CASE STUDY (PART – 2)

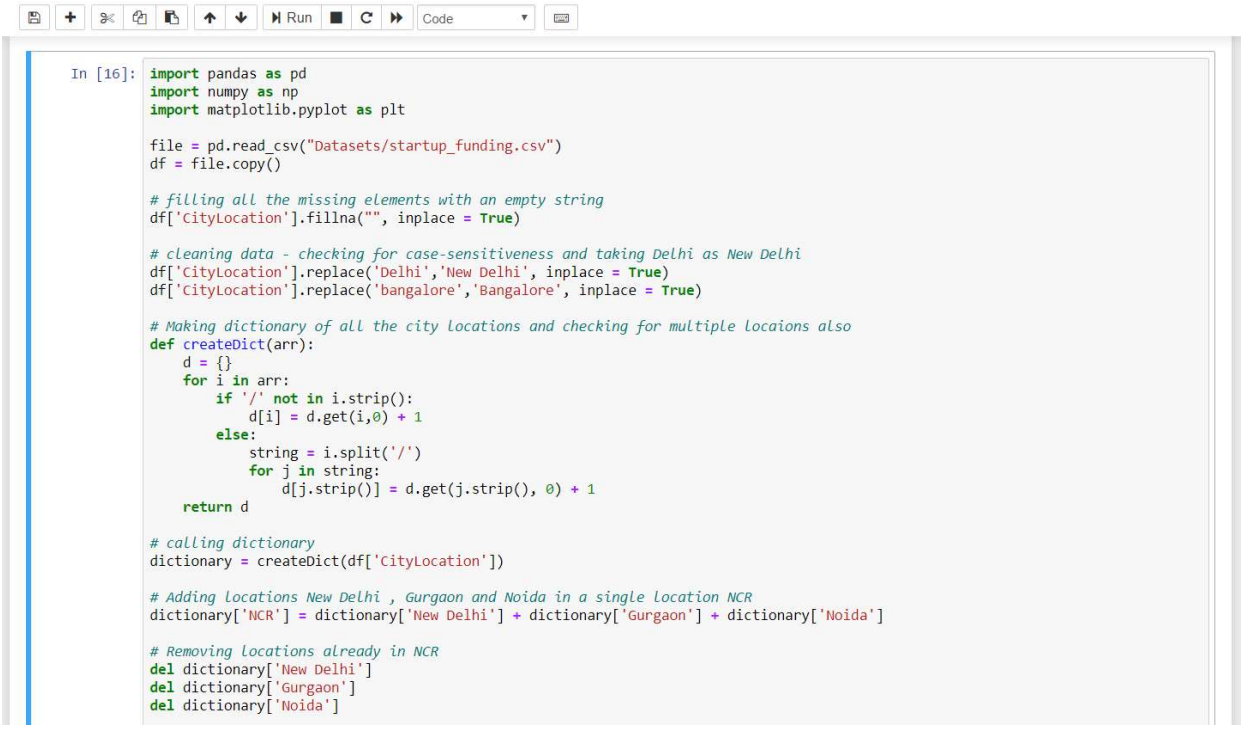
DATASET NAME – Startup_Funding.csv

QUESTIONS

QUES. 1)

Your Friend has developed the Product and he wants to establish the product startup and he is searching for a perfect location where getting the investment has a high chance. But due to its financial restriction, he can choose only between three locations - Bangalore, Mumbai, and NCR. As a friend, you want to help your friend deciding the location. NCR include Gurgaon, Noida and New Delhi. Find the location where the most number of funding is done. That means, find the location where startups has received funding maximum number of times. Plot the bar graph between location and number of funding. Take city name "Delhi" as "New Delhi". Check the case-sensitiveness of cities also. That means, at some place instead of "Bangalore", "bangalore" is given. Take city name as "Bangalore". For few startups multiple locations are given, one Indian and one Foreign. Consider the startup if any one of the city lies in given locations.

