

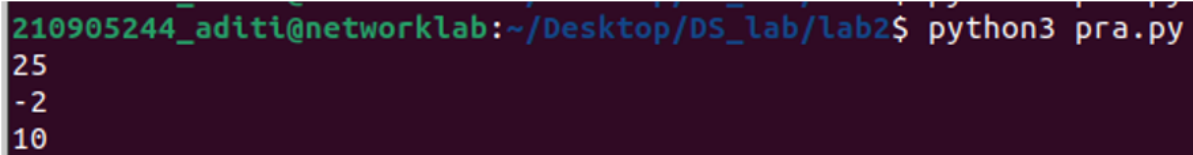
Lab 2
Python Basic Practice-II
Date -
Aditi Shrivastava – 210905244
Roll number – 44

Practice

1) Introduction to Pandas

```
import pandas as pd
import numpy as np
s = pd.Series([3, 9, -2, 10, 5])
print(s.sum())
print(s.min())
print(s.max())
```

OUTPUT:

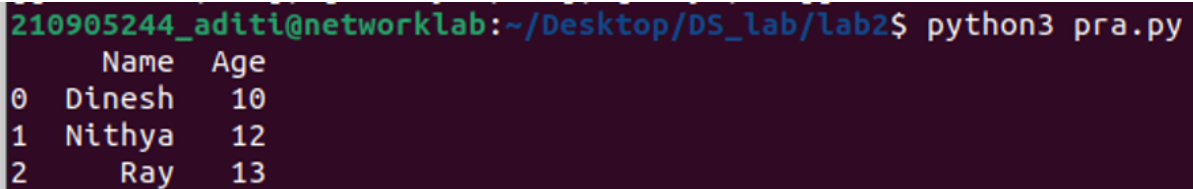


```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
25
-2
10
```

2) Creating a Data Frame

```
import pandas as pd
data=[['Dinesh', 10], ['Nithya', 12], ['Ray', 13]]
df = pd.DataFrame(data, columns = ['Name', 'Age'])
print(df)
```

OUTPUT:



```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
   Name  Age
0  Dinesh   10
1  Nithya   12
2    Ray   13
```

3) Indexed Data Frame

```
import pandas as pd
data = {'Name': ['Kavitha', 'Sudha', 'Raju', 'Vignesh'], 'Age': [28, 34, 29, 42]}
df = pd.DataFrame(data, index= ['rank1', 'rank2', 'rank3', 'rank4'])
print(df)
```

OUTPUT:

```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
      Name  Age
rank1  Kavitha  28
rank2   Sudha  34
rank3   Raju   29
rank4  Vignesh  42
```

4) Creating a DataFrame using Dictionary

```
import pandas as pd
import numpy as np
df1= pd.DataFrame({'A': pd.Timestamp('20130102'), 'B':np.array([3]*4, dtype = 'int32'),
                  'C':pd.Categorical(['Male', 'Female', 'Male', 'Female'])})
df1.shape
df1.head()
df1.tail()
df1.T
```

OUTPUT:

```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
      A  B  C
0 2013-01-02  3  Male
1 2013-01-02  3  Female
2 2013-01-02  3  Male
3 2013-01-02  3  Female
```

5) w

```
Import pandas as pd
Import numpy as np
dates=pd.date_range('20130101', periods=100)
df = pd.DataFrame(np.random.randn(100,4), index=dates, columns=list('ABCD'))
df.head()
df.index
df.tail()
df.columns
df.T
df.sort_index(axis=1, ascending= False)
df.sort_value(by = 'B')
df[0:3]
df['20130105':'20130110']
df.iloc[0]
Df.iloc[0, :2]
df.iloc[0, 0]
df['A']
```

```
df[['A', 'B']]
df[['A', 'B']][:5]
Df.loc['20130101': '20130105', ['A', 'B']][:5]
```

OUTPUT:

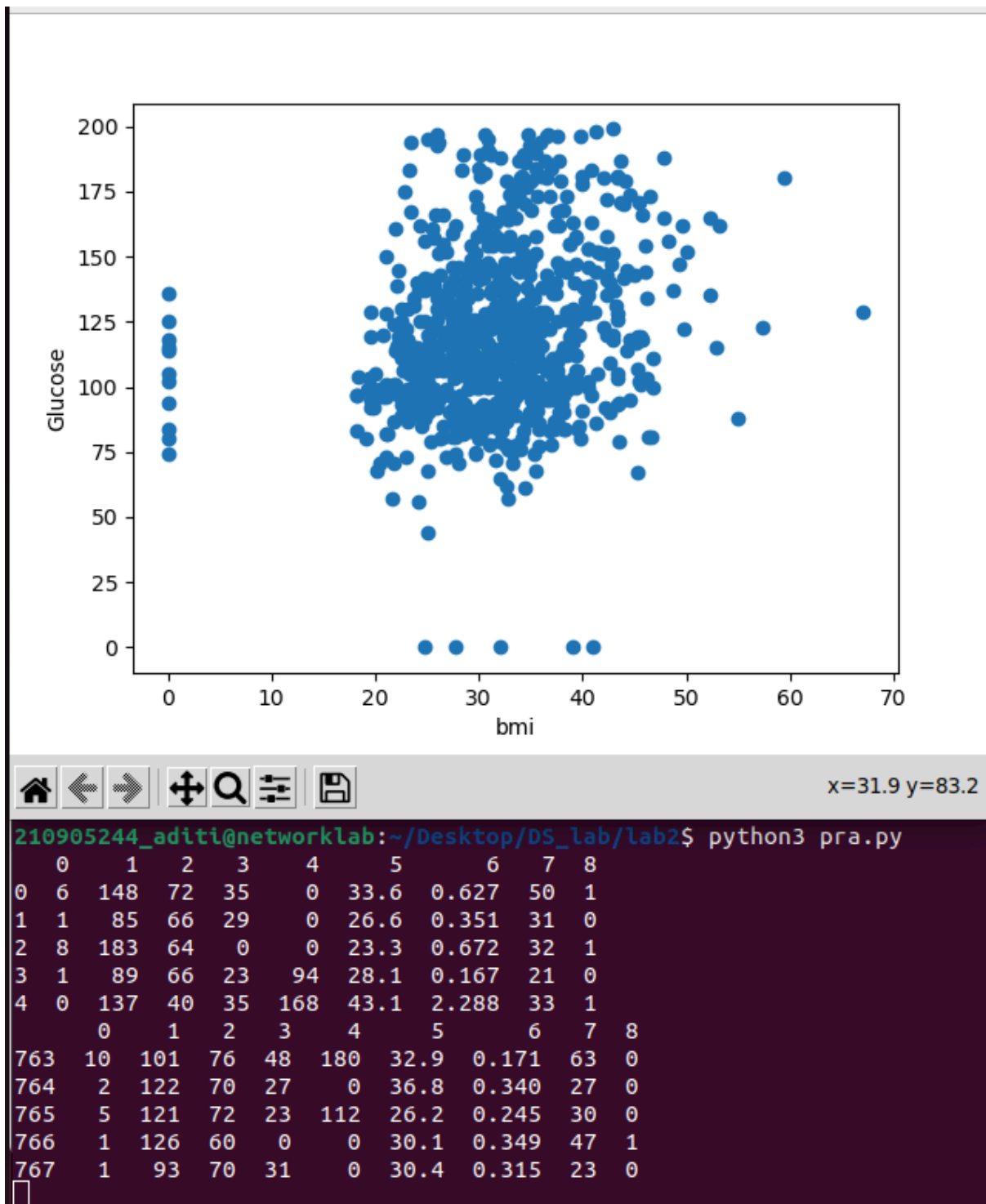
```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
      A      B      C      D
2013-01-01  1.036883 -2.300998 -1.615588  0.364195
2013-01-02  0.400793 -0.588458  1.079088  0.072222
2013-01-03  2.185537  0.329791  0.507921  1.847183
2013-01-04 -0.220727 -0.750867 -0.694508  0.280183
2013-01-05  1.455680 -0.170151  0.764786 -0.434174
...
2013-04-06 -0.549315  0.790297  0.274094  1.503496
2013-04-07  1.029011  0.230499 -0.538879 -1.066146
2013-04-08 -0.282742  0.514951  0.748412 -0.489298
2013-04-09  0.410260 -1.127229 -0.489883 -0.451699
2013-04-10  0.812674 -0.906059  0.586227 -0.232205

[100 rows x 4 columns]
```

Reading CSV file and xls file format

```
df = pd.read_csv('required_files/xyz.csv', header=None)
print(df.head())
print(df.tail())
df.columns=['preg','glu','bp','sft','ins','bmi','dpf','age','class']
plt.scatter(df['bmi'], df['glu'])
plt.xlabel('bmi')
plt.ylabel('Glucose')
plt.show()
```

