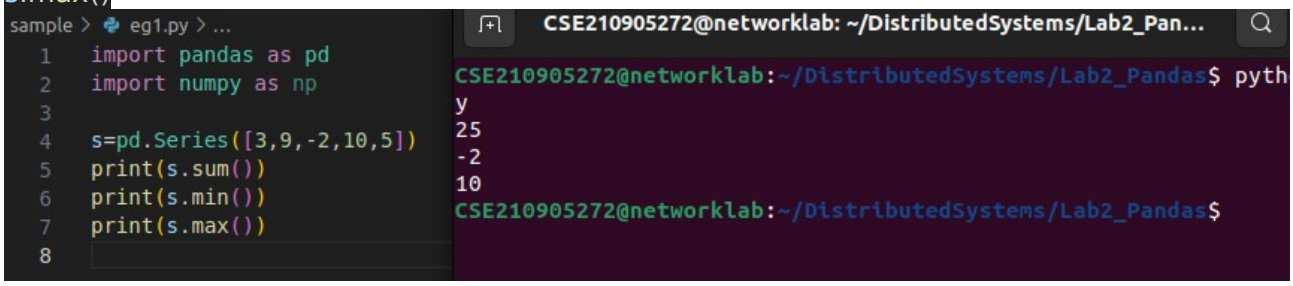
LAB – 2
PYTHON BASIC PRACTICE – II

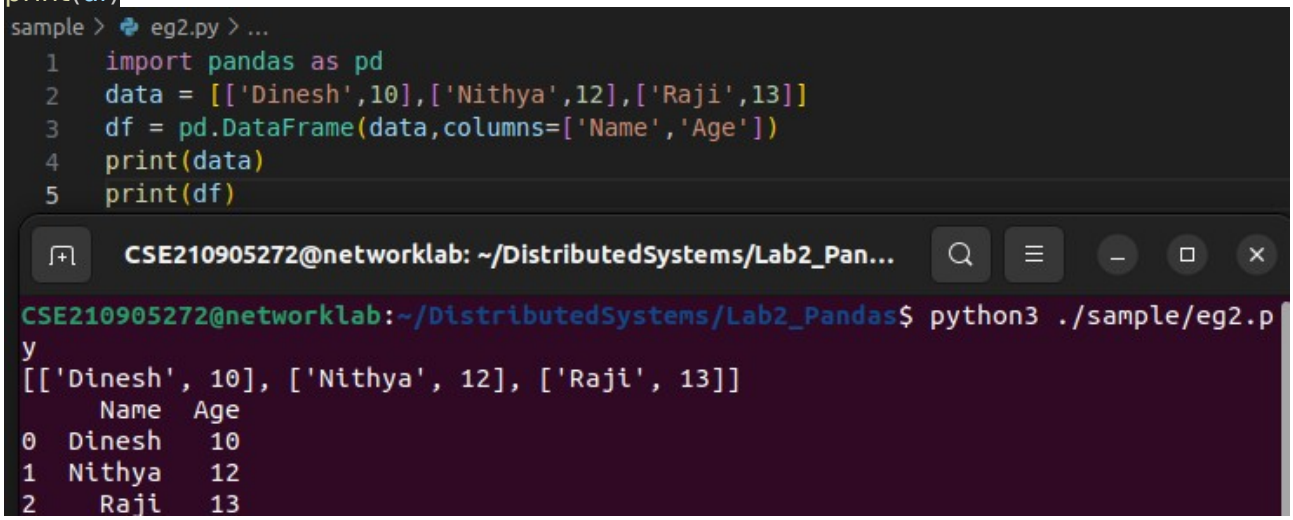
Introduction to PANDAS

```
import pandas as pd
import numpy as np

s=pd.Series([3,9,-2,10,5])
s.sum()
s.min()
s.max()
```



