Project Case Study : Startup Fundings (Part-2)¶

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import heapq
from operator import itemgetter

startup_funding = pd.read_csv('startup_funding.csv')
```

Functions used to remove discrepancies from StartupName and InvestorType :

- **correctStartupNames** : This function takes pandaDataframe , containing data of startup_funding.csv file, as argument. Then it applies replace function on column StartupName to remove name discrepancies in few important companies (Oyo,Flipkart,Ola,Paytm)

```
def correctStartupNames(sf):

    sf['StartupName'].replace('Oyo Rooms','Oyo',inplace=True)
    sf['StartupName'].replace('Oyorooms','Oyo',inplace=True)
    sf['StartupName'].replace('OYORooms','Oyo',inplace=True)
    sf['StartupName'].replace('PaytMarketplace','Paytm',inplace=True)
    sf['StartupName'].replace('Ola Cabs','Ola',inplace=True)
    sf['StartupName'].replace('Olacabs','Ola',inplace=True)
    sf['StartupName'].replace('OyoRooms','Oyo',inplace=True)
    sf['StartupName'].replace('Flipkart.com','Flipkart',inplace=True)
```

- **correctingInvestmentTypeNames :** This function also takes pandaDataframe , containing data of startup_funding.csv file, as argument. Then it applies replace function on column InvestmentType to remove discrepancies in InvestmentType

```
def correctingInvestmentTypeNames(sf):
```

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

Function to fetch Top 5 Investors who have invested in different startups :

- **top5InvestorsDiffStartups** : By observing the InvestorName column we can see that many entries end with ‘& Others’ and ‘and Others’. So to get rid of this we have replaced these words by empty string (“”) using replace function.

  Cells of InvestorsName column can contain name of multiple investors separated by ‘ , ’ , ‘ and ’ and ‘ & ’ . So we have replaced occurrence of ‘ and ’ and ‘ & ’ with ‘,’ . Then ‘,’ is used to split each cell of InvestorsName Column , each splitted entry i.e, investor name is stored inside dictionary dic as key and its value is of set data-type which contains name of all different startups in which this investor has invested.

  Using min heap we have extracted top 5 investors, who have invested in different startups ,from dic and have stored them in topInvestors[] .

  Then we have returned sorted topInvestors numpy array.

```
def top5InvestorsDiffStartups(sf):

    # Cleaning Investor Names
    sf['InvestorsName'].fillna('Undisclosed Investors',inplace=True)
    sf['InvestorsName'] = sf['InvestorsName'].str.title()
    sf['InvestorsName'] = sf['InvestorsName'].str.replace(' & Others','')
    sf['InvestorsName'] = sf['InvestorsName'].str.replace(' & ',',')
    sf['InvestorsName'] = sf['InvestorsName'].str.replace(' And Others','')
    sf['InvestorsName'] = sf['InvestorsName'].str.replace(' And ',',')

    #removing data of undisclosed Investors from sf
    sf = sf[(sf['InvestorsName']!='Undisclosed Investor')  &
                                      (sf['InvestorsName']!='Undisclosed Investors')]
    sf.reset_index(drop=True,inplace=True)
```

```
# storing investor name and different companies he has invested on , in dic using set .
#    structure of dic  ----> dic{'investorname' : set('companyA','companyB' ...list
#                                                  of companies) , ... other entries}

    dic={}
    row=0

    for investors in sf['InvestorsName']:

        for investor in investors.split(','):

            if(investor!=''):

                investor = investor.strip();
                setList = dic.get(investor,set())
                setList.add(sf['StartupName'][row])
                dic[investor] = setList;

        row+=1;

   # mainitaing min heap to store top 5 investors
   # operation on heap ---> heapq.heappush(topInvestors,
   #                                                       (no. of different investments,investorName))

    topInvestors = []

    for investor in dic.keys():

        if(len(topInvestors)<5):
            heapq.heappush(topInvestors,(len(dic.get(investor)),investor))
        else:
            heapq.heappushpop(topInvestors,(len(dic.get(investor)),investor))


    np_topInvestors= np.array(topInvestors)    #converting to numpy array

    # sorting topInvestors by number of investments in different companies in
    # decreasing order
    np_topInvestors = np_topInvestors[np_topInvestors[:,0].argsort()[::-1]]
    return np_topInvestors
```

Case 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.

Sol Case 1 :-

```
sf = startup_funding.copy()
```

Column CityLocation contains multiple cities separated by '/' in many cells . After making first letter of every word of each cell capital, we have splitted each cell of CityLocation by '/' and saved every splitted value i.e, city name inside cities.
In few entries location New Delhi is saved as Delhi so we have replaced every word Delhi with New Delhi in cities (pandas dataframe).