CODE SNIPPETS –

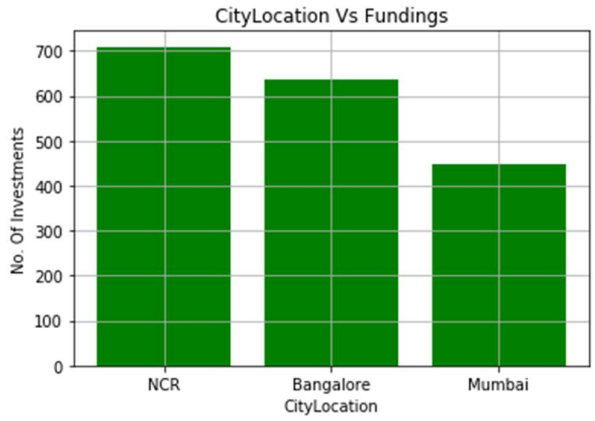


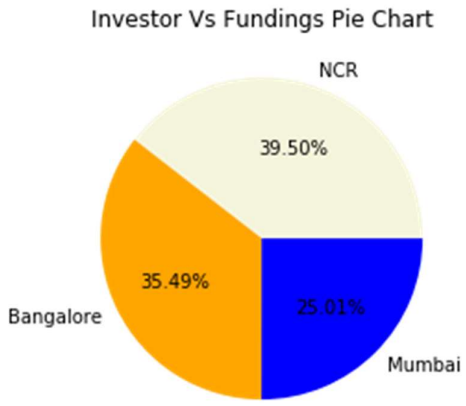
```
In [16]:
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

file = pd.read_csv("Datasets/startup_funding.csv")
df = file.copy()

# filling all the missing elements with an empty string
df['CityLocation'].fillna("", inplace = True)

# cleaning data - checking for case-sensitiveness and taking Delhi as New Delhi
df['CityLocation'].replace('Delhi','New Delhi', inplace = True)
df['CityLocation'].replace('bangalore','Bangalore', inplace = True)

# Making dictionary of all the city locations and checking for multiple locaions also
def createDict(arr):
    d = {}
    for i in arr:
        if '/' not in i.strip():
            d[i] = d.get(i,0) + 1
        else:
            string = i.split('/')
            for j in string:
                d[j.strip()] = d.get(j.strip(), 0) + 1
    return d

# calling dictionary
dictionary = createDict(df['CityLocation'])

# Adding locations New Delhi , Gurgaon and Noida in a single location NCR
dictionary['NCR'] = dictionary['New Delhi'] + dictionary['Gurgaon'] + dictionary['Noida']

# Removing locations already in NCR
del dictionary['New Delhi']
del dictionary['Gurgaon']
del dictionary['Noida']
```

```
# Removing locations already in NCR
del dictionary['New Delhi']
del dictionary['Gurgaon']
del dictionary['Noida']

# sorting the Dictionary in descending order according to it's values
sorted_d = sorted(dictionary.items(), key=lambda kv: kv[1],reverse = True)
city_count = []
cityname= []

# adding top 3 cities to the lists
for i in range(3):
    print(sorted_d[i][0],sorted_d[i][1])
    city_count.append(sorted_d[i][1])
    cityname.append(sorted_d[i][0])

y = city_count
x = cityname

# plotting the bar graph for 3 cities
plt.bar(x,y, color = 'green')
plt.xlabel("CityLocation")
plt.ylabel("No. Of Investments")
plt.title("CityLocation Vs Fundings")
plt.grid()
plt.show()

# plotting the pie chart b/w top 5 investors and number of fundings.
color = ["beige", "orange", "blue", "pink", "grey"]
plt.pie(y, labels = x, colors = color, autopct = "%.2f%%")
plt.title("Investor Vs Fundings Pie Chart")
plt.show()
```

```
NCR 709
Bangalore 637
Mumbai 449
```

GRAPHS –





CODE EXPLANATION –

To find the location , where most number of funding is done , I have included the libraries required that are numpy , pandas , matplotlib.pyplot then –

1) Read the file using pandas read_csv function
2) Cleaned the data by checking case sensitiveness , checking errors in names and fill the NaN values with empty string.
3) Then , I have made a dictionary , which is consisting location as key and number of funding's as values. This dictionary is also checking for those startups, which are present at multiple locations.
4) This is done by using strip() function.
5) Since, NCR consists of NEW DELHI , GURGAON and NOIDA , so I have added all their counts in NCR.
6) Now, I have removed these un – necessary locations.
7) Sorted the dictionary according to the number of fundings in descending order.
8) Printed and plotted graph for only given locations that are NCR, BANGALORE AND MUMBAI.

RESULT AND JUSTIFICATION –

As the result of the code shows and according to graphs also,

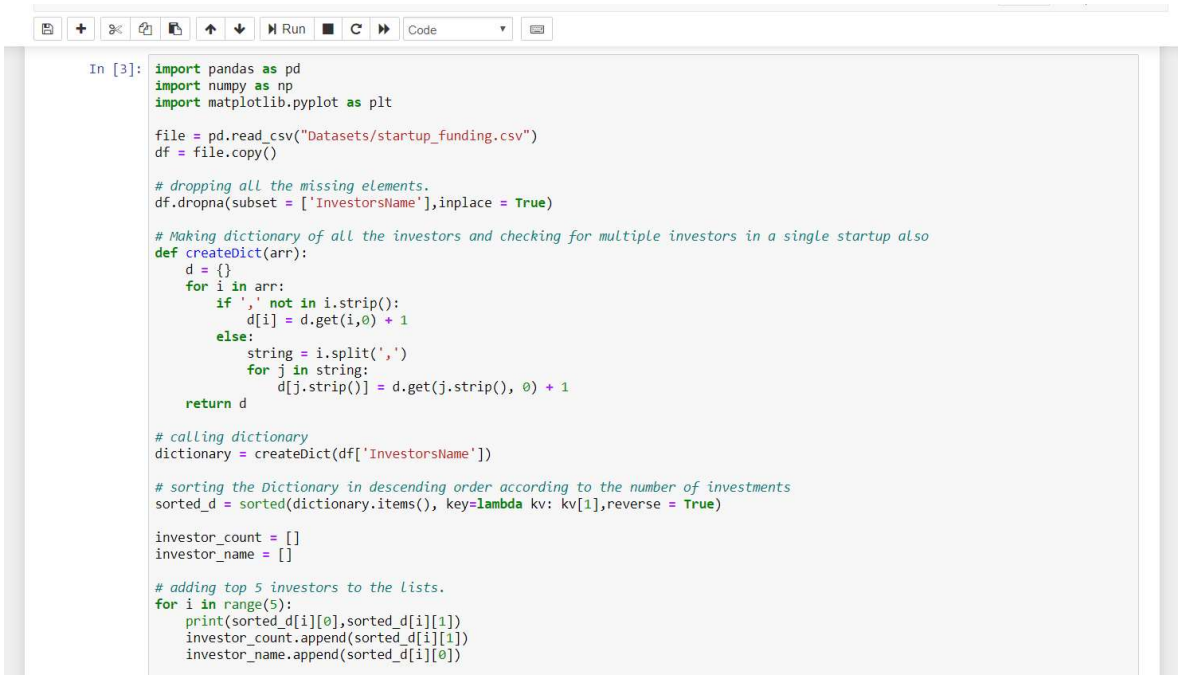
The Top location is ***NCR*** which has maximum number of funding's that is ***703.***

So, I would recommend my friend to establish his start-up in **NCR LOCATION** because that's where maximum chances of getting investment.

