MEERUT PUBLIC GIRLS' SCHOOL

ACADEMIC YEAR: 2021-2022

PRACTICAL FILE

NAME- VITARNA SHARMA

CLASS- XII-C

SUBJECT- INFORMATICS PRACTICES (065)

ROLL NO-

EXAMINER'S SIGNATURE-

CERTIFICATE

THIS IS TO CERTIFY THAT **VITARNA SHARMA** OF **MEERUT PUBLIC GIRLS' SCHOOL** HAS COMPLETED HER PROJECT FILE UNDER MY SUPERVISION. I CERTIFY THAT THIS PROJECT IS UP TO MY EXPECTATIONS ANS AS PER THE GUIDELINES ISSUED BY THE CBSE.

SHIVALI GOEL

(IP TEACHER)

ACKNOWLEDGEMENT

I TAKE THIS OPPORTUNITY TO EXPRESS MY PROFOUND GRATITUDE AND DEEP REGARD TO MY GUIDE **MS. SHIVALI GOEL** FOR HER EXEMPLARY GUIDANCE, MONITORING AND CONSTANT ENCOURAGEMENT THROUGHOUT THE COURSE OF THIS PROJECT. MY THANKS AND APPRECIATION ALSO GO TO THE PEOPLE WHO WILLINGLY HELPED TO ME OUT WITH THE PROJECT

STUDENT'S NAME: **VITARNA SHARMA**

CLASS: **XII-C**

INDEX

1- TO CREATE A SERIES FROM DICTIONARY

2- NAMING A SERIES

3- TO GENERATE A SERIES USING ONE DIMENSIONAL ARRAY

4- TO CREATE A DATAFRAME FROM TWO SERIES

5- TO CREATE A DATAFRAME FROM DICTIONARY OF SERIES

6- CREATING A DATFRAME FROM NUMPY ARRAY

7- TO ADD A NEW COLUMN TO A DATFRAME

8- USING LOC() AND ILOC() METHODS

9- COMPARING TWO SERIES

10- USING HEAD () AND TAIL () METHODS

11- MERGING OF DATAFRAME

12- IMPORTING CSV TO DATAFRAME

13- TO MODIFY THE SALARY OF EMPLOYEE EARNING 16000

14- TO PLOT TWO LINES IN TWO DIFFERENT VIEWS OF THE SAME WINDOW

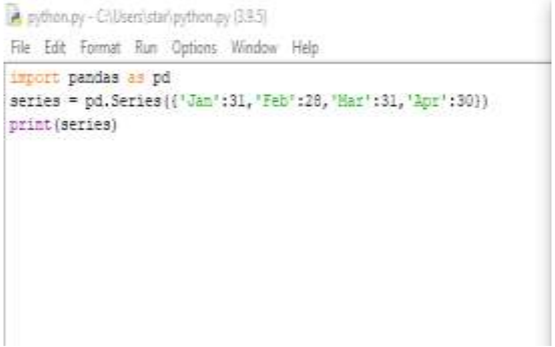
15- PLOTTING LINES WITH DIFFERENT STYLES USING plot () FUNCTION

16- PLOTTING A BAR CHART
17- DISPLAY A HISTOGRAM
18- PLOT ELEMENTS OF TWO LISTS USING BAR CHART
19- BIBLIOGRAPHY

PYTHON PROGRAMS

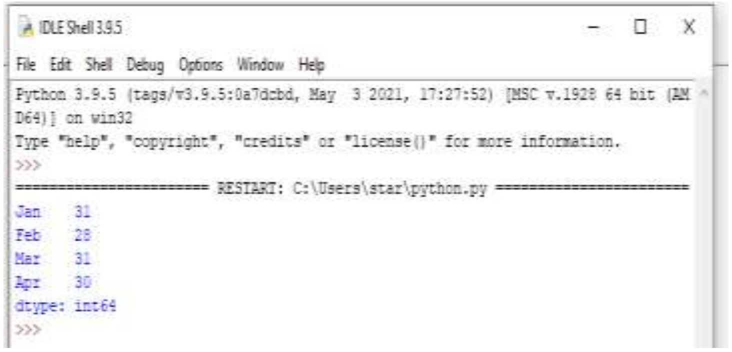
1. TO CREATE A SERIES FROM DICTIONARY

CODE-


python.py - C:\Users\star\python.py (3.9.5)
File Edit Format Run Options Window Help

```
import pandas as pd
series = pd.Series({'Jan':31,'Feb':28,'Mar':31,'Apr':30})
print(series)
```

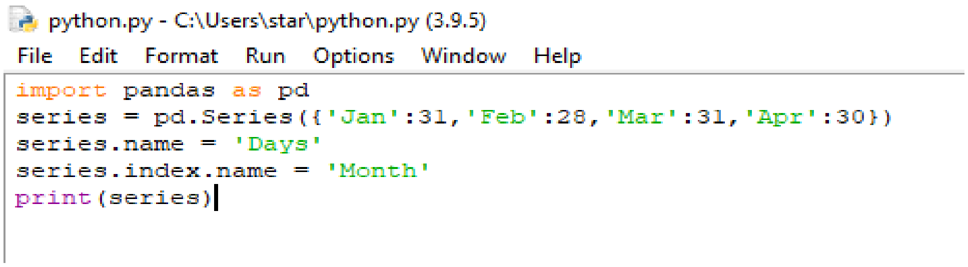
OUTPUT-



```
Python 3.9.5 (tags/v3.9.5:0a7dcbd, May  3 2021, 17:27:52) [MSC v.1928 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
======================= RESTART: C:\Users\star\python.py =======================
Jan    31
Feb    28
Mar    31
Apr    30
dtype: int64
>>>
```

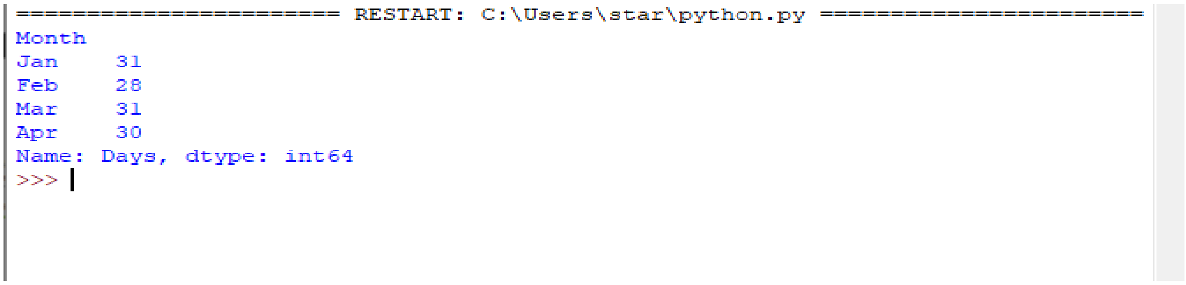
2. NAMING A SERIES

CODE-



```
import pandas as pd
series = pd.Series({'Jan':31,'Feb':28,'Mar':31,'Apr':30})
series.name = 'Days'
series.index.name = 'Month'
print(series)
```

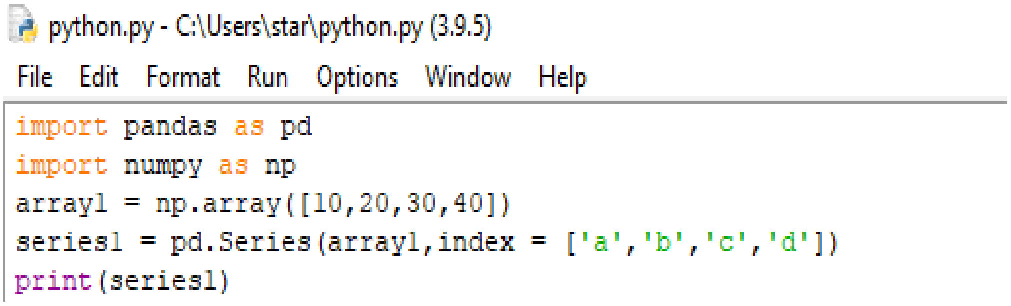
OUTPUT-



```
======================= RESTART: C:\Users\star\python.py =======================
Month
Jan    31
Feb    28
Mar    31
Apr    30
Name: Days, dtype: int64
>>> 
```