OUTPUT:



Reading Excel File

```
W=pd.read_excel('required_files/xyz.xls', sheet_name='Sheet1')  
print(W.head())
```

OUTPUT:

```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 pra.py
  Creditability  CreditAmount  DurationOfCreditInMonths
0              1           1049                18
1              1           2799                9
2              1            841               12
3              1           2122               12
4              1           2171               12
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$
```

Reading TXT File

```
H = pd.read_table('required_files/HR.txt')
print(H.head())
f=H['Department'].value_counts()
f.plot(kind='bar')
plt.show()
```

OUTPUT:

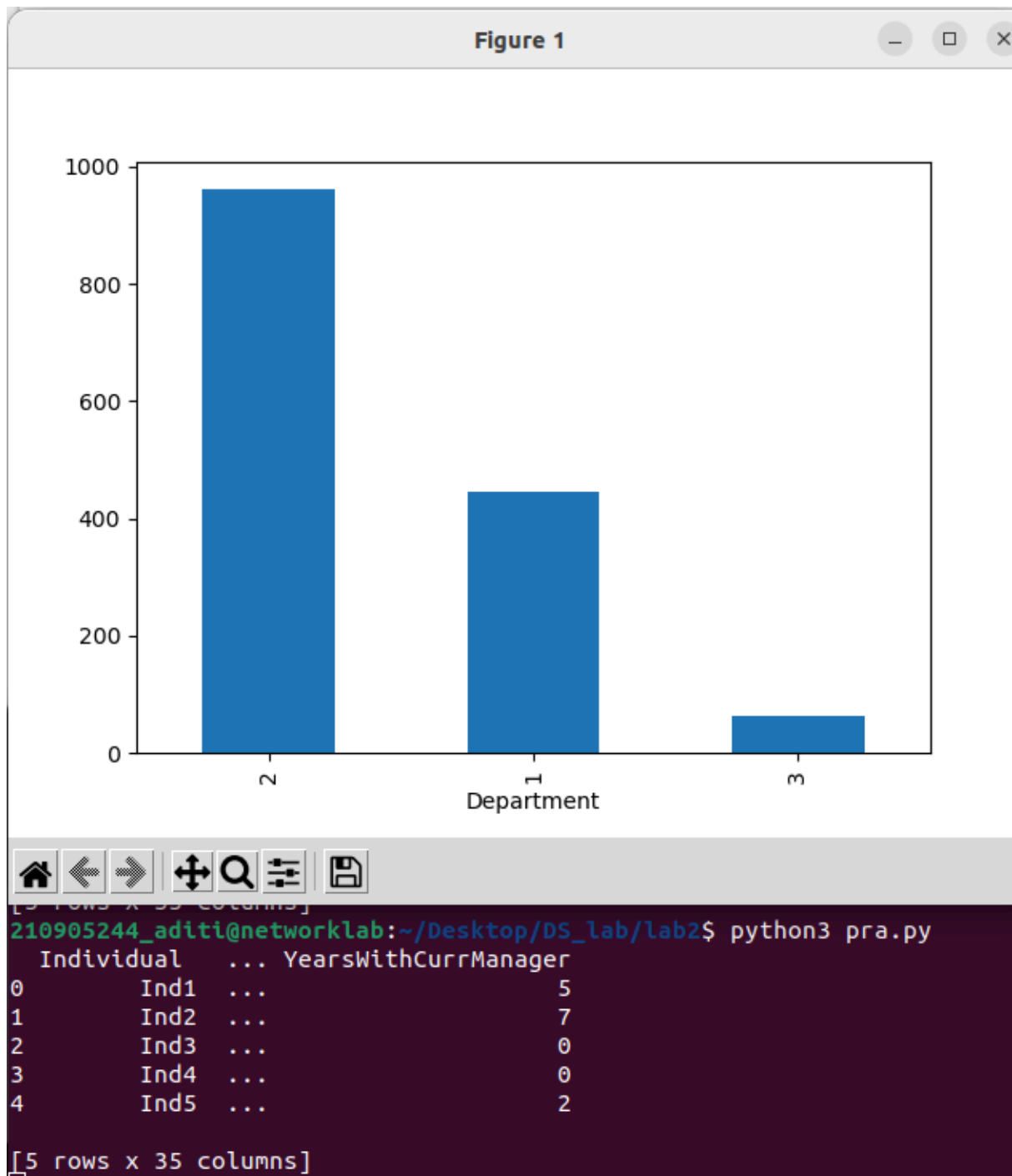
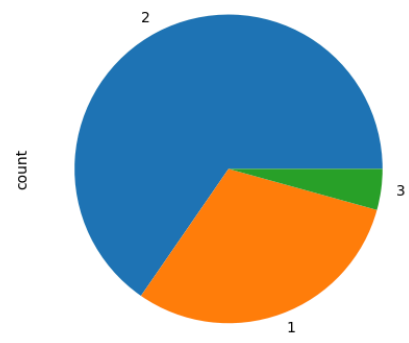