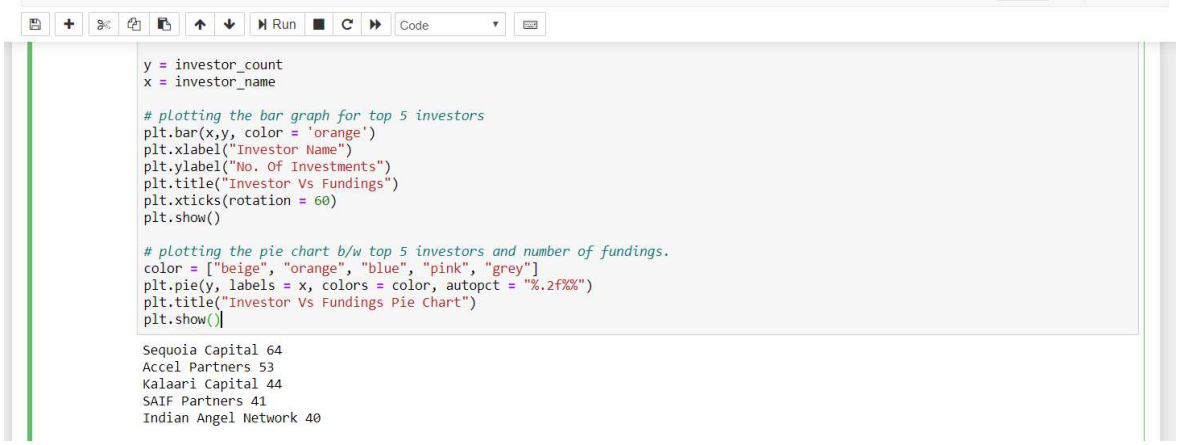
QUES. 2)

Even after trying for so many times, your friend's startup could not find the investment. So you decided to take this matter in your hand and try to find the list of investors who probably can invest in your friend's startup. Your list will increase the chance of your friend startup getting some initial investment by contacting these investors. Find the top 5 investors who have invested maximum number of times (consider repeat investments in one company also). In a startup, multiple investors might have invested. So consider each investor for that startup. Ignore undisclosed investors.

CODE SNIPPETS –



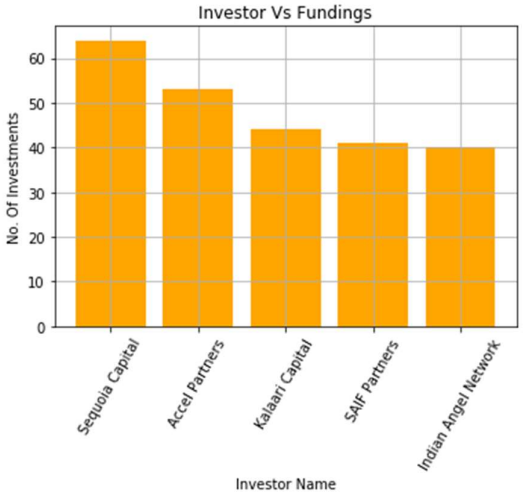
```
In [3]:
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

file = pd.read_csv("Datasets/startup_funding.csv")
df = file.copy()

# dropping all the missing elements.
df.dropna(subset = ['InvestorsName'],inplace = True)

# Making dictionary of all the investors and checking for multiple investors in a single startup also
def createDict(arr):
    d = {}
    for i in arr:
        if ',' not in i.strip():
            d[i] = d.get(i,0) + 1
        else:
            string = i.split(',')
            for j in string:
                d[j.strip()] = d.get(j.strip(), 0) + 1
    return d

# calling dictionary
dictionary = createDict(df['InvestorsName'])

# sorting the Dictionary in descending order according to the number of investments
sorted_d = sorted(dictionary.items(), key=lambda kv: kv[1],reverse = True)

investor_count = []
investor_name = []

# adding top 5 investors to the lists.
for i in range(5):
    print(sorted_d[i][0],sorted_d[i][1])
    investor_count.append(sorted_d[i][1])
    investor_name.append(sorted_d[i][0])
```

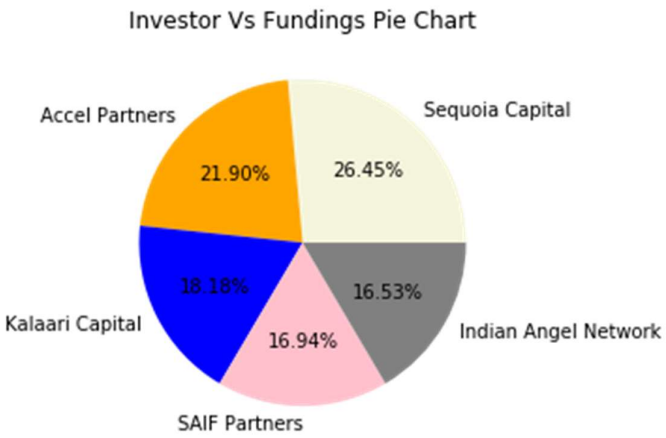


```
y = investor_count
x = investor_name

# plotting the bar graph for top 5 investors
plt.bar(x,y, color = 'orange')
plt.xlabel("Investor Name")
plt.ylabel("No. Of Investments")
plt.title("Investor Vs Fundings")
plt.xticks(rotation = 60)
plt.show()

# plotting the pie chart b/w top 5 investors and number of fundings.
color = ["beige", "orange", "blue", "pink", "grey"]
plt.pie(y, labels = x, colors = color, autopct = "%.2f%%")
plt.title("Investor Vs Fundings Pie Chart")
plt.show()
```

```
Sequoia Capital 64
Accel Partners 53
Kalaari Capital 44
SAIF Partners 41
Indian Angel Network 40
```

GRAPHS –





CODE EXPLANATION –

To find the Investors (Top 5) who have invested maximum number of times , I have included the libraries required that are numpy , pandas , matplotlib.pyplot then –

1) Read the file using pandas read_csv function
2) Cleaned the data by dropping rows with the NaN values.
3) Then , I have made a dictionary , which is consisting Investor Name as key and number of funding's as values. This dictionary is also checking for those startups, which are present at multiple number of Investors.
4) This is done by using strip() function.
5) Now, I have sorted the dictionary according to the number of fundings using SORTED() function in descending order.
6) Took top 5 of the Investors who invested in max number of start-ups.
7) Printed and plotted graph only for TOP 5 of the investors.