Creating a Data Frame

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



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(data)
```

```
print(df)
```

```
sample > eg3.py > ...
1  import pandas as pd
2
3  data = {'Name':['Kavitha', 'Sudha', 'Raju','Vignesh'],'Age':[28,34,29,42]}
4  df = pd.DataFrame(data, index=['rank1','rank2','rank3','rank4'])
5
6  print(data)
7  print(df)
```

```
CSE210905272@networklab: ~/DistributedSystems/Lab2_Pandas
CSE210905272@networklab:~/DistributedSystems/Lab2_Pandas$ python3 ./sample/eg3.py
{'Name': ['Kavitha', 'Sudha', 'Raju', 'Vignesh'], 'Age': [28, 34, 29, 42]}
          Name  Age
rank1  Kavitha   28
rank2    Sudha   34
rank3     Raju   29
rank4  Vignesh   42
```

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'])})

print(df1.shape)
print(df1.dtypes)
print(df1.head())
print(df1.tail())
# print(df1.summary())
print(df1.T)
```

```
CSE210905272@networklab:~/DistributedSystems/Lab2_Panda
s$ python3 ./sample/eg4.py
(4, 3)
A    datetime64[s]
B            int32
C         category
dtype: object
           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
           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
                     0  ...                   3
A  2013-01-02 00:00:00  ...  2013-01-02 00:00:00
B                    3  ...                   3
C                 Male  ...              Female

[3 rows x 4 columns]
```

```
1  #Creating a DataFrame using Dictionary
2
3  import pandas as pd
4  import numpy as np
5
6  df1=pd.DataFrame({'A':pd.Timestamp('20130102'),'B':np.array([3]*4,dtype='int32'),'C
7
8  print(df1.shape)
9  print(df1.dtypes)
10 print(df1.head())
11 print(df1.tail())
12 # print(df1.summary())
13 print(df1.T)
```

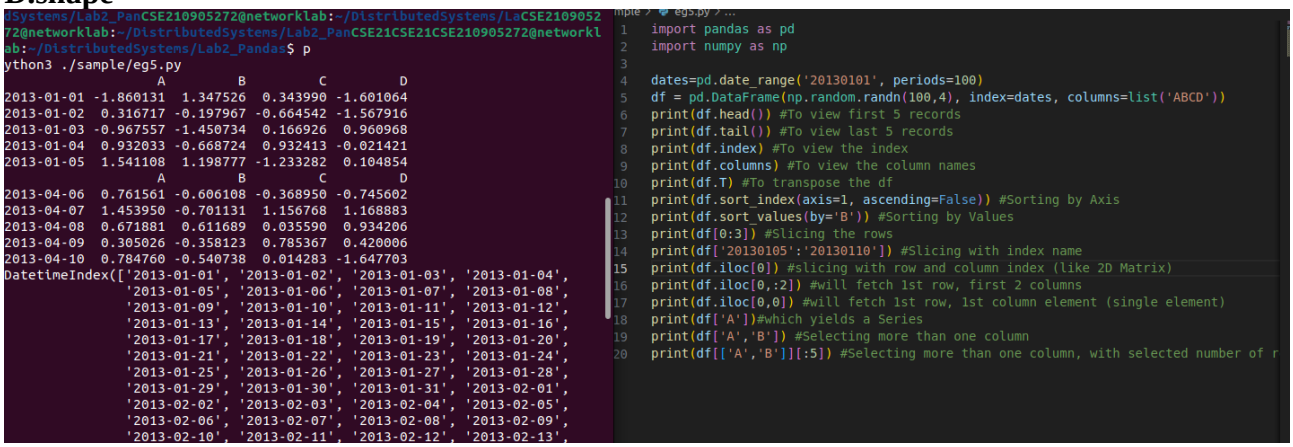
.