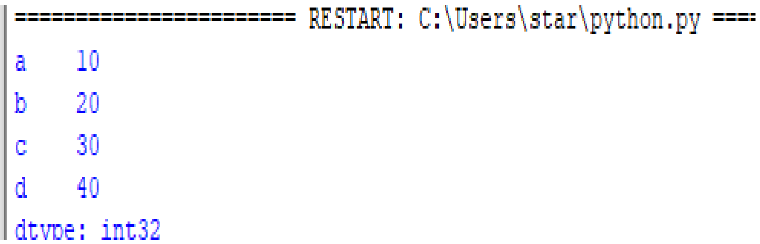
3. *TO GENERATE A SERIES USING ONE DIMENSIONAL ARRAY*

CODE-



```
import pandas as pd
import numpy as np
array1 = np.array([10,20,30,40])
series1 = pd.Series(array1,index = ['a','b','c','d'])
print(series1)
```

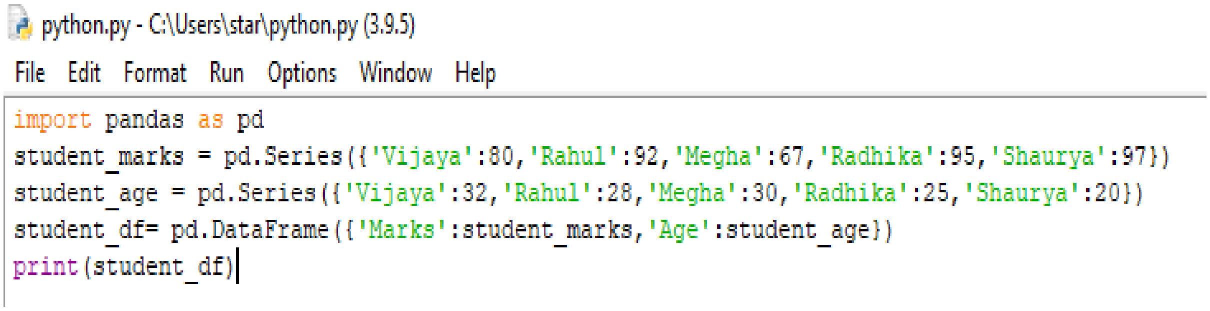
OUTPUT-



```
====================== RESTART: C:\Users\star\python.py ====
a    10
b    20
c    30
d    40
dtype: int32
```

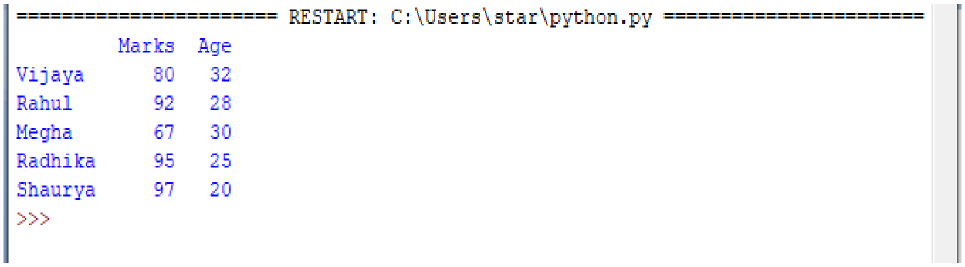
4. TO CREATE A DATAFRAME FROM TWO SERIES

CODE-


python.py - C:\Users\star\python.py (3.9.5)
File Edit Format Run Options Window Help

```
import pandas as pd
student_marks = pd.Series({'Vijaya':80,'Rahul':92,'Megha':67,'Radhika':95,'Shaurya':97})
student_age = pd.Series({'Vijaya':32,'Rahul':28,'Megha':30,'Radhika':25,'Shaurya':20})
student_df= pd.DataFrame({'Marks':student_marks,'Age':student_age})
print(student_df)
```

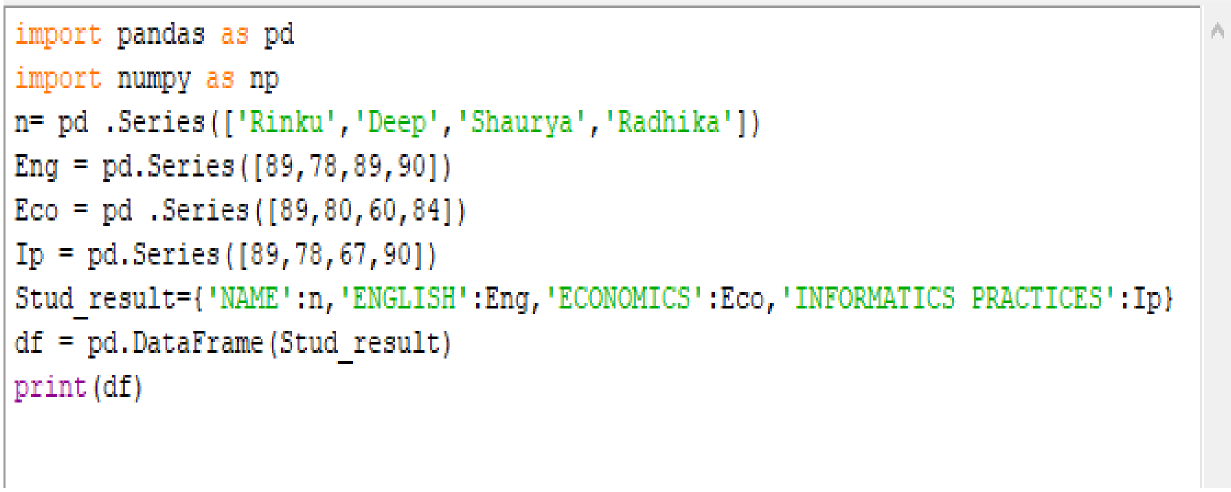
OUTPUT-



```
====================== RESTART: C:\Users\star\python.py ======================
         Marks  Age
Vijaya      80   32
Rahul       92   28
Megha       67   30
Radhika     95   25
Shaurya     97   20
>>>
```

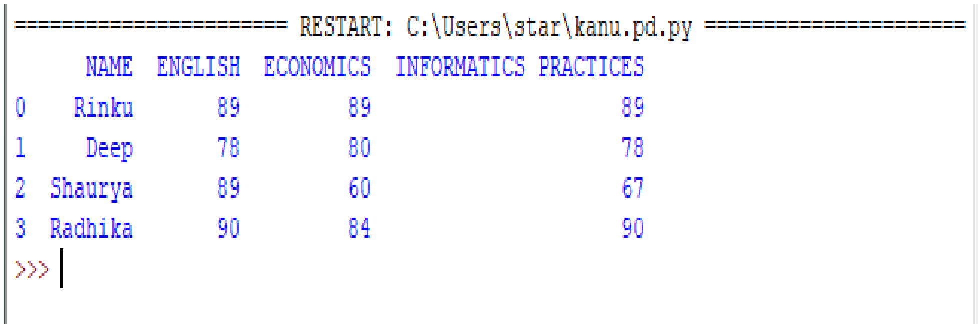
5. TO CREATE A DATAFRAME FROM DICTIONARY OF SERIES

CODE-



```
import pandas as pd
import numpy as np
n= pd .Series(['Rinku','Deep','Shaurya','Radhika'])
Eng = pd.Series([89,78,89,90])
Eco = pd .Series([89,80,60,84])
Ip = pd.Series([89,78,67,90])
Stud_result={'NAME':n,'ENGLISH':Eng,'ECONOMICS':Eco,'INFORMATICS PRACTICES':Ip}
df = pd.DataFrame(Stud_result)
print(df)
```