RESULT AND JUSTIFICATION –

As the result of the code , we get the TOP 5 Investors who had invested maximum number of times in a start – up.

So, I would suggest my Friend to contact these 5 INVESTORS who can probably invest in his start-up and thereby increasing the chances of getting an initial investment.

QUES. 3)

After re-analysing the dataset you found out that some investors have invested in the same startup at different number of funding rounds. So before finalising the previous list, you want to improvise it by finding the top 5 investors who have invested in different number of startups. This list will be more helpful than your previous list in finding the investment for your friend startup. Find the top 5 investors who have invested maximum number of times in different companies. That means, if one investor has invested multiple times in one startup, count one for that company. There are many errors in startup names. Ignore correcting all, just handle the important ones - Ola, Flipkart, Oyo and Paytm.

CODE SNIPPETS –

In [10]:

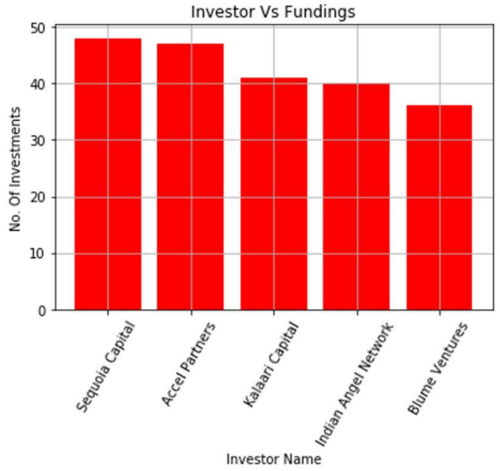
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

file = pd.read_csv("Datasets/startup_funding.csv")
df = file.copy()

# dropping all the missing elements.
df.dropna(subset = ['InvestorsName'],inplace = True)

# Handling the errors in the names of some important start-ups
df['StartupName'].replace('Flipkart.com','Flipkart', inplace = True)
df['StartupName'].replace('Ola Cabs','Ola', inplace = True)
df['StartupName'].replace('Olacabs','Ola', inplace = True)
df['StartupName'].replace('Oyo Rooms','Oyo', inplace = True)
df['StartupName'].replace('Paytm Marketplace','Paytm', inplace = True)
df['StartupName'].replace('Oyorooms','Oyo', inplace = True)
df['StartupName'].replace('OyoRooms','Oyo', inplace = True)
df['StartupName'].replace('OYO Rooms','Oyo', inplace = True)

df.reset_index(drop = True , inplace = True)

# Making dictionary of all the investors and startup combination
def createDict(df):
    d1 = {}
    for i in range(len(df)):
        a = df['InvestorsName'][i].split(',')
        for j in a:
            j = j.strip()
            d1[(df['StartupName'][i],j)] = d1.get((df['StartupName'][i],j),0) + 1

    return d1

# calling dictionary
dictionary = createDict(df)
```

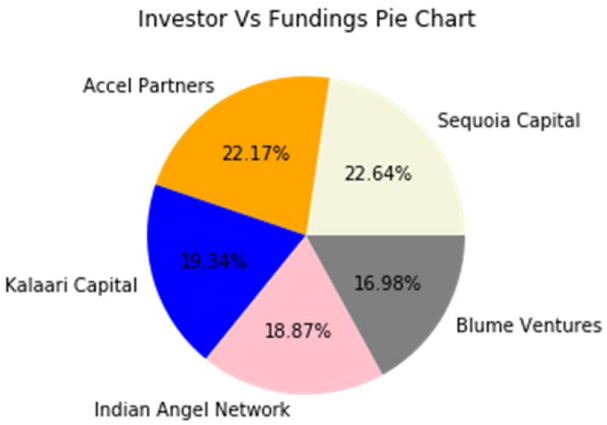
```
# function to create dictionary of non - repetitive investors
def unique_investors(dic):
    x = [] # list of investors
    for i in dictionary:
        x.append(i[1])
    d2 = {} #for finding investors without repetitive invesment done by the investor
    for i in x:
        d2[i] = d2.get(i,0) + 1
    return d2

final_dic = unique_investors(dictionary)
final_dic[''] = 0

# sorting the Dictionary in descending order according to the number of investments
sorted_d = sorted(final_dic.items(), key=lambda kv: kv[1],reverse = True)

investor_count = []
investor_name = []

# adding top 5 investors to the lists.
for i in range(5):
    print(sorted_d[i][0],sorted_d[i][1])
    investor_count.append(sorted_d[i][1])
    investor_name.append(sorted_d[i][0])

y = investor_count
x = investor_name

# plotting the bar graph for top 5 investors
plt.bar(x,y, color = 'red')
plt.xlabel("Investor Name")
plt.ylabel("No. Of Investments")
plt.title("Investor Vs Fundings")
plt.grid()
plt.xticks(rotation = 60)
plt.show()

# plotting the pie chart b/w top 5 investors and number of fundings.
color = ["beige", "orange", "blue", "pink", "grey"]
```

```
plt.ylabel("No. Of Investments")
plt.title("Investor Vs Fundings")
plt.grid()
plt.xticks(rotation = 60)
plt.show()

# plotting the pie chart b/w top 5 investors and number of fundings.
color = ["beige", "orange", "blue", "pink", "grey"]
plt.pie(y, labels = x, colors = color, autopct = "%.2f%%")
plt.title("Investor Vs Fundings Pie Chart")
plt.show()
```