Creating a DataFrame using Dictionary

```
import pandas as pd
import numpy as np
```

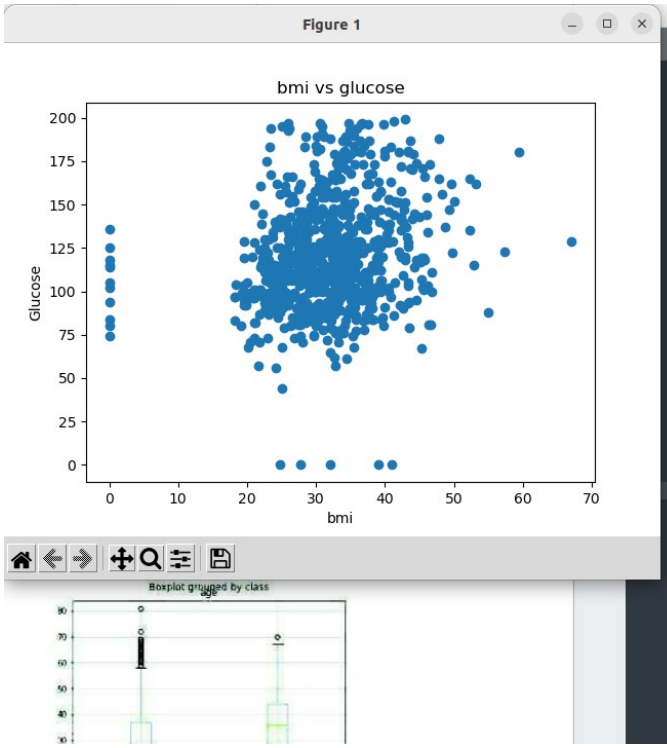
```
dates=pd.date_range('20130101', periods=100)
df = pd.DataFrame(np.random.randn(100,4), index=dates, columns=list('ABCD'))
print(df.head()) #To view first 5 records
print(df.tail()) #To view last 5 records
print(df.index) #To view the index
print(df.columns) #To view the column names
print(df.T) #To transpose the df
print(df.sort_index(axis=1, ascending=False)) #Sorting by Axis
print(df.sort_values(by='B')) #Sorting by Values
print(df[0:3]) #Slicing the rows
print(df['20130105':'20130110']) #Slicing with index name
print(df.iloc[0]) #slicing with row and column index (like 2D Matrix)
print(df.iloc[0,:2]) #will fetch 1st row, first 2 columns
print(df.iloc[0,0]) #will fetch 1st row, 1st column element (single element)
print(df['A'])#which yields a Series
print(df['A','B']) #Selecting more than one column
print(df[['A','B']][:5]) #Selecting more than one column, with selected number of records
```

Boolean Indexing
df[df.A>0], will fetch all positive values of A column
Include a 6th column (a categorical) character data
df['F']=['Male','Female','Female','Male','Female','Female']
Setting by assigning with a numpy array
df.loc[:,'D']=np.array([5]*len(df))
Deleting a row or column
df.drop ('col_name', axis =1, inplace=True)
will drop the column name specified in col_name
df.drop ('row_index', axis =0, inplace=True)
Df_new= pd.concat (df1, df2, axis=1)
Df_new.shape
D= pd.concat (A, B, axis=0)
D.shape



.

I/O operations



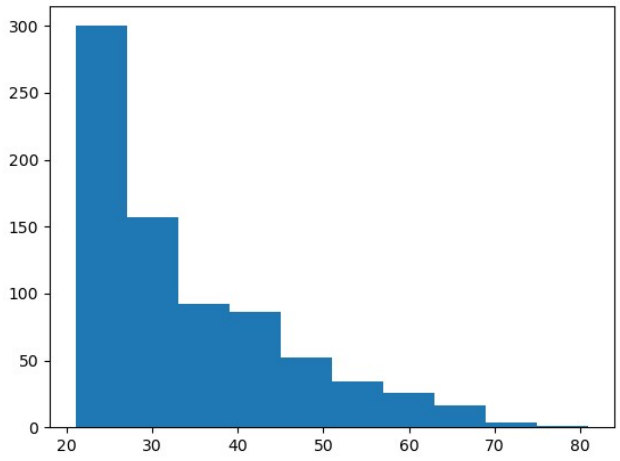
Selection Find View Goto Tools Project Pref