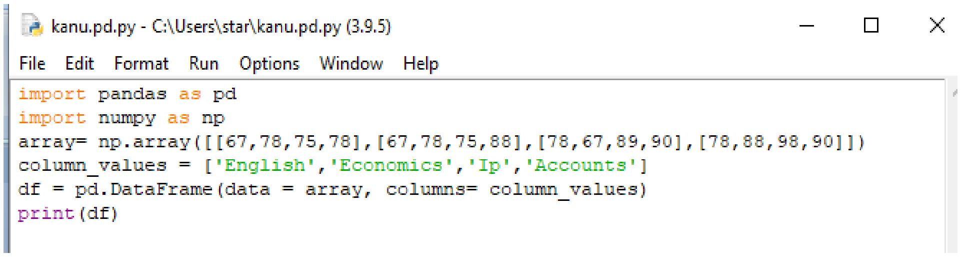
OUTPUT-



```
====================== RESTART: C:\Users\star\kanu.pd.py ======================
      NAME  ENGLISH  ECONOMICS  INFORMATICS PRACTICES
0    Rinku       89         89                     89
1     Deep       78         80                     78
2  Shaurya       89         60                     67
3  Radhika       90         84                     90
>>> 
```

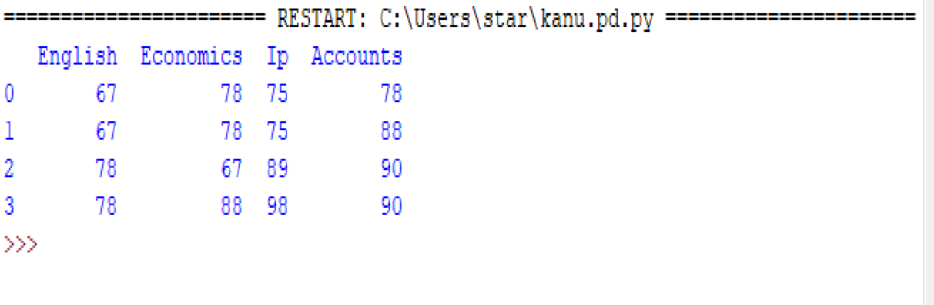
6. CREATING A DATFRAME FROM NUMPY ARRAY

CODE-


kanu.pd.py - C:\Users\star\kanu.pd.py (3.9.5)
File Edit Format Run Options Window Help

```
import pandas as pd
import numpy as np
array= np.array([[67,78,75,78],[67,78,75,88],[78,67,89,90],[78,88,98,90]])
column_values = ['English','Economics','Ip','Accounts']
df = pd.DataFrame(data = array, columns= column_values)
print(df)
```

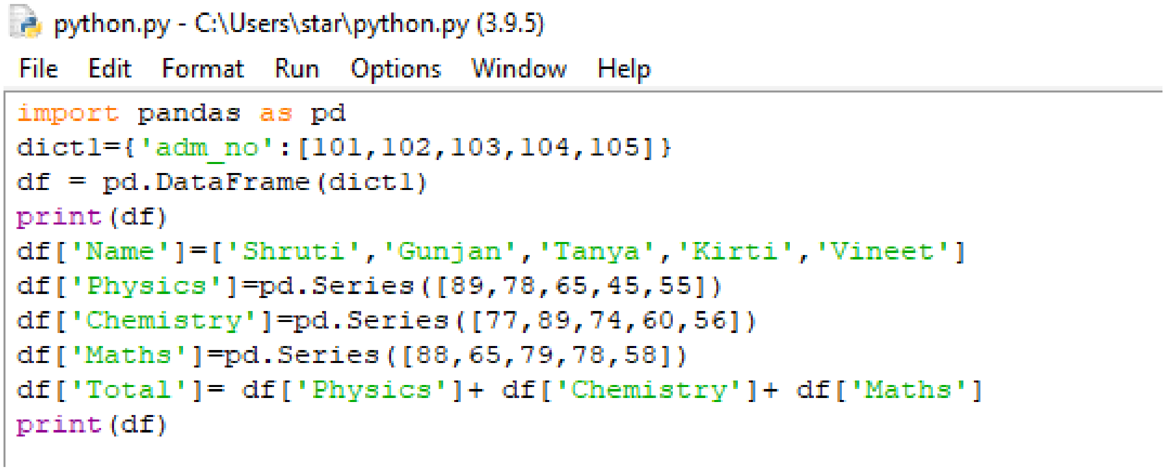
OUTPUT-



```
======================= RESTART: C:\Users\star\kanu.pd.py ======================
   English  Economics  Ip  Accounts
0       67         78  75        78
1       67         78  75        88
2       78         67  89        90
3       78         88  98        90
>>>
```

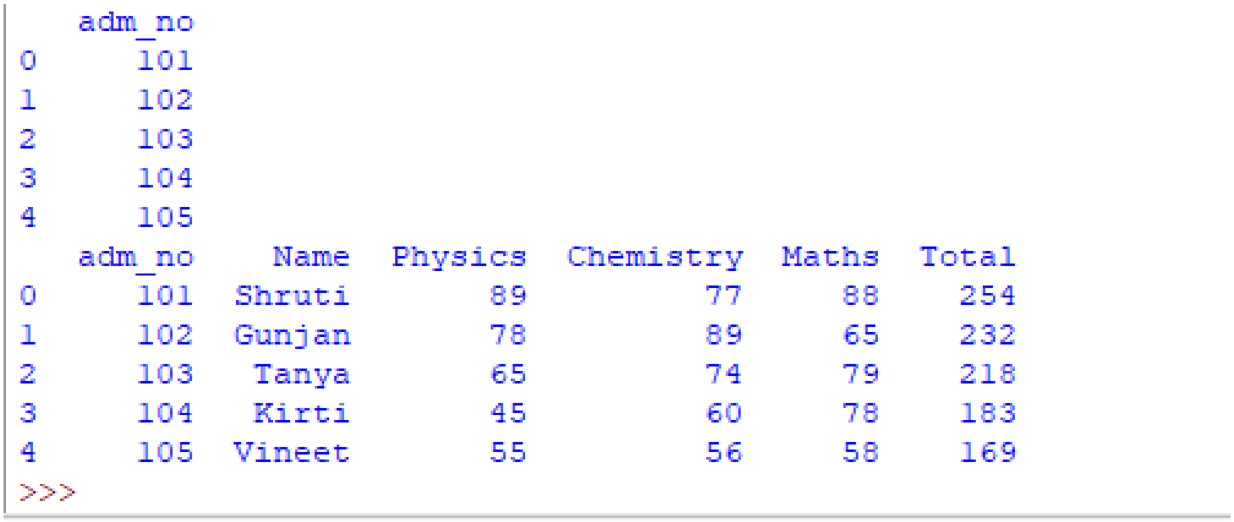
7. TO ADD A NEW COLUMN TO A DATFRAME

CODE-


python.py - C:\Users\star\python.py (3.9.5)
File Edit Format Run Options Window Help

```
import pandas as pd
dict1={'adm_no':[101,102,103,104,105]}
df = pd.DataFrame(dict1)
print(df)
df['Name']=['Shruti','Gunjan','Tanya','Kirti','Vineet']
df['Physics']=pd.Series([89,78,65,45,55])
df['Chemistry']=pd.Series([77,89,74,60,56])
df['Maths']=pd.Series([88,65,79,78,58])
df['Total']= df['Physics']+ df['Chemistry']+ df['Maths']
print(df)
```

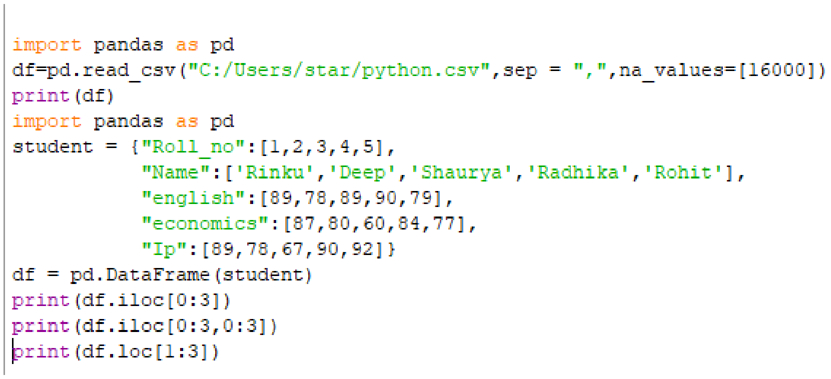
OUTPUT-



```
   adm_no
0     101
1     102
2     103
3     104
4     105
   adm_no    Name  Physics  Chemistry  Maths  Total
0     101  Shruti       89         77     88    254
1     102  Gunjan       78         89     65    232
2     103   Tanya       65         74     79    218
3     104   Kirti       45         60     78    183
4     105  Vineet       55         56     58    169
>>>
```