```
Sequoia Capital 48
Accel Partners 47
Kalaari Capital 41
Indian Angel Network 40
Blume Ventures 36
```

GRAPHS –





CODE EXPLANATION –

To find the Investors (Top 5) who have invested maximum number of times in different start-up's , I have included the libraries required that are numpy , pandas , matplotlib.pyplot then –

1) Read the file using pandas read_csv function
2) Cleaned the data by dropping rows with the NaN values and handling corrections in the names of important start-up's like Flipkart , Oyo, Ola and Paytm.
3) Then , I have made a function which creates dictionary , which is consisting Investor Name and Start-up Name as key and number of funding's as values. This dictionary is also checking for those startups, which are present at multiple number of Investors.
4) This is done by using strip() function and then checking for separate investors.
5) Now, another dictionary is created which consists on Investor name as keys and number of funding's as values. These values do- not contain those investor names who have done repetitive investments in the same start-up.
6) Now, I have sorted the dictionary according to the number of fundings using SORTED() function in descending order.
7) Took top 5 of the Investors who invested in max number of start-ups.
8) Printed and plotted graph only for TOP 5 of the investors.

RESULT AND JUSTIFICATION –

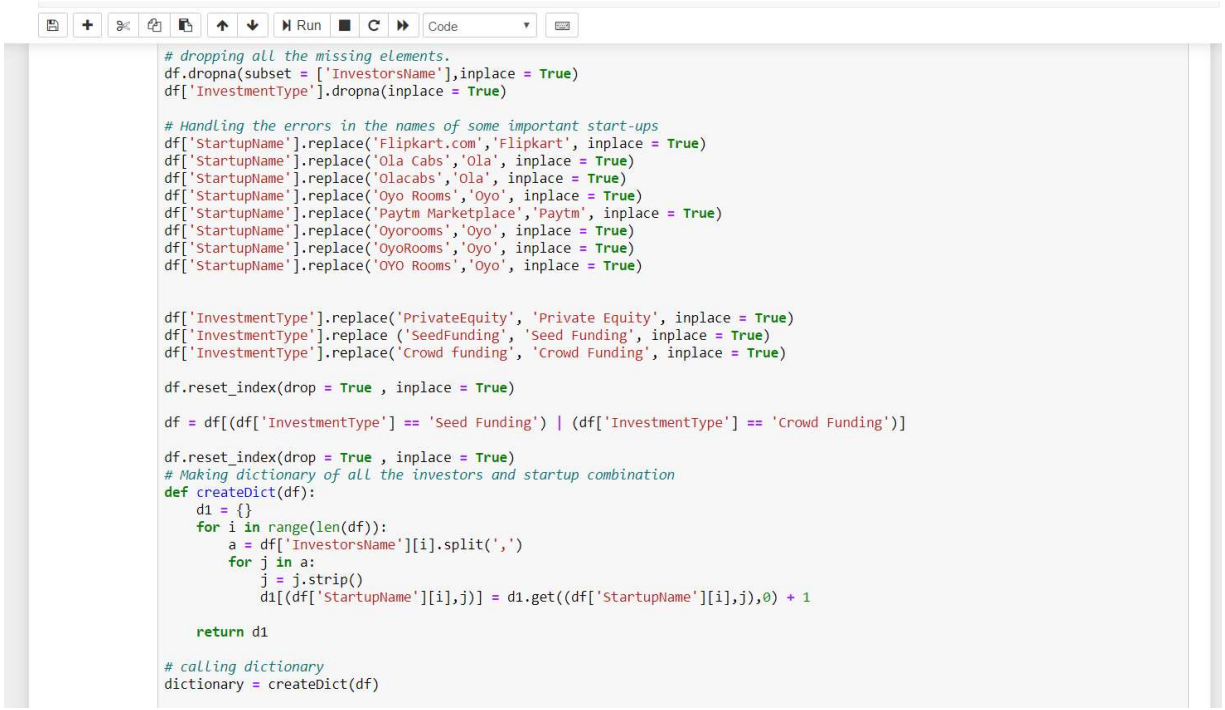
As the result of the code , we get the improved list of TOP 5 Investors who had invested maximum number of times in different start – up's.

So, I would suggest my Friend to contact these 5 INVESTORS who can probably invest in his start-up and thereby increasing the chances of getting an initial investment.

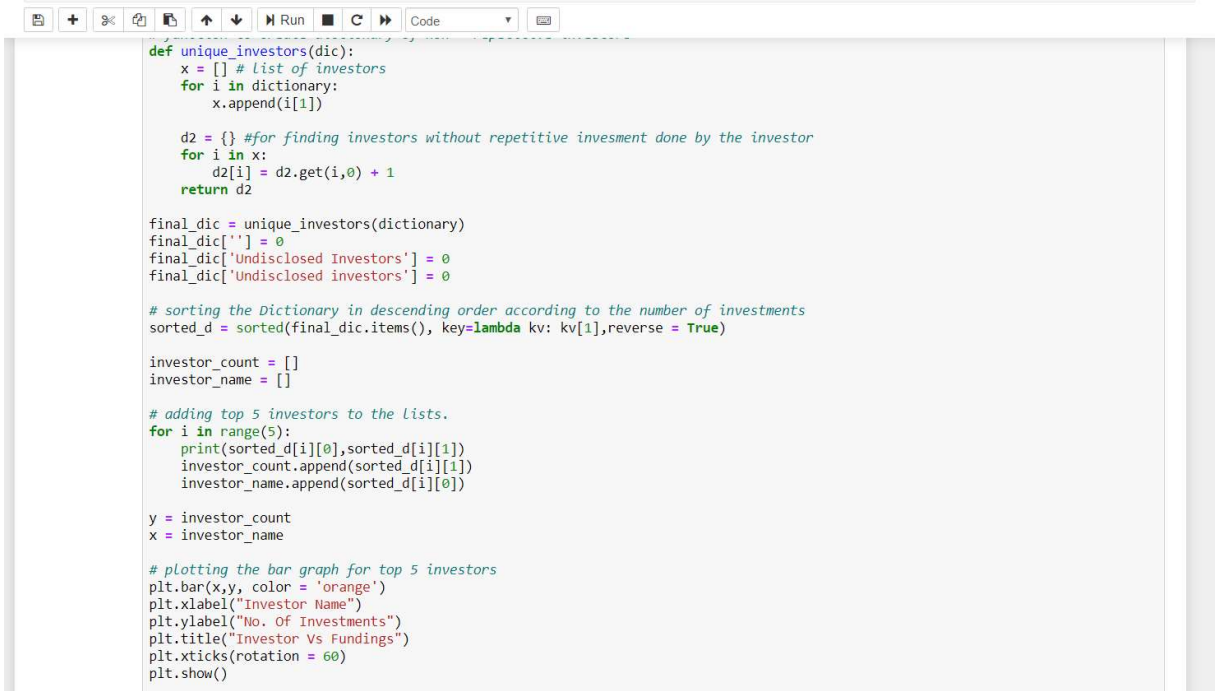
QUES. 4)

Even after putting so much effort in finding the probable investors, it didn't turn out to be helpful for your friend. So you went to your investor friend to understand the situation better and your investor friend explained to you about the different Investment Types and their features. This new information will be helpful in finding the right investor. Since your friend startup is at an early stage startup, the best-suited investment type would be - Seed Funding and Crowdfunding. Find the top 5 investors who have invested in a different number of startups and their investment type is Crowdfunding or Seed Funding. Correct spelling of investment types are - "Private Equity", "Seed Funding", "Debt Funding", and "Crowd Funding". Keep an eye for any spelling mistake. You can find this by printing unique values from this column. There are many errors in startup names. Ignore correcting all, just handle the important ones - Ola, Flipkart, Oyo and Paytm.

CODE SNIPPETS –