```
cities = sf['CityLocation'].str.title().str.split('/',expand=True)
cities = cities.iloc[0:,0:].replace('Delhi','New Delhi')
cities.fillna('unknown',inplace=True)
```

Now **<u>cities</u>** contains data in the form :-

```
          0          1
0   Bangalore  unknown
1      Mumbai  unknown
2   New Delhi  unknown
```

```
3        Mumbai  unknown
4     Hyderabad  unknown
...         ...      ...
2367    unknown  unknown
2368    unknown  unknown
2369    unknown  unknown
2370    unknown  unknown
2371    unknown  unknown

[2372     rows x 2 columns]
```

Here column 1 contains first splitted value and column 2 contains 2nd splitted value (if present otherwise ‘unknown’)

Now using below code we have appended col2 to col1 and stored the result inside **data**

```
data = cities[0].append(cities[1])
data.reset_index(drop=True,inplace=True)
```

data contains :-

```
0           Bangalore
1           Mumbai
2           New Delhi
3           Mumbai
4           Hyderabad
         ...
4739     unknown
4740     unknown
4741     unknown
4742     unknown
4743     unknown

Length: 4744,  dtype: object
```

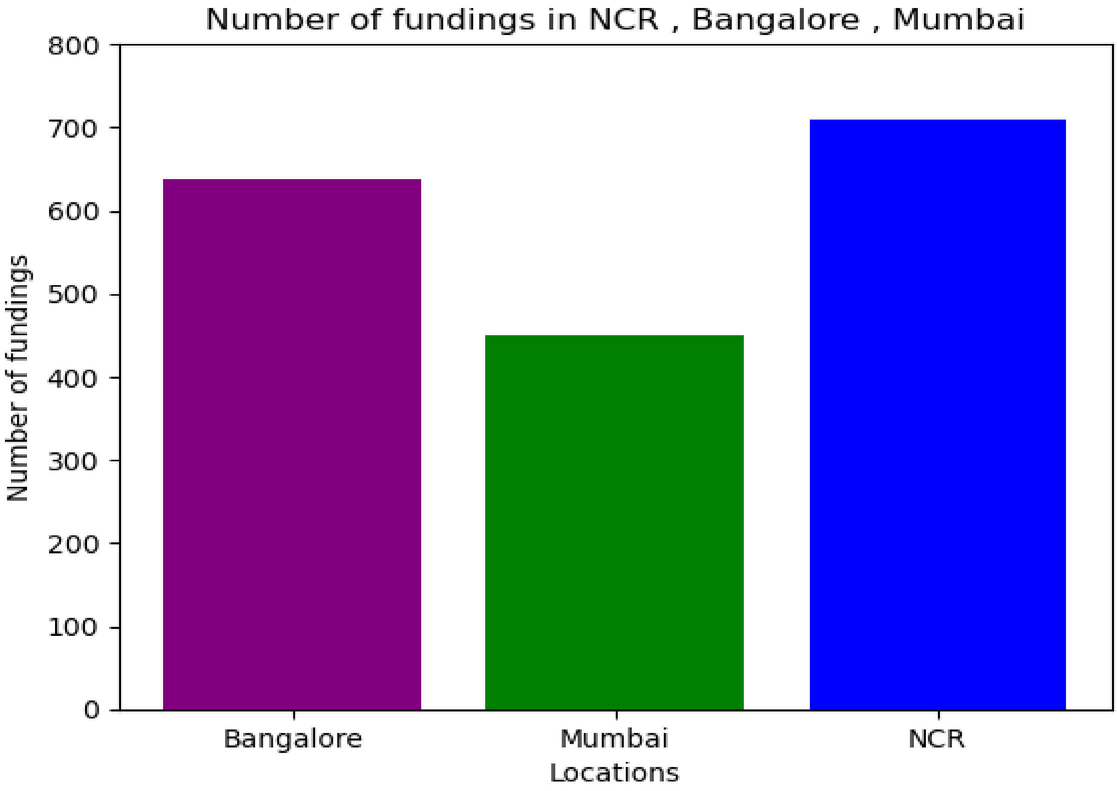
Then Calculating frequency of ‘Bangalore’, ‘Mumbai’ and ‘NCR’ (‘Noida’, ‘Gurgaon’, ,‘New Delhi’) : -