eg6.py

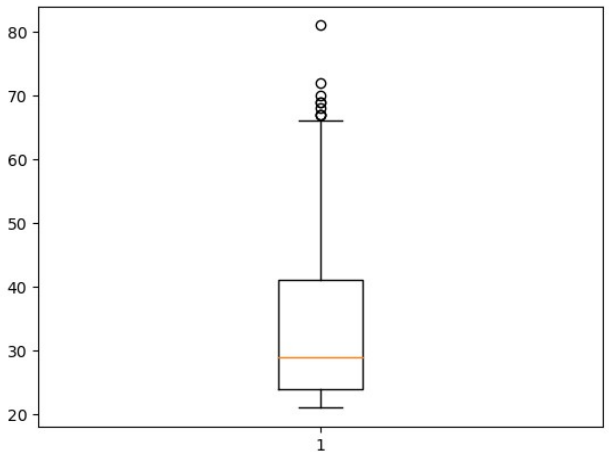
```
import pandas as pd
import matplotlib.pyplot as plt
# %matplotlib inline

df=pd.read_csv('././lab2_req_files/xyz.csv',
print(df.head())
print(df.tail())
#attach header
df.columns=['preg','glu','bp','sft','ins','

#Let us visualize the scatter plot of two co

plt.scatter(df['bmi'],df['glu'])
plt.xlabel('bmi')
plt.ylabel('Glucose')
plt.title('bmi vs glucose')
plt.show()

#Let us visualize the histogram of another
plt.hist(df['age'])
plt.show()

#box plot
plt.boxplot(df['age'])
plt.show()
```

CSE210905272@networklab: ~/DistributedSystems/L...

```
1  1   85  66  29    0  26.6  0.351  31  0
2  8  183  64   0    0  23.3  0.672  32  1
3  1   89  66  23   94  28.1  0.167  21  0
4  0  137  40  35  168  43.1  2.288  33  1
       0    1   2   3    4     5      6   7  8
763  10  101  76  48  180  32.9  0.171  63  0
764   2  122  70  27    0  36.8  0.340  27  0
765   5  121  72  23  112  26.2  0.245  30  0
766   1  126  60   0    0  30.1  0.349  47  1
767   1   93  70  31    0  30.4  0.315  23  0
CSE210905272@networklab:~/DistributedSystems/Lab2_P
/eg6.py
     0    1   2   3    4     5      6   7  8
0    6  148  72  35    0  33.6  0.627  50  1
1    1   85  66  29    0  26.6  0.351  31  0
2    8  183  64   0    0  23.3  0.672  32  1
3    1   89  66  23   94  28.1  0.167  21  0
4    0  137  40  35  168  43.1  2.288  33  1
       0    1   2   3    4     5      6   7  8
763  10  101  76  48  180  32.9  0.171  63  0
764   2  122  70  27    0  36.8  0.340  27  0
765   5  121  72  23  112  26.2  0.245  30  0
766   1  126  60   0    0  30.1  0.349  47  1
767   1   93  70  31    0  30.4  0.315  23  0
CSE210905272@networklab:~/DistributedSystems/Lab2_P
/eg6.py
     0    1   2   3    4     5      6   7  8
0    6  148  72  35    0  33.6  0.627  50  1
1    1   85  66  29    0  26.6  0.351  31  0
2    8  183  64   0    0  23.3  0.672  32  1
3    1   89  66  23   94  28.1  0.167  21  0
4    0  137  40  35  168  43.1  2.288  33  1
       0    1   2   3    4     5      6   7  8
763  10  101  76  48  180  32.9  0.171  63  0
764   2  122  70  27    0  36.8  0.340  27  0
765   5  121  72  23  112  26.2  0.245  30  0
766   1  126  60   0    0  30.1  0.349  47  1
767   1   93  70  31    0  30.4  0.315  23  0
```




```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3  # %matplotlib inline
4
5  df=pd.read_csv('././lab2_req_files/xyz.csv',header=None)
6  print(df.head())
7  print(df.tail())
8  #attach header
9  df.columns=['preg','glu','bp','sft','ins','bmi','dpf','age','class']
10
11 #Let us visualize the scatter plot of two continuous variable
12
13 plt.scatter(df['bmi'],df['glu'])
14 plt.xlabel('bmi')
15 plt.ylabel('Glucose')
16 plt.title('bmi vs glucose')
17 plt.show()
18
19 #Let us visualize the histogram of another continuous variable 'Age'
20 plt.hist(df['age'])
21 plt.show()
22
23 #box plot
24 plt.boxplot(df['age'])
25 plt.show()
```




```python
import pandas as pd
import matplotlib.pyplot as plt
# %matplotlib inline

df=pd.read_csv('./lab2_req_files/xyz.csv',header=None)
print(df.head())
print(df.tail())
#attach header
df.columns=['preg','glu','bp','sft','ins','bmi','dpf','age','class']

#Let us visualize the scatter plot of two continuous variable

plt.scatter(df['bmi'],df['glu'])
plt.xlabel('bmi')
plt.ylabel('Glucose')
plt.title('bmi vs glucose')
plt.show()

#Let us visualize the histogram of another continuous variable 'Age'
plt.hist(df['age'])
plt.show()

#box plot
plt.boxplot(df['age'])
plt.show()
```