```
# dropping all the missing elements.
df.dropna(subset = ['InvestorsName'],inplace = True)
df['InvestmentType'].dropna(inplace = True)

# Handling the errors in the names of some important start-ups
df['StartupName'].replace('Flipkart.com','Flipkart', inplace = True)
df['StartupName'].replace('Ola Cabs','Ola', inplace = True)
df['StartupName'].replace('Olacabs','Ola', inplace = True)
df['StartupName'].replace('Oyo Rooms','Oyo', inplace = True)
df['StartupName'].replace('Paytm Marketplace','Paytm', inplace = True)
df['StartupName'].replace('Oyorooms','Oyo', inplace = True)
df['StartupName'].replace('OyoRooms','Oyo', inplace = True)
df['StartupName'].replace('OYO Rooms','Oyo', inplace = True)


df['InvestmentType'].replace('PrivateEquity', 'Private Equity', inplace = True)
df['InvestmentType'].replace ('SeedFunding', 'Seed Funding', inplace = True)
df['InvestmentType'].replace('Crowd funding', 'Crowd Funding', inplace = True)

df.reset_index(drop = True , inplace = True)

df = df[(df['InvestmentType'] == 'Seed Funding') | (df['InvestmentType'] == 'Crowd Funding')]

df.reset_index(drop = True , inplace = True)
# Making dictionary of all the investors and startup combination
def createDict(df):
    d1 = {}
    for i in range(len(df)):
        a = df['InvestorsName'][i].split(',')
        for j in a:
            j = j.strip()
            d1[(df['StartupName'][i],j)] = d1.get((df['StartupName'][i],j),0) + 1

    return d1

# calling dictionary
dictionary = createDict(df)
```

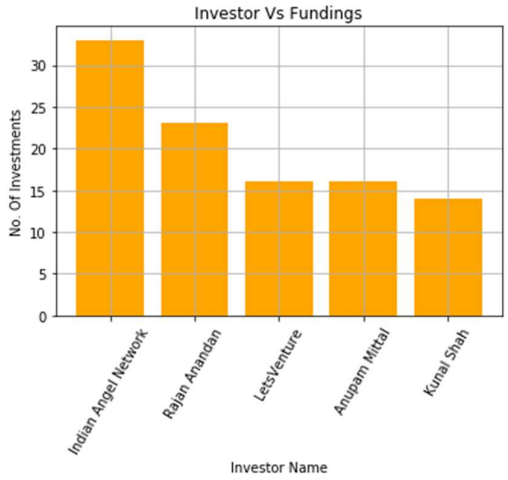


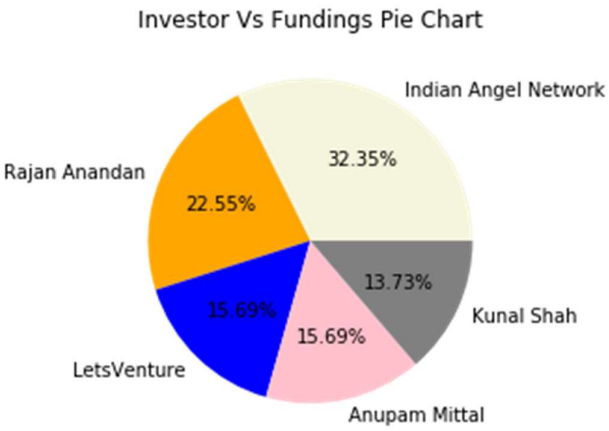
```
def unique_investors(dic):
    x = [] # list of investors
    for i in dictionary:
        x.append(i[1])

    d2 = {} #for finding investors without repetitive invesment done by the investor
    for i in x:
        d2[i] = d2.get(i,0) + 1
    return d2

final_dic = unique_investors(dictionary)
final_dic[''] = 0
final_dic['Undisclosed Investors'] = 0
final_dic['Undisclosed investors'] = 0

# sorting the Dictionary in descending order according to the number of investments
sorted_d = sorted(final_dic.items(), key=lambda kv: kv[1],reverse = True)

investor_count = []
investor_name = []

# adding top 5 investors to the lists.
for i in range(5):
    print(sorted_d[i][0],sorted_d[i][1])
    investor_count.append(sorted_d[i][1])
    investor_name.append(sorted_d[i][0])

y = investor_count
x = investor_name

# plotting the bar graph for top 5 investors
plt.bar(x,y, color = 'orange')
plt.xlabel("Investor Name")
plt.ylabel("No. Of Investments")
plt.title("Investor Vs Fundings")
plt.xticks(rotation = 60)
plt.show()
```

```
# plotting the bar graph for top 5 investors
plt.bar(x,y, color = 'orange')
plt.xlabel("Investor Name")
plt.ylabel("No. Of Investments")
plt.title("Investor Vs Fundings")
plt.xticks(rotation = 60)
plt.show()