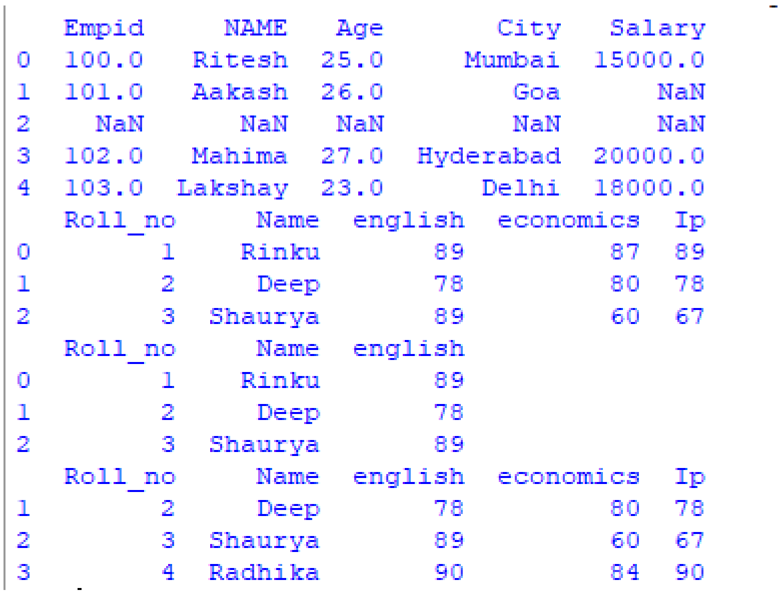
8. USING LOC() AND ILOC() METHODS

CODE-



```
import pandas as pd
df=pd.read_csv("C:/Users/star/python.csv",sep = ",",na_values=[16000])
print(df)
import pandas as pd
student = {"Roll_no":[1,2,3,4,5],
           "Name":['Rinku','Deep','Shaurya','Radhika','Rohit'],
           "english":[89,78,89,90,79],
           "economics":[87,80,60,84,77],
           "Ip":[89,78,67,90,92]}
df = pd.DataFrame(student)
print(df.iloc[0:3])
print(df.iloc[0:3,0:3])
print(df.loc[1:3])
```

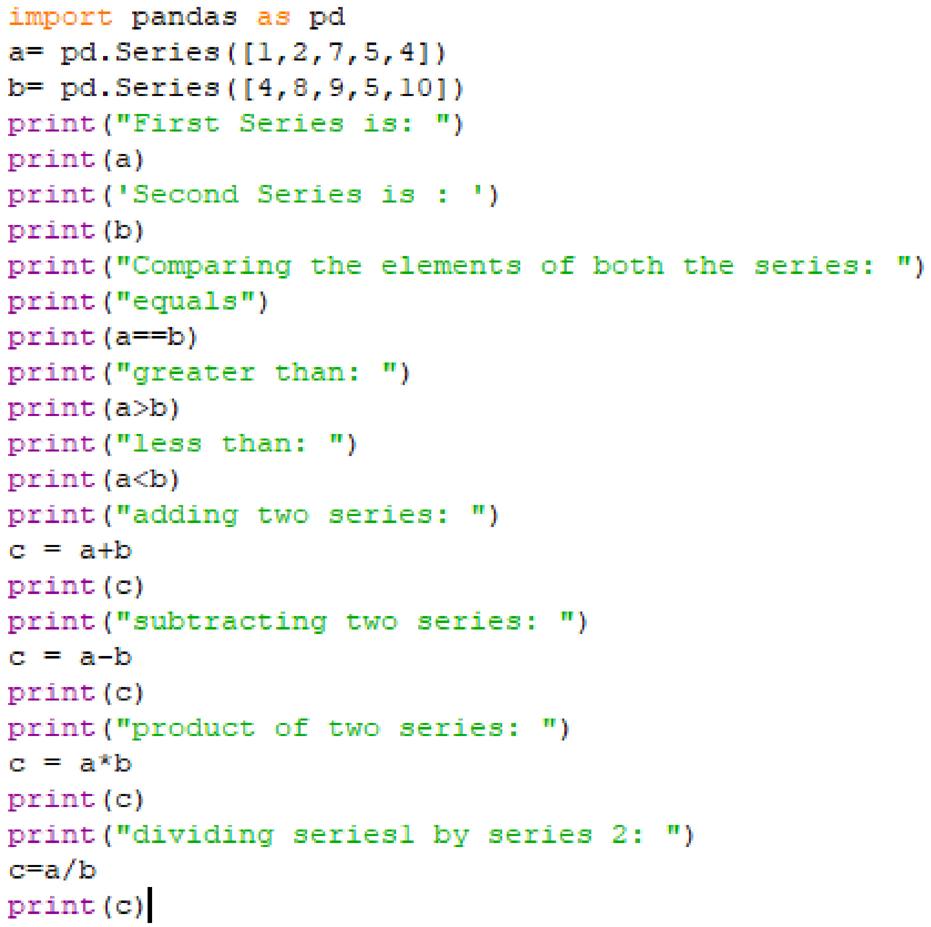
OUTPUT –



```
   Empid     NAME   Age       City   Salary
0  100.0   Ritesh  25.0     Mumbai  15000.0
1  101.0   Aakash  26.0        Goa      NaN
2    NaN      NaN   NaN        NaN      NaN
3  102.0   Mahima  27.0  Hyderabad  20000.0
4  103.0  Lakshay  23.0      Delhi  18000.0
   Roll_no     Name  english  economics  Ip
0        1    Rinku       89         87  89
1        2     Deep       78         80  78
2        3  Shaurya       89         60  67
   Roll_no     Name  english
0        1    Rinku       89
1        2     Deep       78
2        3  Shaurya       89
   Roll_no     Name  english  economics  Ip
1        2     Deep       78         80  78
2        3  Shaurya       89         60  67
3        4  Radhika       90         84  90
```