#W = pd.read_csv('xyz.xls',header=None)
#W.head() #XLS file format also, we can read using pd.read_csv
#D= np.loadtxt('xyz.data',delimiter=",")
#D[:5,:] # this file is loaded in Numpy 2D array format
 # Reading a XLSX file format
#G=pd.read_excel(xyz.xlsx',sheet_name='Sheet1')
#Pandas can read table tabs off of html. For example:
B = pd.read_html('./lab2_req_files/Test runs-1.html')

```
CSE210905272@networklab:~/DistributedSystems/Lab2_Pandas$ python3 ./sample
/eg7.py
     0      1     2     3     4   ...     9      10    11    12    13
0     1  14.23  1.71  2.43  15.6  ...  2.29   5.64  1.04  3.92  1065
1     1  13.20  1.78  2.14  11.2  ...  1.28   4.38  1.05  3.40  1050
2     1  13.16  2.36  2.67  18.6  ...  2.81   5.68  1.03  3.17  1185
3     1  14.37  1.95  2.50  16.8  ...  2.18   7.80  0.86  3.45  1480
4     1  13.24  2.59  2.87  21.0  ...  1.82   4.32  1.04  2.93   735
..   ..    ...   ...   ...   ...  ...   ...    ...   ...   ...   ...
173   3  13.71  5.65  2.45  20.5  ...  1.06   7.70  0.64  1.74   740
174   3  13.40  3.91  2.48  23.0  ...  1.41   7.30  0.70  1.56   750
175   3  13.27  4.28  2.26  20.0  ...  1.35  10.20  0.59  1.56   835
176   3  13.17  2.59  2.37  20.0  ...  1.46   9.30  0.60  1.62   840
177   3  14.13  4.10  2.74  24.5  ...  1.35   9.20  0.61  1.60   560

[178 rows x 14 columns]
```

```
eg6.py | eg7.py
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  w = pd.read_csv('./lab2_req_files/wine_for_Week2.xls',header=Non
5
6  print(w)
```

```
CSE210905272@networklab:~/DistributedSystems/Lab2_Pandas$ python3 ./sample
/eg8.py
[   Unnamed: 0   0    1   2   3    4     5      6   7  8
0           763  10  101  76  48  180  32.9  0.171  63  0
1           764   2  122  70  27    0  36.8  0.340  27  0
2           765   5  121  72  23  112  26.2  0.245  30  0
3           766   1  126  60   0    0  30.1  0.349  47  1
4           767   1   93  70  31    0  30.4  0.315  23  0,    Unnamed: 0  pr
eg  glu  bp  sft  ins   bmi    dpf  age  class
0           0     6  148  72   35    0  33.6  0.627   50      1
1           1     1   85  66   29    0  26.6  0.351   31      0
2           2     8  183  64    0    0  23.3  0.672   32      1
3           3     1   89  66   23   94  28.1  0.167   21      0
4           4     0  137  40   35  168  43.1  2.288   33      1,    Unname
d: 0  0      1     2     3  ...     9    10    11    12    13
0           0  1  14.23  1.71  2.43  ...  2.29  5.64  1.04  3.92  1065
1           1  1  13.20  1.78  2.14  ...  1.28  4.38  1.05  3.40  1050
2           2  1  13.16  2.36  2.67  ...  2.81  5.68  1.03  3.17  1185
3           3  1  14.37  1.95  2.50  ...  2.18  7.80  0.86  3.45  1480
4           4  1  13.24  2.59  2.87  ...  1.82  4.32  1.04  2.93   735

[5 rows x 15 columns],    Unnamed: 0  0      1     2     3  ...     9    1
0    11    12   13
0         173  3  13.71  5.65  2.45  ...  1.06   7.7  0.64  1.74  740
1         174  3  13.40  3.91  2.48  ...  1.41   7.3  0.70  1.56  750
```

```
q1.py | q2.py | q3.py
1  import pandas as pd
2  B = pd.read_html('./lab2_req_files/Test runs-1.html')
3  print(B)
```