Figure 1

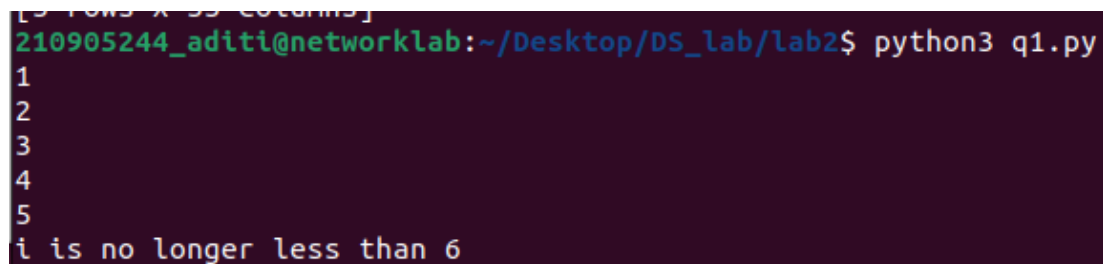


Lab Exercise

11) Write a program to demonstrate while loop with else

```
i = 1
while i<6:
    print(i)
    i+=1
else:
    print("i is no longer less than 6")
```

OUTPUT:

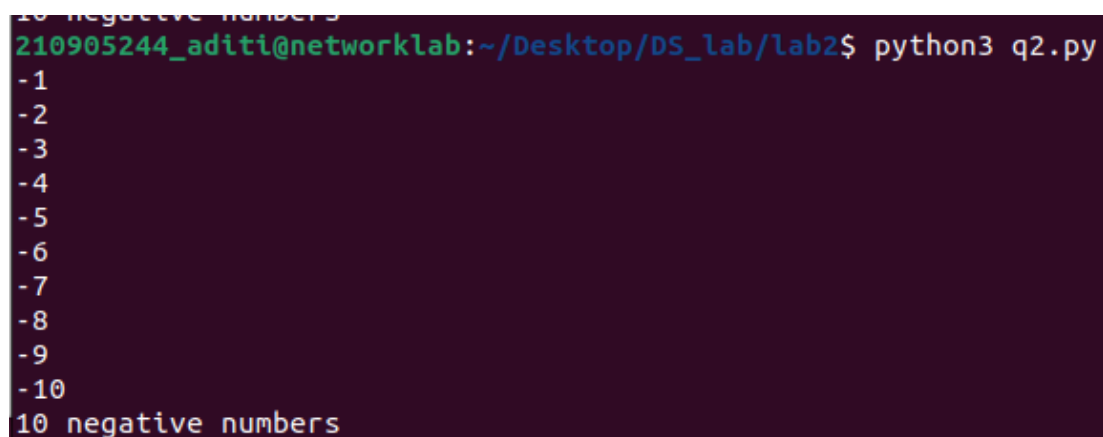


```
[5 rows x 99 columns]
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 q1.py
1
2
3
4
5
i is no longer less than 6
```

12) Write a program to print negative Numbers in a List using while loop.

```
i = 0
while i>=-10:
    i-=1
    print(i)
else:
    print("10 negative numbers")
```

OUTPUT:



```
10 negative numbers
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 q2.py
-1
-2
-3
-4
-5
-6
-7
-8
-9
-10
10 negative numbers
```

13. Define a dictionary containing Students data {Name, Height, Qualification}.

a) Convert the dictionary into DataFrame

b) Declare a list that is to be converted into a new column (Address)

c) Using 'Address' as the column name and equate it to the list and display the result.


```
import pandas as pd
list =['home1', 'home2', 'home3']

df=pd.DataFrame({'Name':pd.Categorical(['rob', 'John', 'sam']),
    'Height':pd.Categorical([160, 170, 180]),
    'Qualification':pd.Categorical(['Btech', 'BA', 'Mtech']),
    'Address':list})
print(df)
```