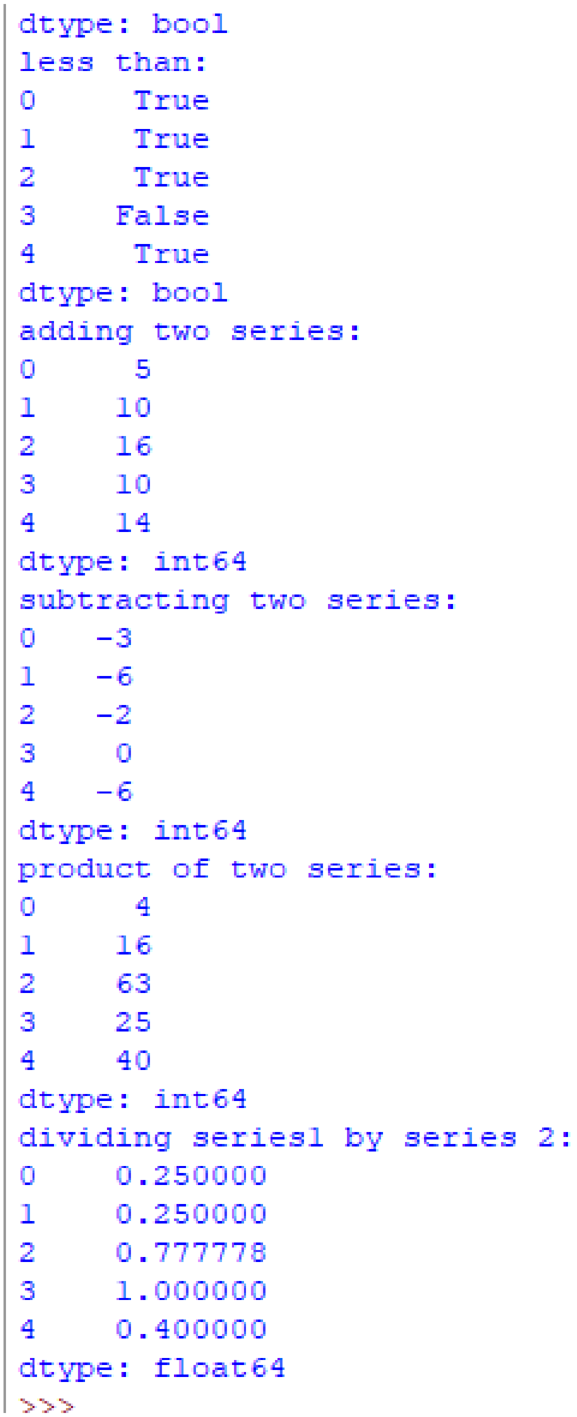
9-COMPARING TWO SERIES

CODE-



```
import pandas as pd
a= pd.Series([1,2,7,5,4])
b= pd.Series([4,8,9,5,10])
print("First Series is: ")
print(a)
print('Second Series is : ')
print(b)
print("Comparing the elements of both the series: ")
print("equals")
print(a==b)
print("greater than: ")
print(a>b)
print("less than: ")
print(a<b)
print("adding two series: ")
c = a+b
print(c)
print("subtracting two series: ")
c = a-b
print(c)
print("product of two series: ")
c = a*b
print(c)
print("dividing series1 by series 2: ")
c=a/b
print(c)
```

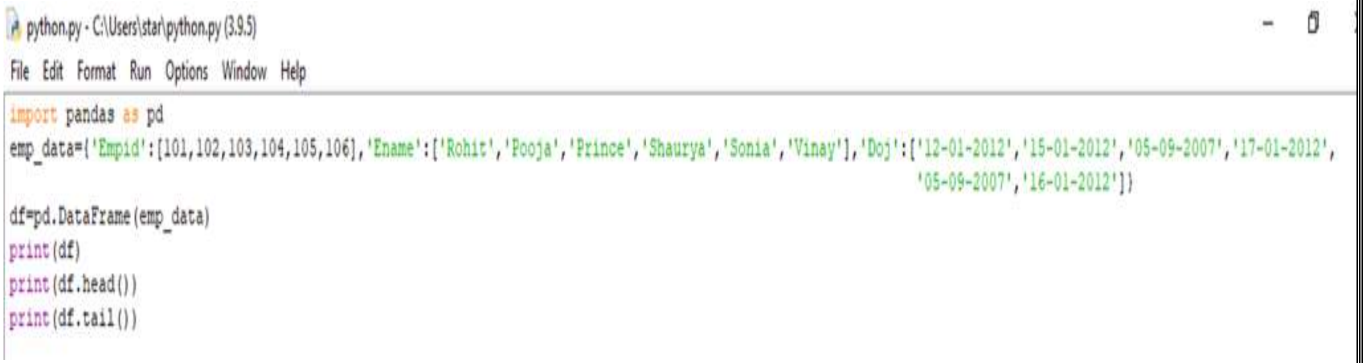
OUTPUT-



```
dtype: bool
less than:
0     True
1     True
2     True
3    False
4     True
dtype: bool
adding two series:
0     5
1    10
2    16
3    10
4    14
dtype: int64
subtracting two series:
0   -3
1   -6
2   -2
3    0
4   -6
dtype: int64
product of two series:
0     4
1    16
2    63
3    25
4    40
dtype: int64
dividing series1 by series 2:
0    0.250000
1    0.250000
2    0.777778
3    1.000000
4    0.400000
dtype: float64
>>>
```

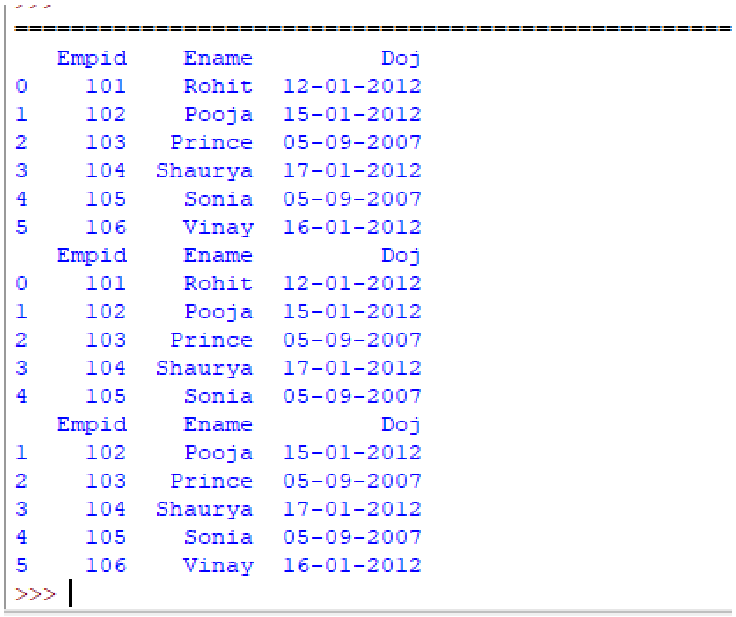
10. USING HEAD () AND TAIL () METHODS

CODE-


python.py - C:\Users\star\python.py (3.9.5)
File Edit Format Run Options Window Help

```
import pandas as pd
emp_data={'Empid':[101,102,103,104,105,106],'Ename':['Rohit','Pooja','Prince','Shaurya','Sonia','Vinay'],'Doj':['12-01-2012','15-01-2012','05-09-2007','17-01-2012',
                                                                                                         '05-09-2007','16-01-2012']}
df=pd.DataFrame(emp_data)
print(df)
print(df.head())
print(df.tail())
```

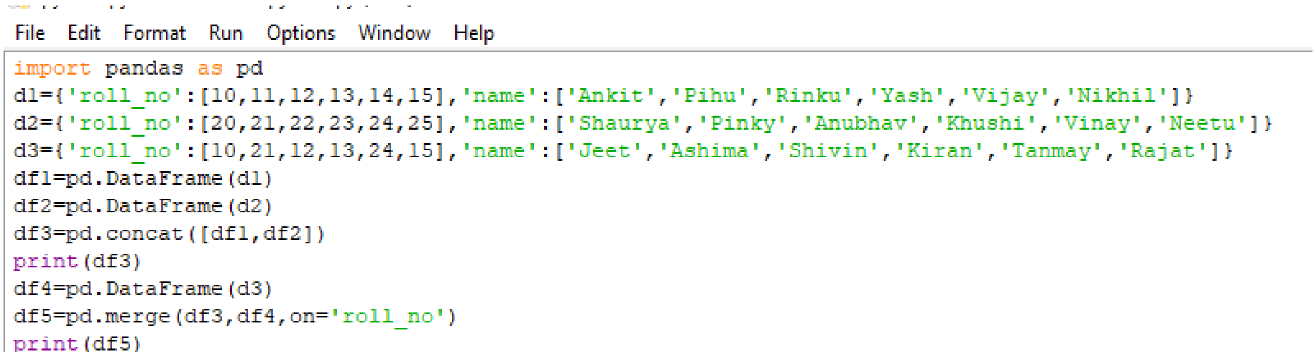
OUTPUT-



```
==================================================
   Empid    Ename         Doj
0    101    Rohit  12-01-2012
1    102    Pooja  15-01-2012
2    103   Prince  05-09-2007
3    104  Shaurya  17-01-2012
4    105    Sonia  05-09-2007
5    106    Vinay  16-01-2012
   Empid    Ename         Doj
0    101    Rohit  12-01-2012
1    102    Pooja  15-01-2012
2    103   Prince  05-09-2007
3    104  Shaurya  17-01-2012
4    105    Sonia  05-09-2007
   Empid    Ename         Doj
1    102    Pooja  15-01-2012
2    103   Prince  05-09-2007
3    104  Shaurya  17-01-2012
4    105    Sonia  05-09-2007
5    106    Vinay  16-01-2012
>>> 
```