```
freq = data[(data=='Bangalore') | (data=='Mumbai') | (data=='Gurgaon')
                                        | (data=="Noida") |  (data=='New Delhi')].value_counts()
freq.loc['NCR'] = freq['Noida']+freq['New Delhi']+freq['Gurgaon']
freq.drop(freq.index[2:5],inplace=True) # dropping data of Noida , New Delhi and Gurgaon
```

Data inside freq :-

```
Bangalore    637
Mumbai       449
NCR          709
dtype: int64
```

Plotting Bar graph using freq (library used matplotlib.pyplot)



AnsCase 1 :- NCR has received most number of fundings 709 , among the three location

Case 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.

Solution Case 2:-

```
sf = startup_funding.copy()
```

Some values in InvestorsName column of sf is NaN. We have filled these NaN value with ‘Undisclosed Investor’ using below code:-

```
sf['InvestorsName'].fillna('Undisclosed Investor',inplace=True)
```

Filling Investor Names and number of investments made by them in dic

```
for investors in sf['InvestorsName'].str.split(','):
   for investor in investors:
       investor = investor.strip()
       dic[investor] = dic.get(investor,0)+1
```

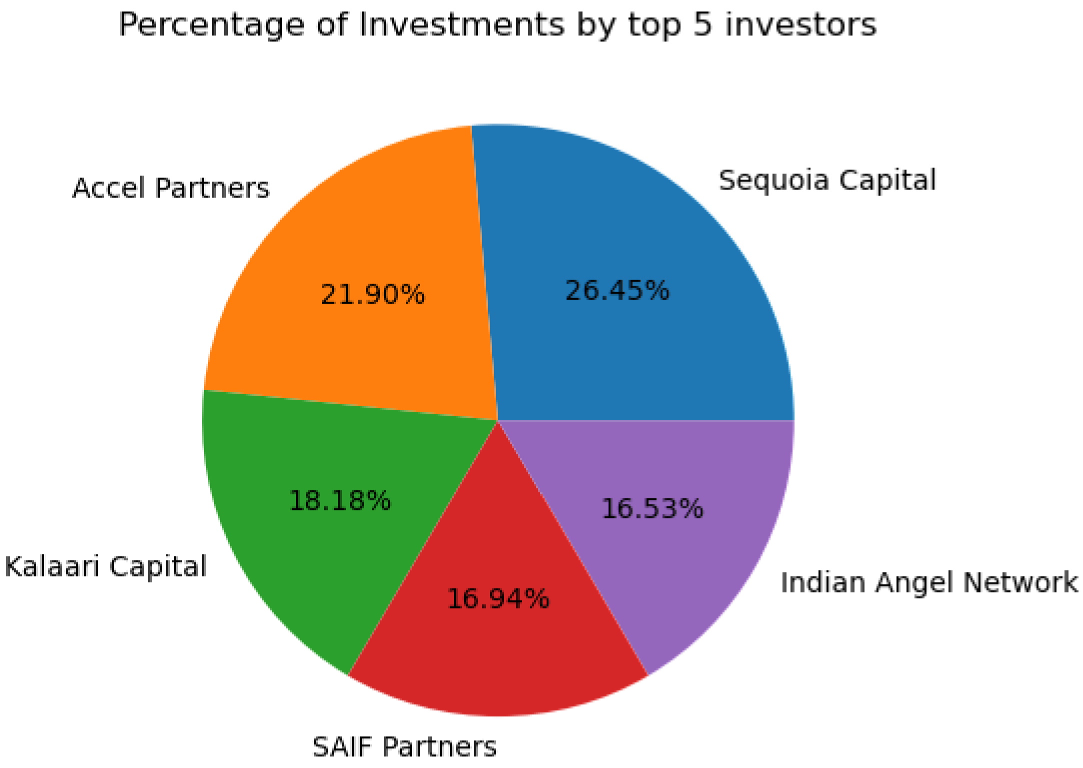
Fetching top 5 Investors who have invested maximum number of times

```
topInvestors = dict(sorted(dic.items(),key=itemgetter(1),reverse=True)[:5])
```

<u>Data inside topInvestors</u> : -

