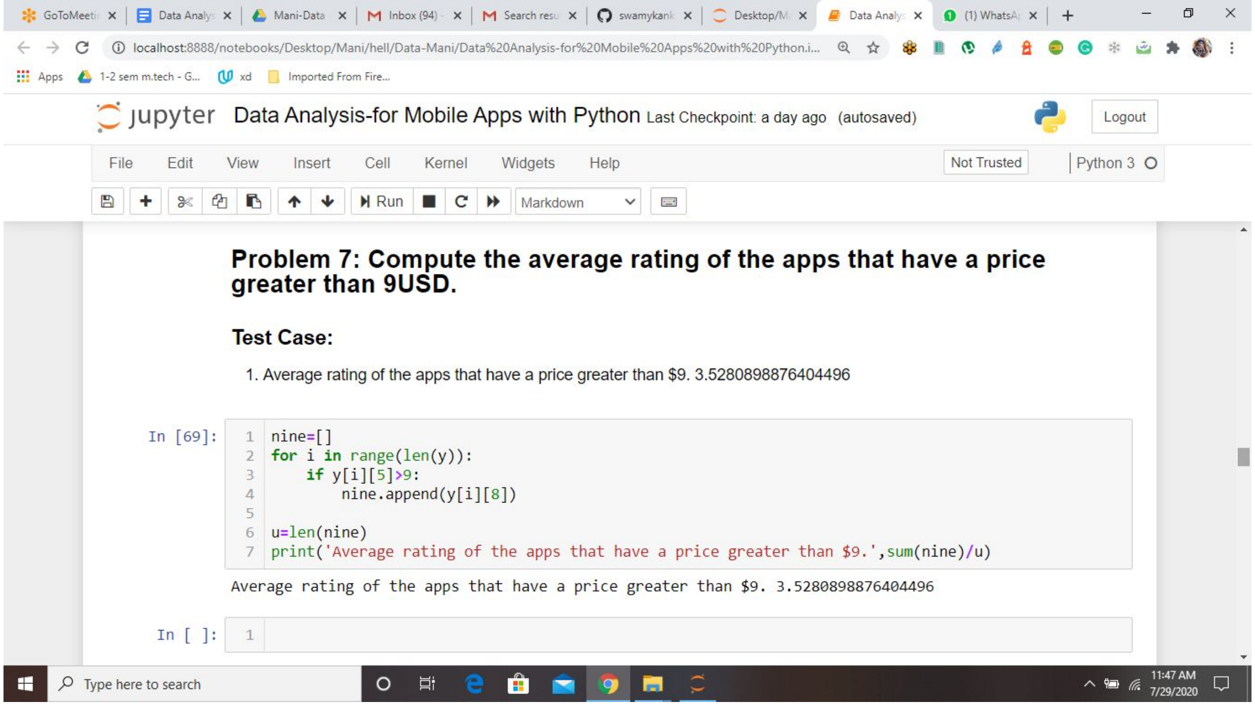
1. Average rating of the non-free apps whose genre is either "Social Networking" or "Games" is 3.8904235727440146

In [68]:

```
SNG=[]
for i in range(len(y)):
    if y[i][5]==0:
        y[i][5]
    elif y[i][12]=='Social Networking' or y[i][12]=='Games':
        SNG.append(y[i][8])

t=len(SNG)
print('Average rating of the non-free apps whose genre is either "Social Networking" or "Games" is
```

```
Average rating of the non-free apps whose genre is either "Social Networking" or "Games" is 3.8904235
727440146
```


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:47 AM
7/29/2020

Problem 7: Compute the average rating of the apps that have a price greater than 9USD.

Test Case:

1. Average rating of the apps that have a price greater than $9. 3.5280898876404496

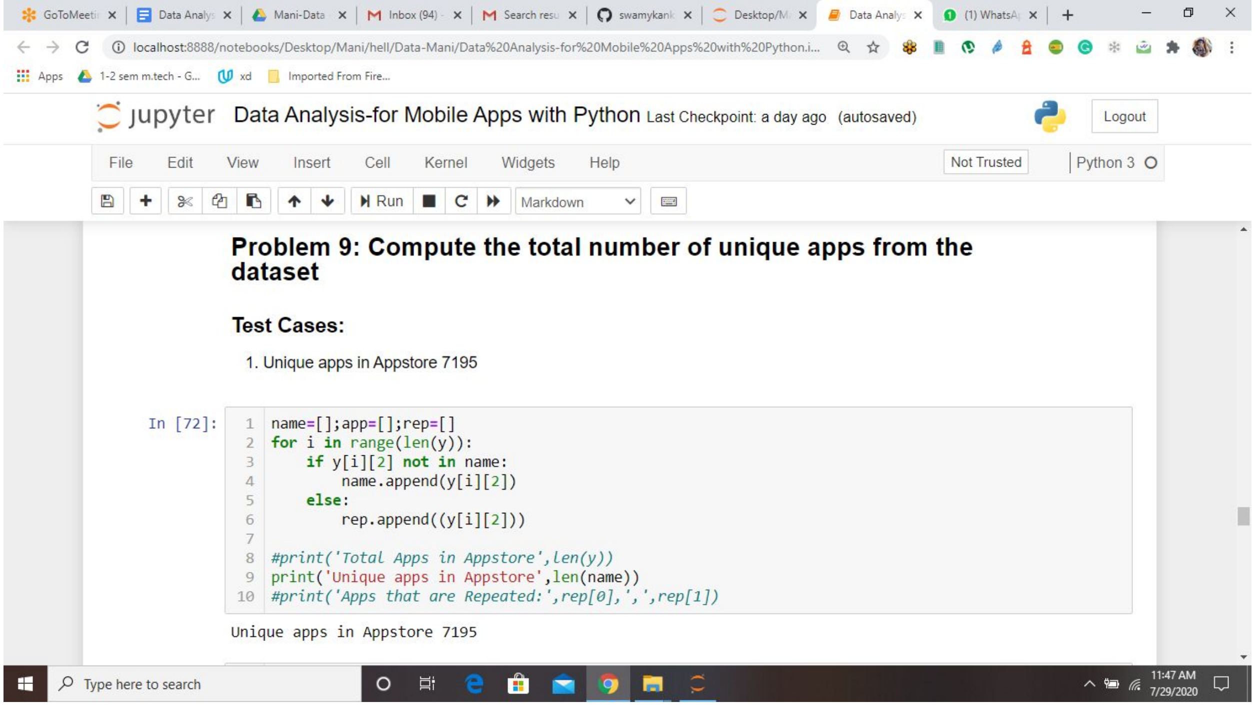
In [69]:

```
nine=[]
for i in range(len(y)):
    if y[i][5]>9:
        nine.append(y[i][8])

u=len(nine)
print('Average rating of the apps that have a price greater than $9.',sum(nine)/u)
```

```
Average rating of the apps that have a price greater than $9. 3.5280898876404496
```

In [ ]:


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter
Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:47 AM
7/29/2020

Problem 9: Compute the total number of unique apps from the dataset

Test Cases:

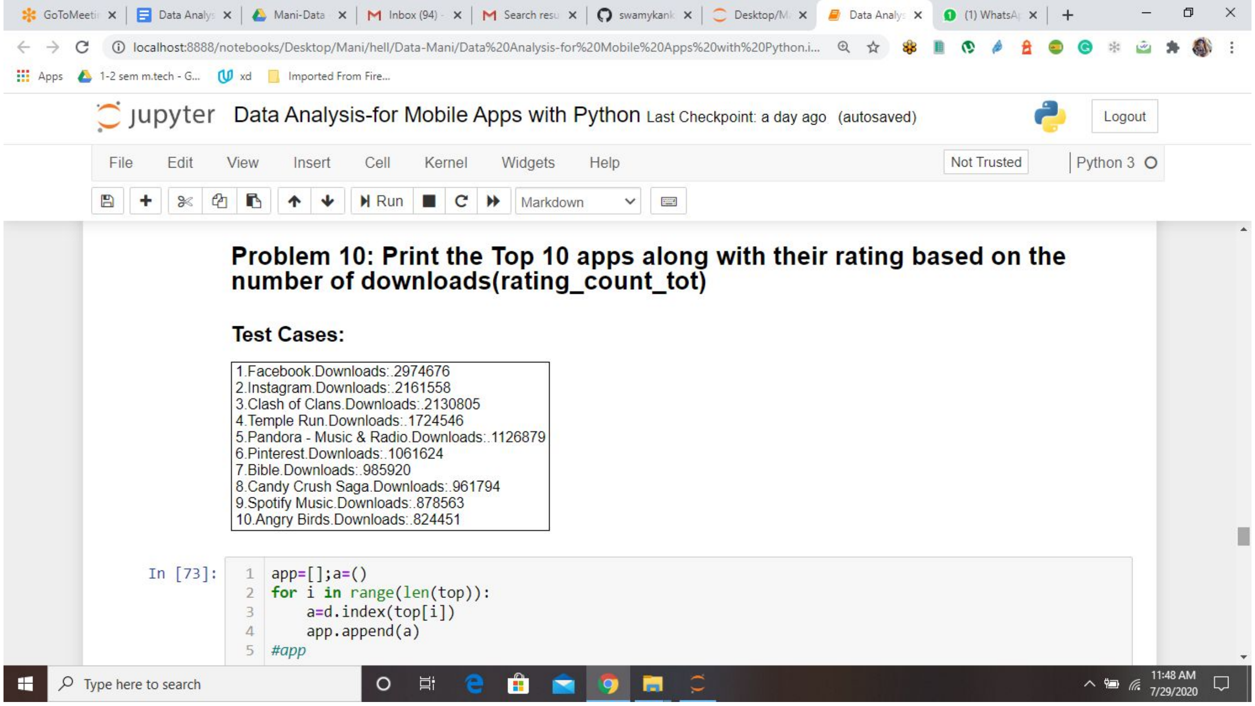
1. Unique apps in Appstore 7195

In [72]:

```
name=[];app=[];rep=[]
for i in range(len(y)):
    if y[i][2] not in name:
        name.append(y[i][2])
    else:
        rep.append((y[i][2]))

#print('Total Apps in Appstore',len(y))
print('Unique apps in Appstore',len(name))
#print('Apps that are Repeated:',rep[0],',',rep[1])
```

```
Unique apps in Appstore 7195
```


localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter
Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:48 AM
7/29/2020

Problem 10: Print the Top 10 apps along with their rating based on the number of downloads(rating_count_tot)

Test Cases:

```
1.Facebook.Downloads:.2974676
2.Instagram.Downloads:.2161558
3.Clash of Clans.Downloads:.2130805
4.Temple Run.Downloads:.1724546
5.Pandora - Music & Radio.Downloads:.1126879
6.Pinterest.Downloads:.1061624
7.Bible.Downloads:.985920
8.Candy Crush Saga.Downloads:.961794
9.Spotify Music.Downloads:.878563
10.Angry Birds.Downloads:.824451
```

In [73]:

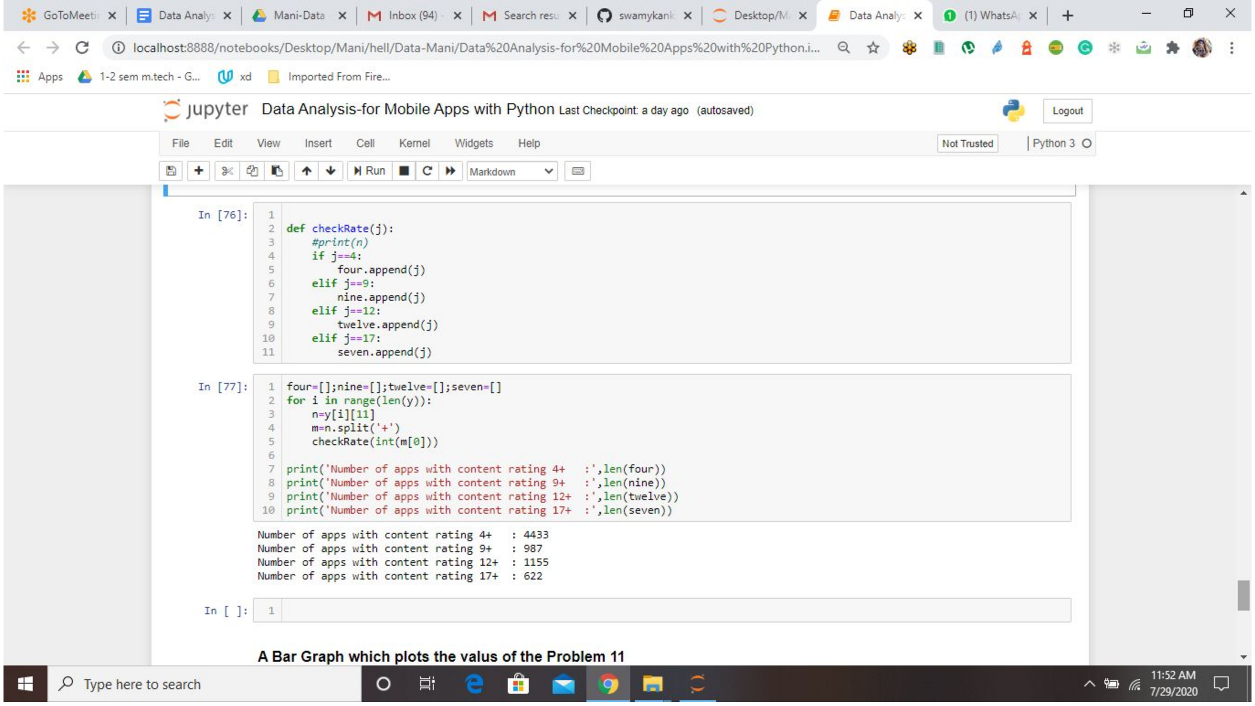
```
app=[];a=()
for i in range(len(top)):
    a=d.index(top[i])
    app.append(a)
#app
```

Problem 11: Categorise the dataset based on content rating into the following

- Number of apps with content rating 4+
- Number of apps with content rating 9+
- Number of apps with content rating 12+
- Number of apps with content rating 17+

Test Cases:

1. Number of apps with content rating 4+ : 4433
2. Number of apps with content rating 9+ : 987
3. Number of apps with content rating 12+ : 1155
4. Number of apps with content rating 17+ : 622



localhost:8888/notebooks/Desktop/Mani/hell/Data-Mani/Data%20Analysis-for%20Mobile%20Apps%20with%20Python.i...
jupyter Data Analysis-for Mobile Apps with Python Last Checkpoint: a day ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
Type here to search
11:52 AM
7/29/2020

In [76]:

```python

def checkRate(j):
    #print(n)
    if j==4:
        four.append(j)
    elif j==9:
        nine.append(j)
    elif j==12:
        twelve.append(j)
    elif j==17:
        seven.append(j)
```

In [77]:

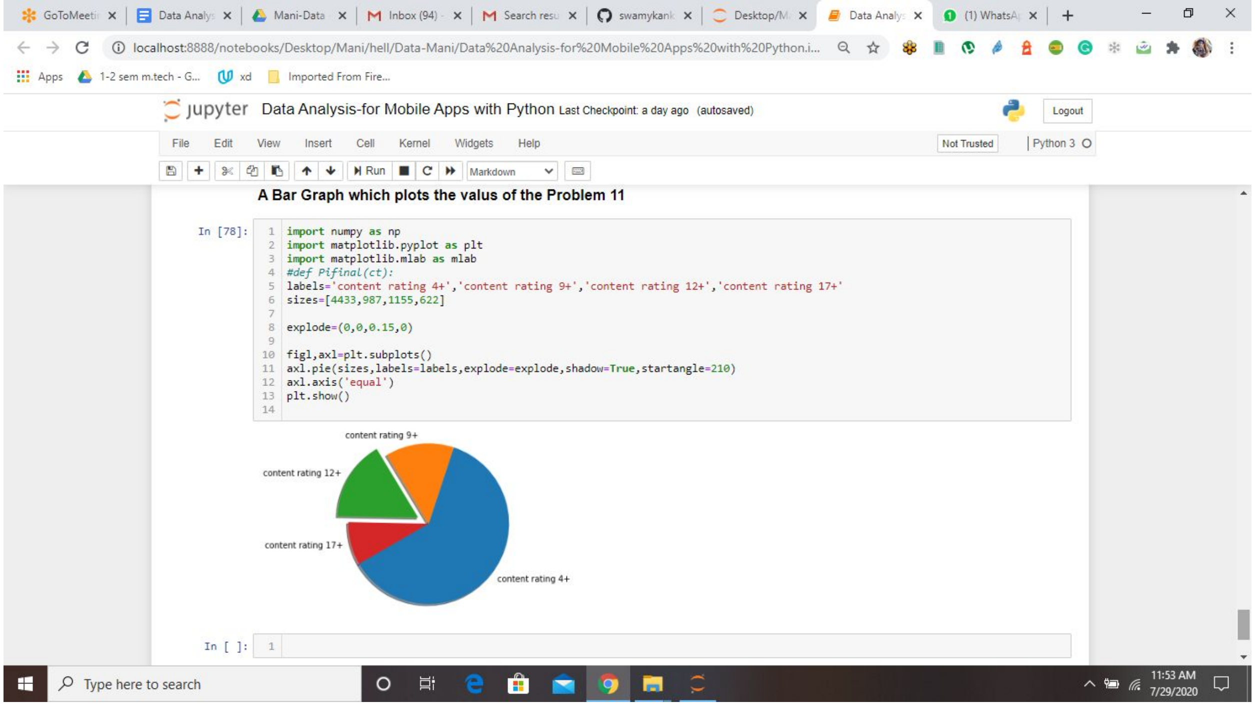
```python
four=[];nine=[];twelve=[];seven=[]
for i in range(len(y)):
    n=y[i][11]
    m=n.split('+')
    checkRate(int(m[0]))

print('Number of apps with content rating 4+   :',len(four))
print('Number of apps with content rating 9+   :',len(nine))
print('Number of apps with content rating 12+  :',len(twelve))
print('Number of apps with content rating 17+  :',len(seven))
```

```
Number of apps with content rating 4+   : 4433
Number of apps with content rating 9+   : 987
Number of apps with content rating 12+  : 1155
Number of apps with content rating 17+  : 622
```

In [ ]:

A Bar Graph which plots the valus of the Problem 11



A Bar Graph which plots the valus of the Problem 11

In [78]:

```python
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.mlab as mlab
#def Pifinal(ct):
labels='content rating 4+','content rating 9+','content rating 12+','content rating 17+'
sizes=[4433,987,1155,622]

explode=(0,0,0.15,0)

figl,axl=plt.subplots()
axl.pie(sizes,labels=labels,explode=explode,shadow=True,startangle=210)
axl.axis('equal')
plt.show()
```