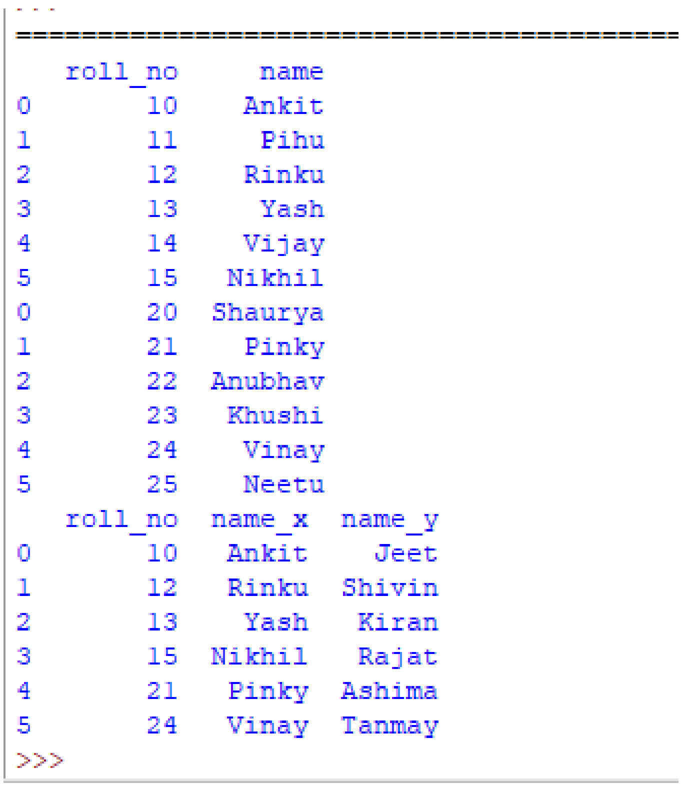
11. MERGING OF DATAFRAME

CODE-



```
import pandas as pd
d1={'roll_no':[10,11,12,13,14,15],'name':['Ankit','Pihu','Rinku','Yash','Vijay','Nikhil']}
d2={'roll_no':[20,21,22,23,24,25],'name':['Shaurya','Pinky','Anubhav','Khushi','Vinay','Neetu']}
d3={'roll_no':[10,21,12,13,24,15],'name':['Jeet','Ashima','Shivin','Kiran','Tanmay','Rajat']}
df1=pd.DataFrame(d1)
df2=pd.DataFrame(d2)
df3=pd.concat([df1,df2])
print(df3)
df4=pd.DataFrame(d3)
df5=pd.merge(df3,df4,on='roll_no')
print(df5)
```

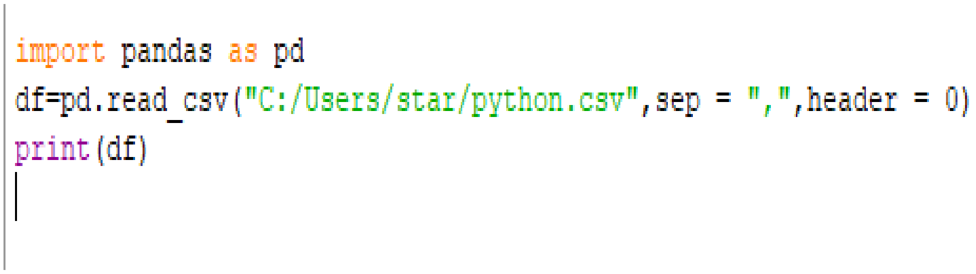
OUTPUT-



```
========================================
   roll_no     name
0       10    Ankit
1       11     Pihu
2       12    Rinku
3       13     Yash
4       14    Vijay
5       15   Nikhil
0       20  Shaurya
1       21    Pinky
2       22  Anubhav
3       23   Khushi
4       24    Vinay
5       25    Neetu
   roll_no  name_x  name_y
0       10   Ankit    Jeet
1       12   Rinku  Shivin
2       13    Yash   Kiran
3       15  Nikhil   Rajat
4       21   Pinky  Ashima
5       24   Vinay  Tanmay
>>>
```

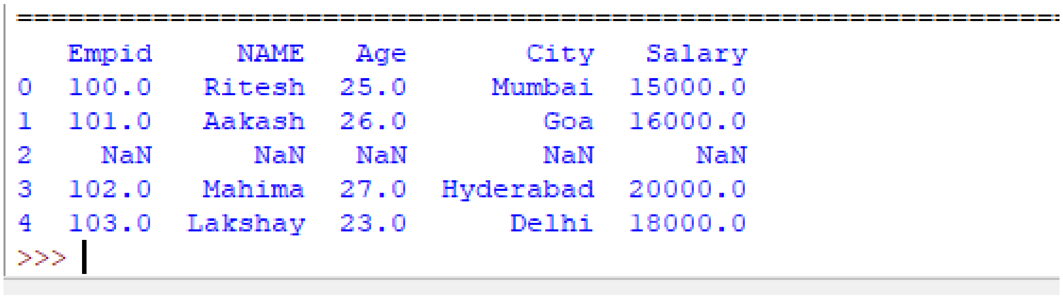
12. IMPORTING CSV TO DATAFRAME

CODE-



```
import pandas as pd
df=pd.read_csv("C:/Users/star/python.csv",sep = ",",header = 0)
print(df)
```

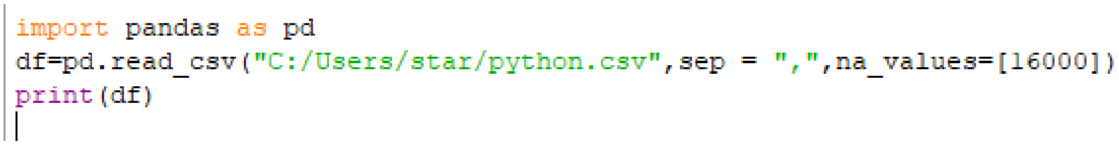
OUTPUT-



```
=====================================================================
   Empid     NAME   Age       City   Salary
0  100.0   Ritesh  25.0     Mumbai  15000.0
1  101.0   Aakash  26.0        Goa  16000.0
2    NaN      NaN   NaN        NaN      NaN
3  102.0   Mahima  27.0  Hyderabad  20000.0
4  103.0  Lakshay  23.0      Delhi  18000.0
>>> 
```

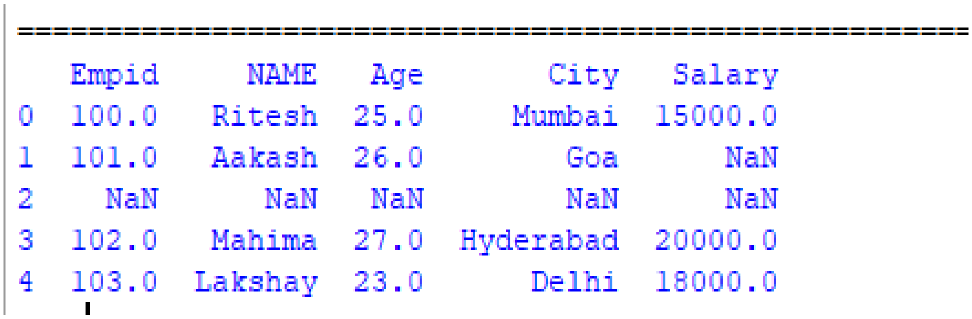
13. TO MODIFY THE SALARY OF EMPLOYEE EARNING 16000

CODE-



```
import pandas as pd
df=pd.read_csv("C:/Users/star/python.csv",sep = ",",na_values=[16000])
print(df)
```