# plotting the pie chart b/w top 5 investors and number of fundings.
color = ["beige", "orange", "blue", "pink", "grey"]
plt.pie(y, labels = x, colors = color, autopct = "%.2f%%")
plt.title("Investor Vs Fundings Pie Chart")
plt.show()
```

```
Indian Angel Network 33
Rajan Anandan 23
LetsVenture 16
Anupam Mittal 16
Kunal Shah 14
```

GRAPHS –





CODE EXPLANATION –

To find the Investors (Top 5) who have invested maximum number of times in different start-up's for Crowd and Seed Funding , I have included the libraries required that are numpy , pandas , matplotlib.pyplot then –

1) Read the file using pandas read_csv function
2) Cleaned the data by dropping rows with the NaN values and handling corrections in the names of important start-up's like Flipkart , Oyo, Ola and Paytm and also correcting the names of Investment Type.
3) Reset the index of the DataFrame.
4) Set the DataFrame only for those rows which has Investment type as crowd or Seed Funding.
5) Then , I have made a function which creates dictionary , which is consisting Investor Name and Start-up Name as key and number of funding's as values. This dictionary is also checking for those startups, which are present at multiple number of Investors.
6) This is done by using strip() function and then checking for separate investors.
7) Now, another dictionary is created which consists on Investor name as keys and number of funding's as values. These values do- not contain those investor names who have done repetitive investments in the same start-up.
8) Made investments my by undisclosed investors as 0.
9) Now, I have sorted the dictionary according to the number of fundings using SORTED() function in descending order.
10) Took top 5 of the Investors who invested in max number of start-ups.
11) Printed and plotted graph only for TOP 5 of the investors.

RESULT AND JUSTIFICATION –

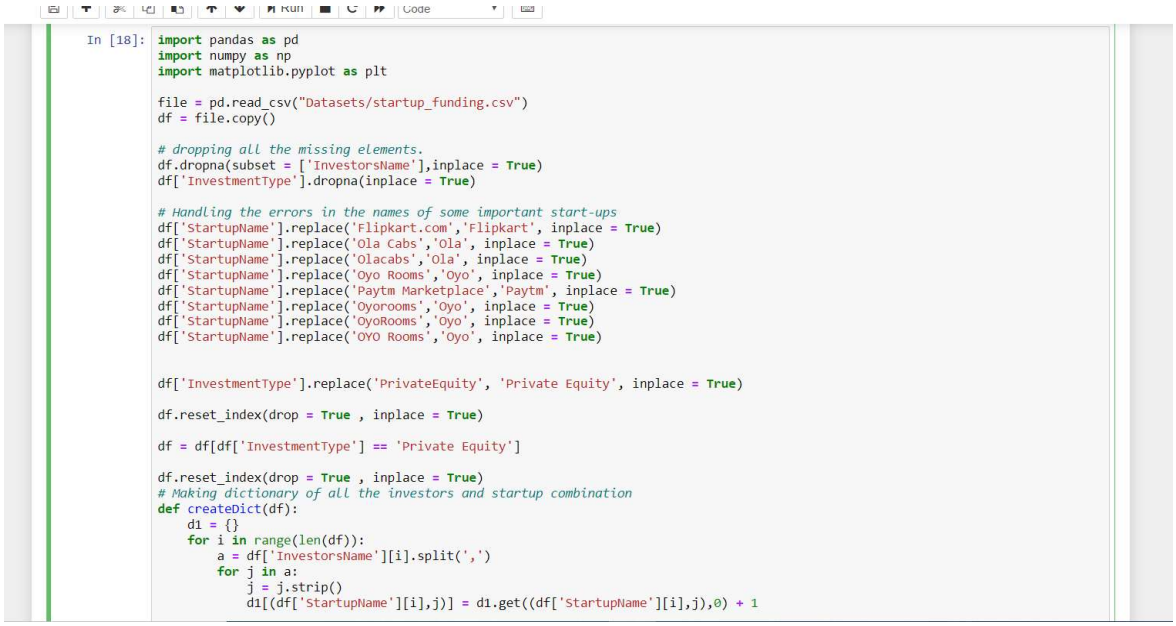
As the result of the code , we get the improved list of TOP 5 Investors who had invested maximum number of times in different start – up's and invested in Seed and Crowd type of Funding's.

So, I would suggest my Friend to contact these 5 INVESTORS who can probably invest in his start-up and thereby increasing the chances of getting an initial investment.

QUES. 5)

Due to your immense help, your friend startup successfully got seed funding and it is on the operational mode. Now your friend wants to expand his startup and he is looking for new investors for his startup. Now you again come as a saviour to help your friend and want to create a list of probable new new investors. Before moving forward you remember your investor friend advice that finding the investors by analysing the investment type. Since your friend startup is not in early phase it is in growth stage so the best-suited investment type is Private Equity. Find the top 5 investors who have invested in a different number of startups and their investment type is Private Equity. Correct spelling of investment types are - "Private Equity", "Seed Funding", "Debt Funding", and "Crowd Funding". Keep an eye for any spelling mistake. You can find this by printing unique values from this column.There are many errors in startup names. Ignore correcting all, just handle the important ones - Ola, Flipkart, Oyo and Paytm.

CODE SNIPPETS –



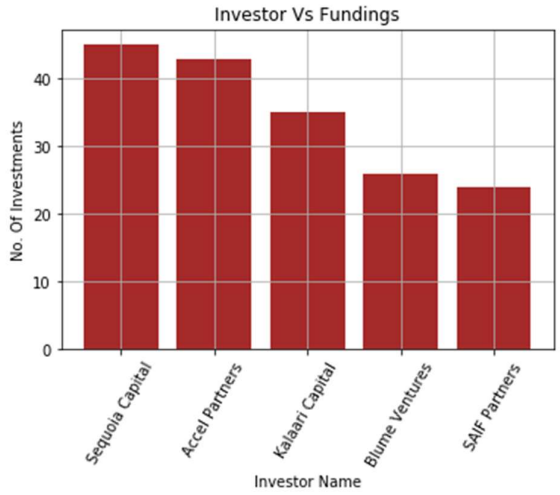
```
In [18]: import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt

         file = pd.read_csv("Datasets/startup_funding.csv")
         df = file.copy()