Reading a TXT file format
H = pd.read_table('HR_for_week2.txt')

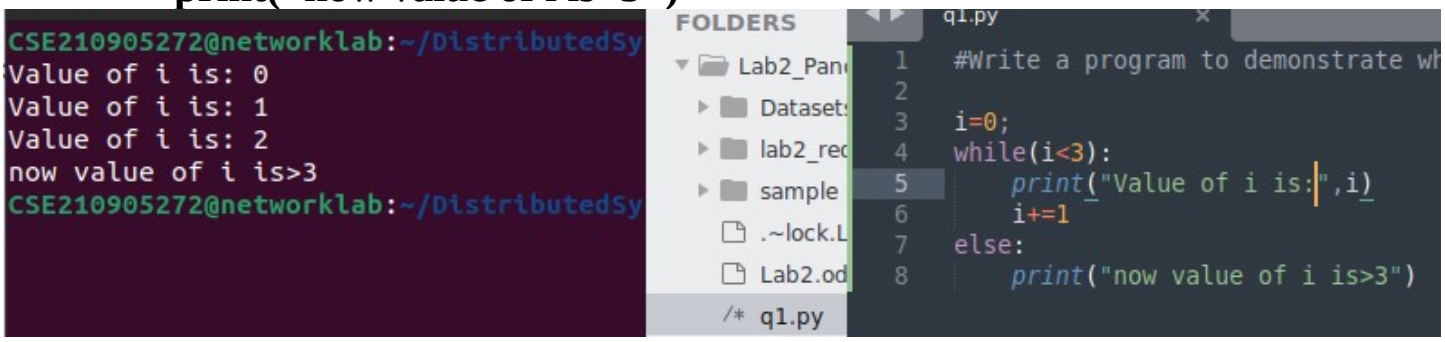
.

LAB EXERCISES

Q1)Write a program to demonstrate while loop with else

->

```
i=0;
while(i<3):
        print("Value of i is:",i)
        i+=1
else:
        print("now value of i is>3")
```