```
{
      'Sequoia Capital': 64,
      'Accel Partners': 53,
     'Kalaari Capital': 44,
     'SAIF Partners': 41,
     'Indian Angel Network': 40
}
```

Plotting pie chart using topInvestors data :-



Ans Case 2 :-

Top 5 Investors are :-

1.	**Sequoia Capital**	**:**	**64 investments**
2.	**Accel Partners**	**:**	**53 investments**
3.	**Kalaari Capital**	**:**	**44 investments**
4.	**SAIF Partners**	**:**	**41 investments**
5.	**Indian Angel Network**	**:**	**40 investments**

Case 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.

Solution Case 3 :-

```
sf = startup_funding.copy()
```

Using function **correctStartupNames** to remove discrepancies in startup names

```
correctStartupNames(sf)
```

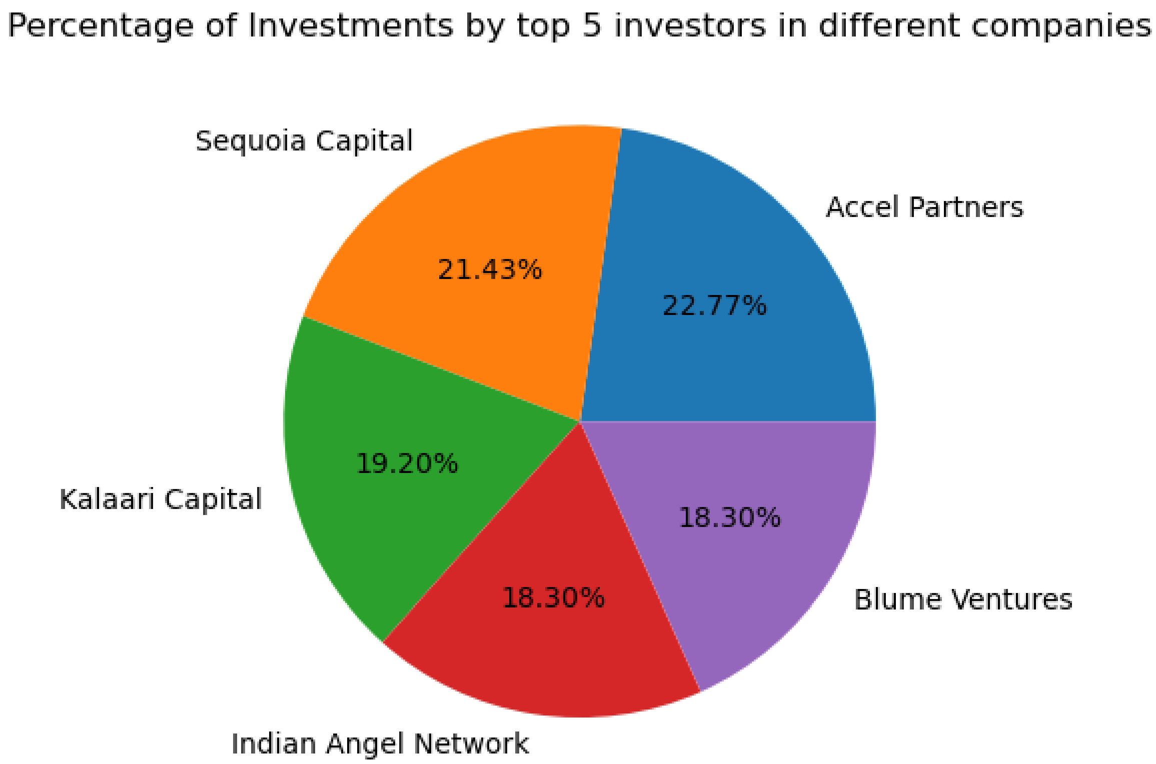
Fetching top 5 Investors who have invested in different startups using **top5InvestorsDiffStartups** function

```
np_topInvestors = top5InvestorsDiffStartups(sf)
```

Data inside np_topInvestors :-

```
[['51' 'Accel Partners']
 ['48' 'Sequoia Capital']
 ['43' 'Kalaari Capital']
 ['41' 'Indian Angel Network']
 ['41' 'Blume Ventures']]
```

Plotting pie chart using np_topInvestors data :-



Ans Case 3 :-

Top 5 investors who have invested in different number of startups are :

1. Accel Partners	:	51 startups
2. Sequoia Capital	:	48 startups
3. Kalaari Capital	:	43 startups
4. Indian Angel Network	:	41 startups
5. Blume Ventures	:	41 startups

Case 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.

Solution Case 4 :-