OUTPUT:

```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 q3.py
   Name Height Qualification Address
0   rob    160         Btech  home1
1  John    170          BA   home2
2   sam    180         Mtech  home3
```

14) Define a dictionary containing Students data {Name, Height, Qualification}.

a) Convert the dictionary into DataFrame

b) Use DataFrame.insert() to add a column and display the result.

```
import pandas as pd
df=pd.DataFrame({'Name':pd.Categorical(['rob', 'John', 'sam']),
    'Height':pd.Categorical([160, 170, 180]),
    'Qualification':pd.Categorical(['10th', '12th', '11th'])})
list =['home1', 'home2', 'home3']
```

```
df.insert(3, 'add', list)
```

```
print(df)
```

OUTPUT:

```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 q4.py
   Name Height Qualification  add
0   rob    160         10th  home1
1  John    170         12th  home2
2   sam    180         11th  home3
```

15)

a) Create two data frames df1 and df2. df1 contains one column 'Name' and df2 contains 4 columns 'Maths', 'Physics', 'Chemistry' and 'Biology' .

b) Concatenate two data frames df1 and df2. Now insert one column 'Total' to the new data frame df_new and find the sum of all marks.

```

import pandas as pd
df1=pd.DataFrame({'Name':pd.Categorical(['Ram', 'Diya', 'Chandan', 'James', 'Alice'])})
print(df1)
df2=pd.DataFrame({'Maths':pd.Categorical([80.0, 90.0, 77.5, 87.5, 86.5]),
                  'Physics':pd.Categorical([81.0, 94.0, 74.5, 83.0, 82.5]),
                  'Chemistry':pd.Categorical([91.5, 86.5, 85.5, 90.0, 82.5]),
                  'Biology':pd.Categorical([82.5, 83.5, 84.5, 85.0, 93.0])})
print(df2)
df_new=pd.concat([df1, df2], axis=1)
df_new['Total']=df2.sum(axis=1)
print(df_new)

```

OUTPUT:

```

210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 q5.py

```

	Name
0	Ram
1	Diya
2	Chandan
3	James
4	Alice

	Maths	Physics	Chemistry	Biology
0	80.0	81.0	91.5	82.5
1	90.0	94.0	86.5	83.5
2	77.5	74.5	85.5	84.5
3	87.5	83.0	90.0	85.0
4	86.5	82.5	82.5	93.0

	Name	Maths	Physics	Chemistry	Biology	Total
0	Ram	80.0	81.0	91.5	82.5	335.0
1	Diya	90.0	94.0	86.5	83.5	354.0
2	Chandan	77.5	74.5	85.5	84.5	322.0
3	James	87.5	83.0	90.0	85.0	345.5
4	Alice	86.5	82.5	82.5	93.0	344.5

16) Create a data frame with column- Name, Quiz_1 /10, In-Sem_1 /15, Quiz_2 /10 and In-Sem_2 /15. Now insert a column Total and find the total and mean as given in the below table.

```

import pandas as pd
list1=[8.0, 9.0, 7.5, 8.5, 6.5]
list2=[11.5, 14.0, 14.5, 13.0, 12.5]
list3=[9.5, 6.5, 8.5, 9.0, 9.0]
list4=[12.5, 13.5, 14.5, 15.0, 13.0]
df=pd.DataFrame({'Name':pd.Categorical(['Annie', 'Diya', 'Charles', 'James', 'Emily']),

```

```
'Quiz_1/10':list1,  
'In-sem_1/15':list2,  
'Quiz_2/10':list3,  
'In-sem_2/15':list4})
```

```
df['Total']=df.iloc[:, 1:].sum(axis=1)  
df_mean= df.select_dtypes(include=['float64', 'int64']).mean()  
df_mean.name='Mean'  
df_new=pd.concat([df, df_mean.to_frame().T])  
print(df_new)
```

OUTPUT:

```
210905244_aditi@networklab:~/Desktop/DS_lab/lab2$ python3 q6.py
```

	Name	Quiz_1/10	In-sem_1/15	Quiz_2/10	In-sem_2/15	Total
0	Annie	8.0	11.5	9.5	12.5	41.5
1	Diya	9.0	14.0	6.5	13.5	43.0
2	Charles	7.5	14.5	8.5	14.5	45.0
3	James	8.5	13.0	9.0	15.0	45.5
4	Emily	6.5	12.5	9.0	13.0	41.0
Mean	NaN	7.9	13.1	8.5	13.7	43.2