         # dropping all the missing elements.
         df.dropna(subset = ['InvestorsName'],inplace = True)
         df['InvestmentType'].dropna(inplace = True)

         # Handling the errors in the names of some important start-ups
         df['StartupName'].replace('Flipkart.com','Flipkart', inplace = True)
         df['StartupName'].replace('Ola Cabs','Ola', inplace = True)
         df['StartupName'].replace('Olacabs','Ola', inplace = True)
         df['StartupName'].replace('Oyo Rooms','Oyo', inplace = True)
         df['StartupName'].replace('Paytm Marketplace','Paytm', inplace = True)
         df['StartupName'].replace('Oyorooms','Oyo', inplace = True)
         df['StartupName'].replace('OyoRooms','Oyo', inplace = True)
         df['StartupName'].replace('OYO Rooms','Oyo', inplace = True)


         df['InvestmentType'].replace('PrivateEquity', 'Private Equity', inplace = True)

         df.reset_index(drop = True , inplace = True)

         df = df[df['InvestmentType'] == 'Private Equity']

         df.reset_index(drop = True , inplace = True)
         # Making dictionary of all the investors and startup combination
         def createDict(df):
             d1 = {}
             for i in range(len(df)):
                 a = df['InvestorsName'][i].split(',')
                 for j in a:
                     j = j.strip()
                     d1[(df['StartupName'][i],j)] = d1.get((df['StartupName'][i],j),0) + 1
```

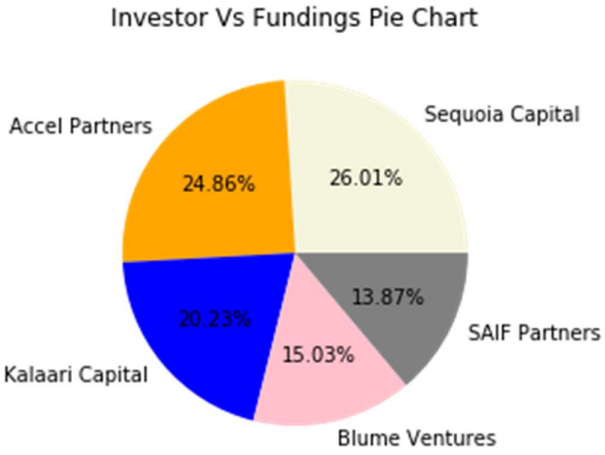
```
             return d1

         # calling dictionary
         dictionary = createDict(df)

         # function to create dictionary of non - repetitive investors
         def unique_investors(dic):
             x = [] # list of investors
             for i in dictionary:
                 x.append(i[1])

             d2 = {} #for finding investors without repetitive invesment done by the investor
             for i in x:
                 d2[i] = d2.get(i,0) + 1
             return d2

         final_dic = unique_investors(dictionary)
         final_dic[''] = 0
         final_dic['Undisclosed Investors'] = 0
         final_dic['Undisclosed investors'] = 0
```

```
# sorting the Dictionary in descending order according to the number of investments
sorted_d = sorted(final_dic.items(), key=lambda kv: kv[1],reverse = True)

investor_count = []
investor_name = []

# adding top 5 investors to the lists.
for i in range(5):
    print(sorted_d[i][0],sorted_d[i][1])
    investor_count.append(sorted_d[i][1])
    investor_name.append(sorted_d[i][0])

y = investor_count
x = investor_name

# plotting the bar graph for top 5 investors
plt.bar(x,y, color = 'brown')
plt.xlabel("Investor Name")
plt.ylabel("No. Of Investments")
plt.title("Investor Vs Fundings")
plt.grid()
plt.xticks(rotation = 60)
plt.show()

# plotting the pie chart b/w top 5 investors and number of fundings.
color = ["beige", "orange", "blue", "pink", "grey"]
plt.pie(y, labels = x, colors = color, autopct = "%.2f%%")
plt.title("Investor Vs Fundings Pie Chart")
plt.show()
```

```
Sequoia Capital 45
Accel Partners 43
Kalaari Capital 35
Blume Ventures 26
SAIF Partners 24
```

GRAPHS –





CODE EXPLANATION –

To find the Investors (Top 5) who have invested maximum number of times in different start-up's for Private Equity , I have included the libraries required that are numpy , pandas , matplotlib.pyplot then –

1) Read the file using pandas read_csv function
2) Cleaned the data by dropping rows with the NaN values and handling corrections in the names of important start-up's like Flipkart , Oyo, Ola and Paytm and also correcting the spelling in names of Investment Type.
3) Reset the index of the DataFrame.
4) Set the DataFrame only for those rows which has Investment type as Private Equity.
5) Then , I have made a function which creates dictionary , which is consisting Investor Name and Start-up Name as key and number of funding's as values. This dictionary is also checking for those startups, which are present at multiple number of Investors.
6) This is done by using strip() function and then checking for separate investors.
7) Now, another dictionary is created which consists on Investor name as keys and number of funding's as values. These values do- not contain those investor names who have done repetitive investments in the same start-up.

8) Made investments my by undisclosed investors as 0.
9) Now, I have sorted the dictionary according to the number of fundings using SORTED() function in descending order.
10) Took top 5 of the Investors who invested in max number of start-ups for Private Equity .
11) Printed and plotted graph only for TOP 5 of the investors.

RESULT AND JUSTIFICATION –

As the result of the code , we get the improved list of TOP 5 Investors who had invested maximum number of times in different start – up's and invested in Private Equity.

So, I would suggest my Friend to contact these 5 INVESTORS who can probably invest in his start-up and thereby increasing the chances of getting investments to expand his start-up and hence get funding in Growth stage.