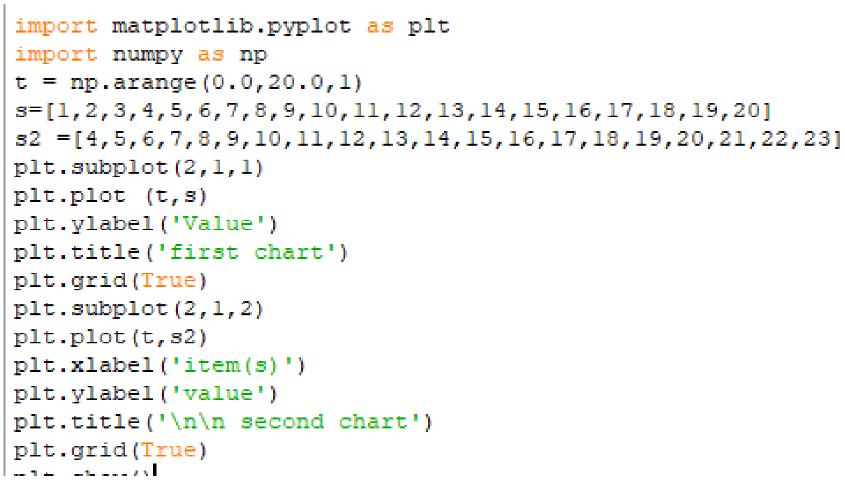
OUTPUT-



```
======================================================
   Empid     NAME   Age       City   Salary
0  100.0   Ritesh  25.0     Mumbai  15000.0
1  101.0   Aakash  26.0        Goa      NaN
2    NaN      NaN   NaN        NaN      NaN
3  102.0   Mahima  27.0  Hyderabad  20000.0
4  103.0  Lakshay  23.0      Delhi  18000.0
```

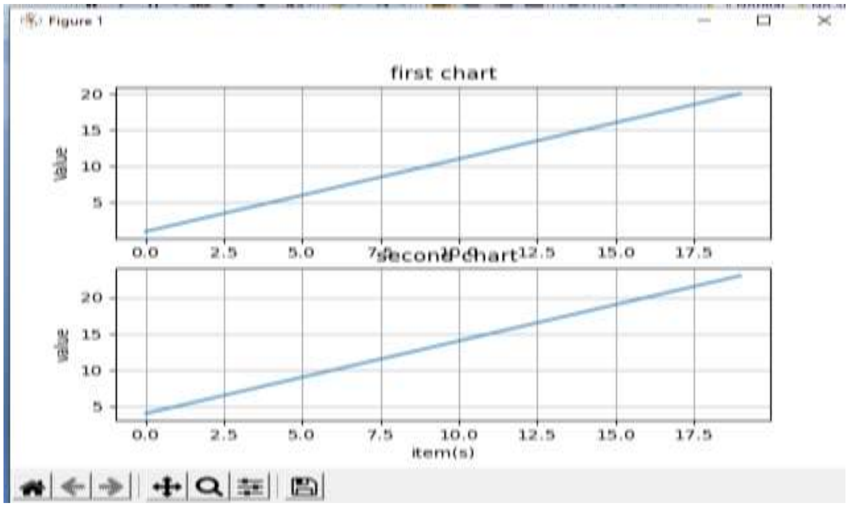
14. TO PLOT TWO LINES IN TWO DIFFERENT VIEWS OF THE SAME WINDOW

CODE-



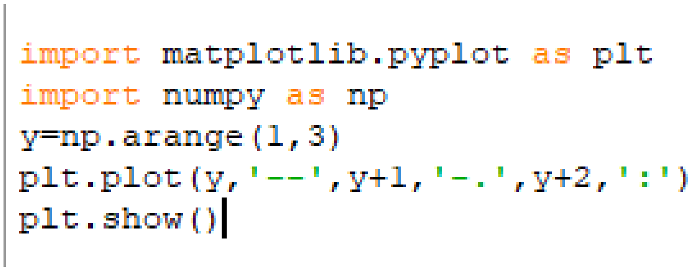
```
import matplotlib.pyplot as plt
import numpy as np
t = np.arange(0.0,20.0,1)
s=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]
s2 =[4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23]
plt.subplot(2,1,1)
plt.plot (t,s)
plt.ylabel('Value')
plt.title('first chart')
plt.grid(True)
plt.subplot(2,1,2)
plt.plot(t,s2)
plt.xlabel('item(s)')
plt.ylabel('value')
plt.title('\n\n second chart')
plt.grid(True)
```

OUTPUT-



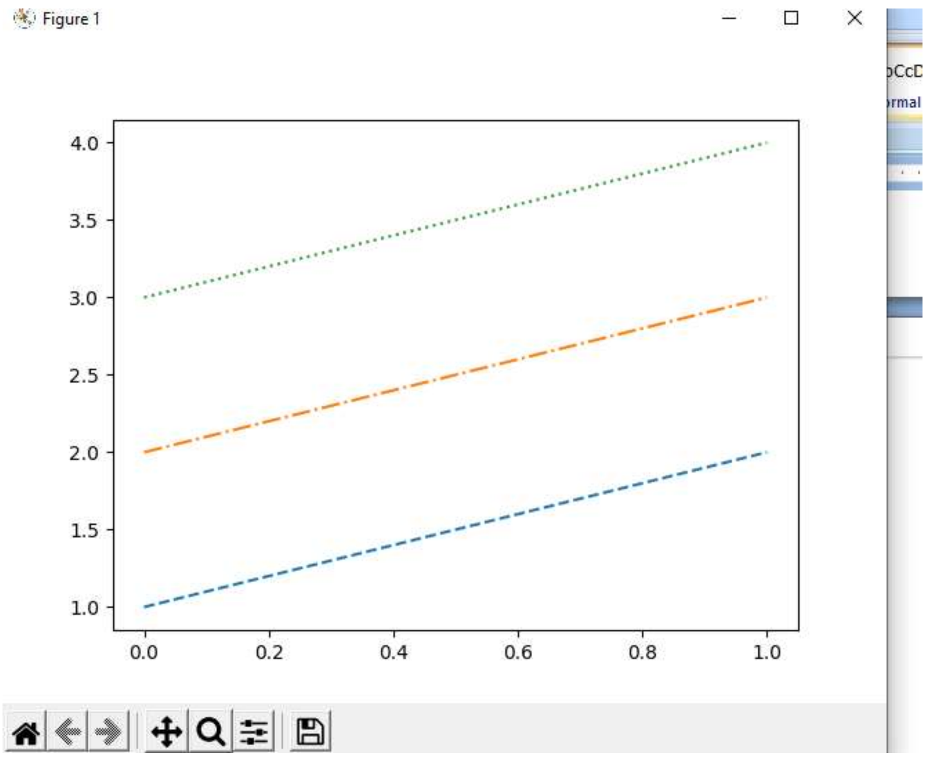
15. PLOTTING LINES WITH DIFFERENT STYLES USING plot () FUNCTION

CODE-



```
import matplotlib.pyplot as plt
import numpy as np
y=np.arange(1,3)
plt.plot(y,'--',y+1,'-.',y+2,':')
plt.show()
```