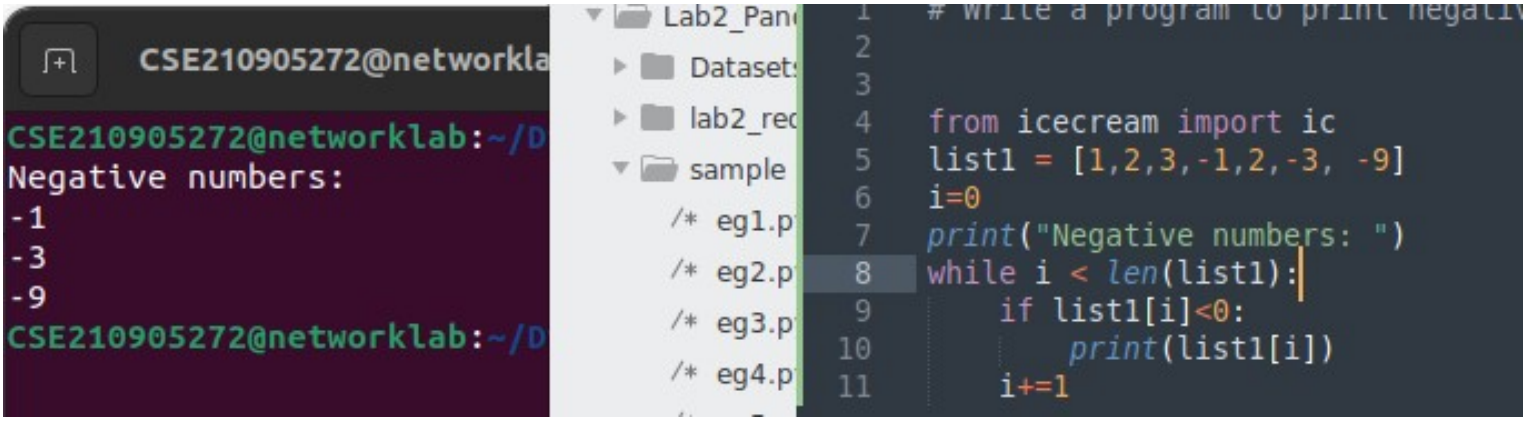


.

Q2)Write a program to print negative Numbers in a List using while loop.

->

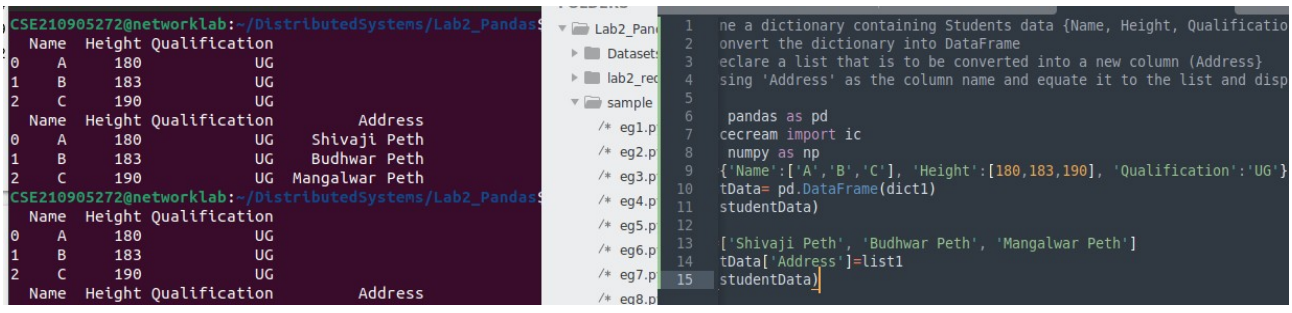
```
from icecream import ic
list1 = [1,2,3,-1,2,-3, -9]
i=0
print("Negative numbers: ")
while i < len(list1):
        if list1[i]<0:
                print(list1[i])
        i+=1
```