```
sf = startup_funding.copy()
```

Using function **<u>correctingInvestmentTypeNames</u>** to remove discrepancies in column Investment type

```
correctingInvestmentTypeNames(sf)
```

Filling sf with only those rows which have InvestmentType as ‘Seed Funding’ or ‘Crowd Funding’

```
sf =  sf[(sf['InvestmentType']=='Seed Funding') |
                                        (sf['InvestmentType']=='Crowd  Funding')]
```

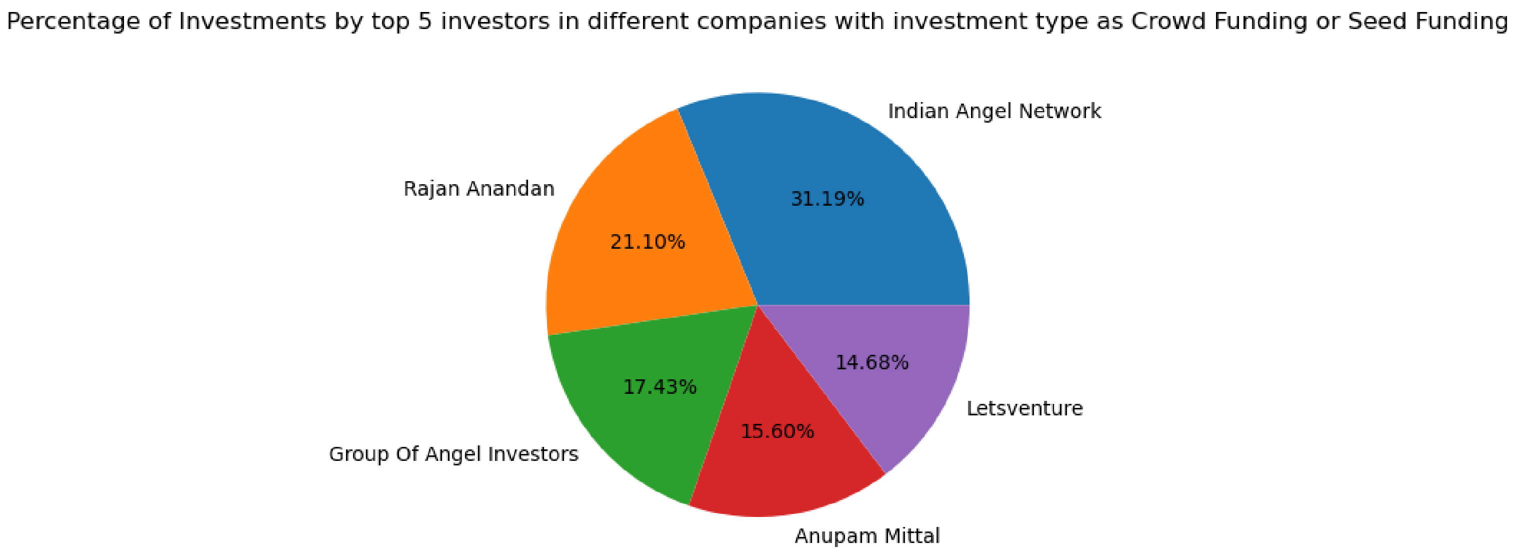
Correcting important StartupNames and fetching top 5 Investors who have invested in different startups

```
correctStartupNames(sf)
np_topInvestors = top5InvestorsDiffStartups(sf)
```

Data inside np_topInvestors :-

```
[['34' 'Indian Angel Network']
 ['23' 'Rajan Anandan']
 ['19' 'Group Of Angel Investors']
 ['17' 'Anupam Mittal']
 ['16' 'Letsventure']]
```

Plotting Pie chart using np_topInvestors data :-



Ans case 4 :-

Top 5 investors with investment type as Crowd Funding or Seed Funding , who have invested in different number of startups are :

1.	Indian Angel Network	:	34 startups
2.	Rajan Anandan	:	23 startups
3.	Group Of Angel Investors	:	19 startups
4.	Anupam Mittal	:	17 startups
5.	Letsventure	:	16 startups

Case 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.

Solution case 5 :-