OUTPUT -

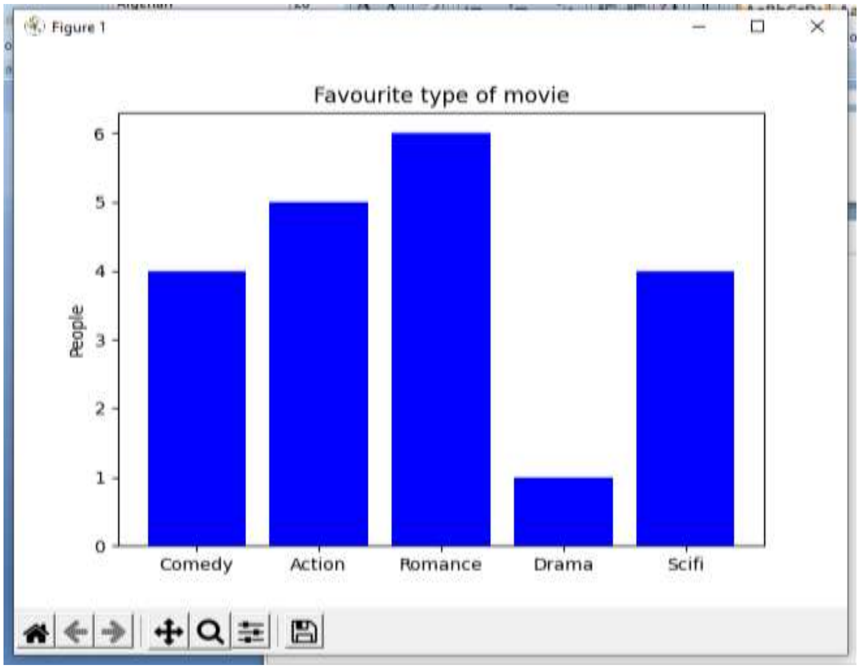


16. PLOTTING A BAR CHART

CODE-

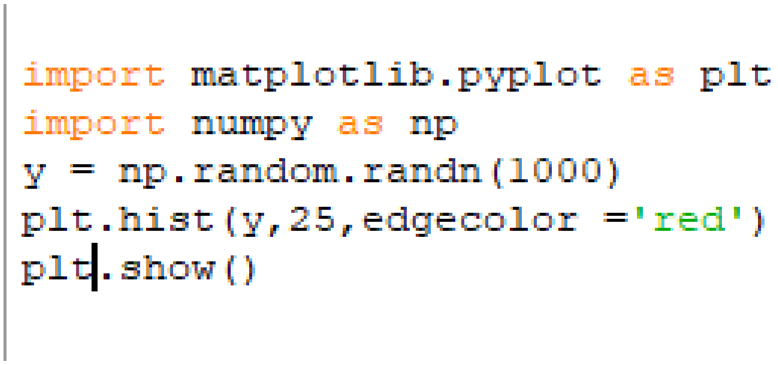
```
import matplotlib.pyplot as plt
import numpy as np
objects = ('Comedy','Action','Romance','Drama','Scifi')
y_pos = np.arange(len(objects))
Types = (4,5,6,1,4)
plt.bar(y_pos,Types,align ='center',color = 'blue')
plt.xticks(y_pos,objects)
plt.ylabel('People')
plt.title('Favourite type of movie')
plt.show()
import matplotlib.pyplot as plt
import numpy as np
y=np.random.randn(1000)
plt.hist(y,25,edgecolor ='red')
plt.show()
```

OUTPUT -



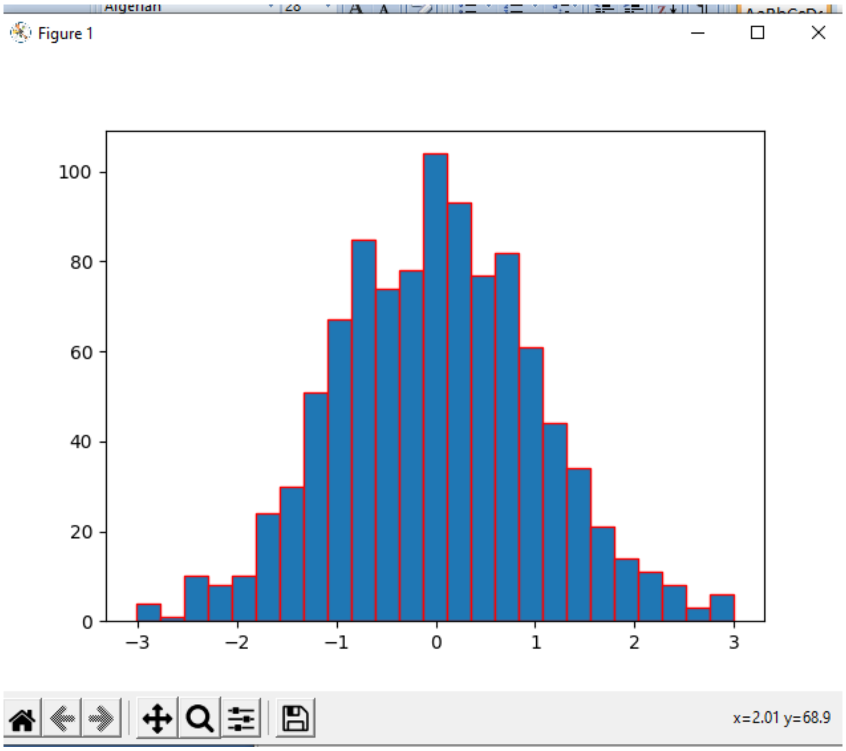
17. DISPLAY A HISTOGRAM

CODE-



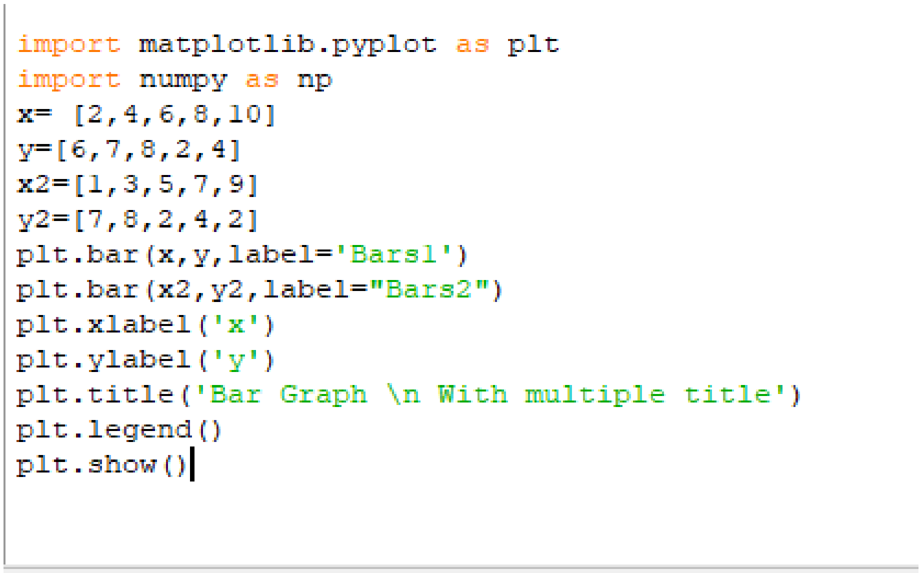
```python
import matplotlib.pyplot as plt
import numpy as np
y = np.random.randn(1000)
plt.hist(y,25,edgecolor ='red')
plt.show()
```

OUTPUT-



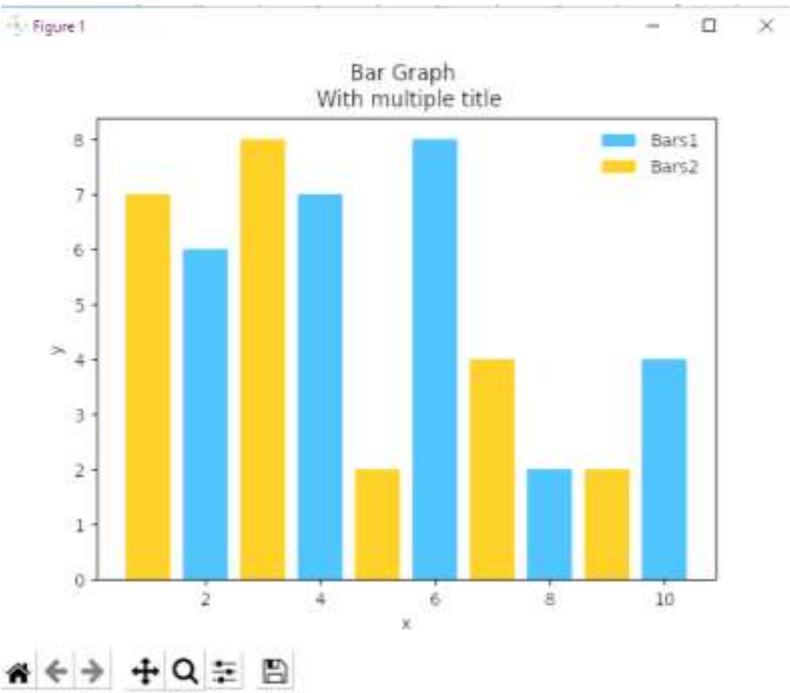
18. PLOT ELEMENTS OF TWO LISTS USING BAR CHART

CODE-



```
import matplotlib.pyplot as plt
import numpy as np
x= [2,4,6,8,10]
y=[6,7,8,2,4]
x2=[1,3,5,7,9]
y2=[7,8,2,4,2]
plt.bar(x,y,label='Bars1')
plt.bar(x2,y2,label="Bars2")
plt.xlabel('x')
plt.ylabel('y')
plt.title('Bar Graph \n With multiple title')
plt.legend()
plt.show()
```

OUTPUT-



BIBLIOGRAPHY

- NCERT
- PREETI ARORA
- INTERNET

Thank You