```
sf = startup_funding.copy()
```

Using function **correctingInvestmentTypeNames** to remove discrepancies in column Investment type

```
correctingInvestmentTypeNames(sf)
```

Filling sf with only those rows which have InvestmentType as ‘Private Equity’

```
sf = sf[(sf['InvestmentType']=='Private Equity')]
```

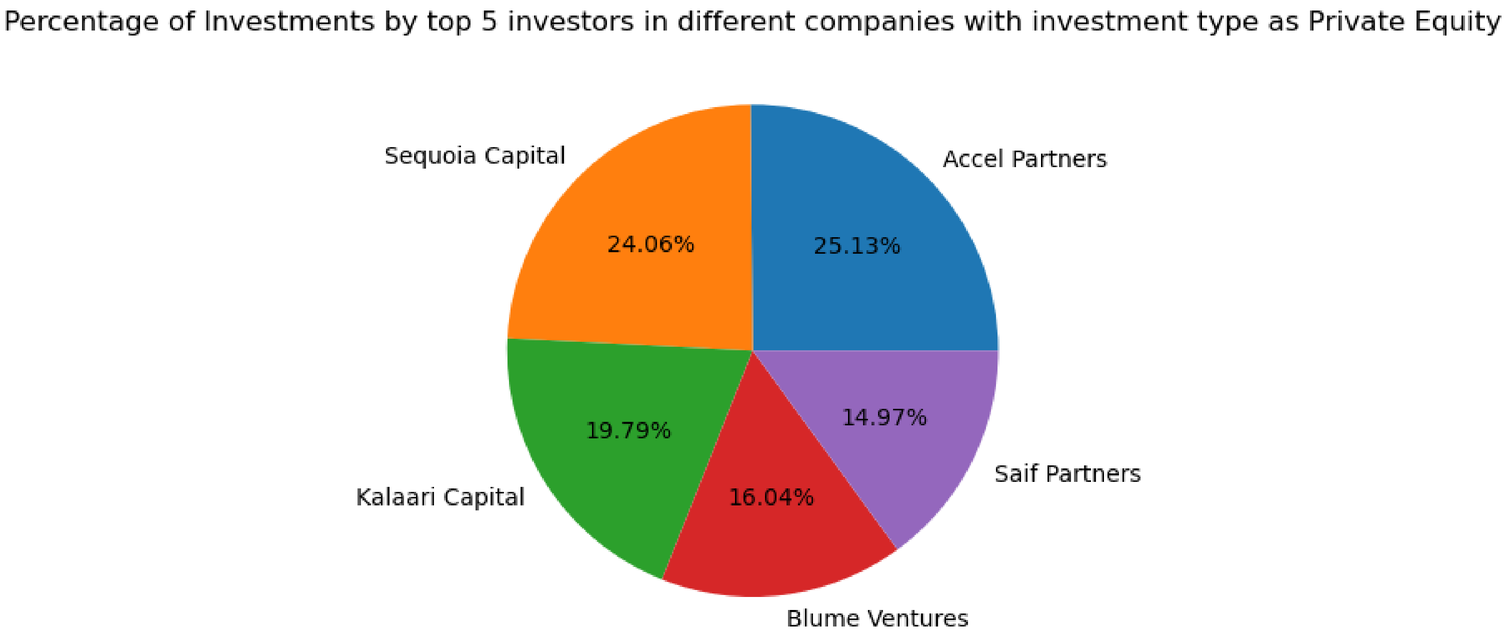
Correcting important StartupNames and fetching top 5 Investors who have invested in different startups

```
correctStartupNames(sf)
np_topInvestors = top5InvestorsDiffStartups(sf)
```

Data inside np_topInvestors :-

```
[['47' 'Accel Partners']
 ['45' 'Sequoia Capital']
 ['37' 'Kalaari Capital']
 ['30' 'Blume Ventures']
 ['28' 'Saif Partners']]
```

Plotting Pie chart using np_topInvestors data :-



Ans case 5 :-

Top 5 investors with investment type as Private Equity , who have invested in different number of startups are :

1.	**Accel Partners**	**:**	**47 startups**
2.	**Sequoia Capital**	**:**	**45 startups**
3.	**Kalaari Capital**	**:**	**37 startups**
4.	**Blume Ventures**	**:**	**30 startups**
5.	**Saif Partners**	**:**	**28 startups**