Q3Define 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
from icecream import ic
import numpy as np
dict1={'Name':['A','B','C'], 'Height':[180,183,190],
'Qualification':'UG'}
studentData= pd.DataFrame(dict1)
print(studentData)

list1=['Shivaji Peth', 'Budhwar Peth', 'Mangalwar Peth']
studentData['Address']=list1
print(studentData)
```

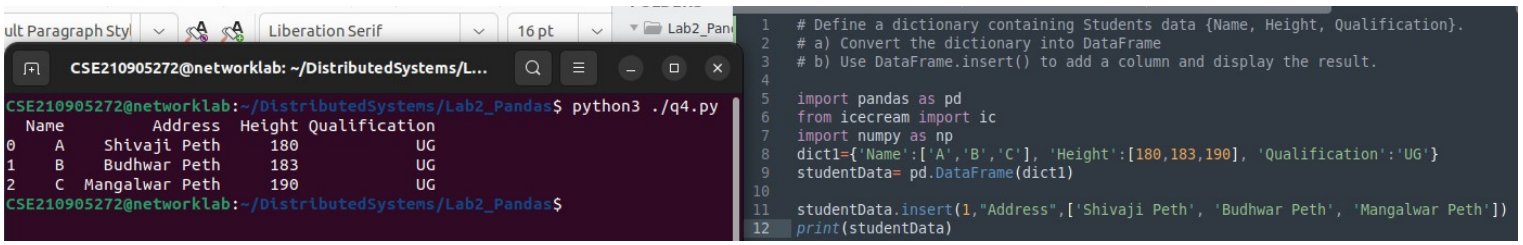


.

Q4) 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
from icecream import ic
import numpy as np
dict1={'Name':['A','B','C'], 'Height':[180,183,190],
'Qualification':'UG'}
studentData= pd.DataFrame(dict1)

studentData.insert(1,"Address",['Shivaji Peth', 'Budhwar Peth',
'Mangalwar Peth'])
print(studentData)
```



.

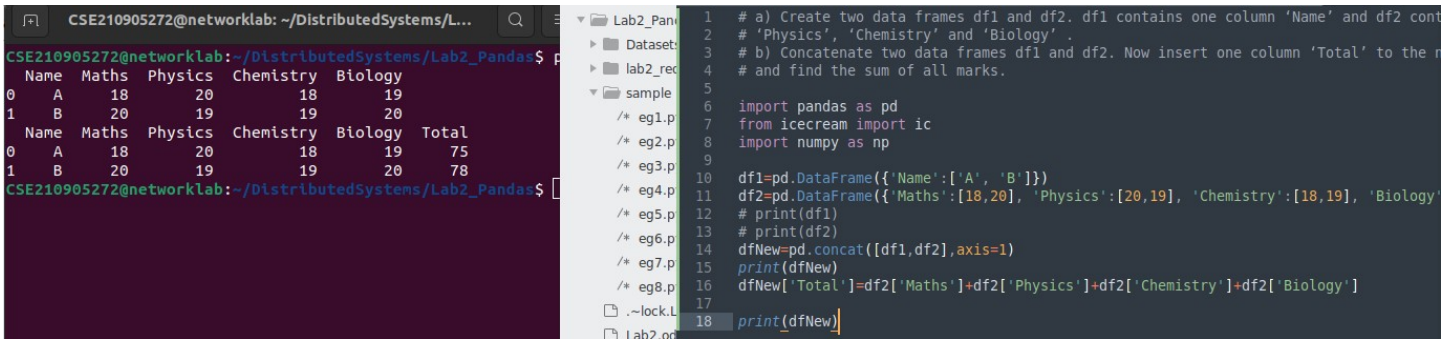
Q5)
a) Create two data frames df1 and df2. df1 contains one column 'Name' and df2 contains 4 columns 'Maths', 'Physics', 'Chemistry' and 'Biology' .
b) Concatenate two data frames df1 and df2. Now insert one column 'Total' to the new data frame df_new and find the sum of all marks.

->

```
import pandas as pd
from icecream import ic
import numpy as np

df1=pd.DataFrame({'Name':['A', 'B']})
df2=pd.DataFrame({'Maths':[18,20], 'Physics':[20,19], 'Chemistry':
[18,19], 'Biology':[19,20]})
```

 # print(df1)
 # print(df2)
dfNew=pd.concat([df1,df2],axis=1)
print(dfNew)
dfNew['Total']=df2['Maths']+df2['Physics']+df2['Chemistry']
+df2['Biology']

print(dfNew)

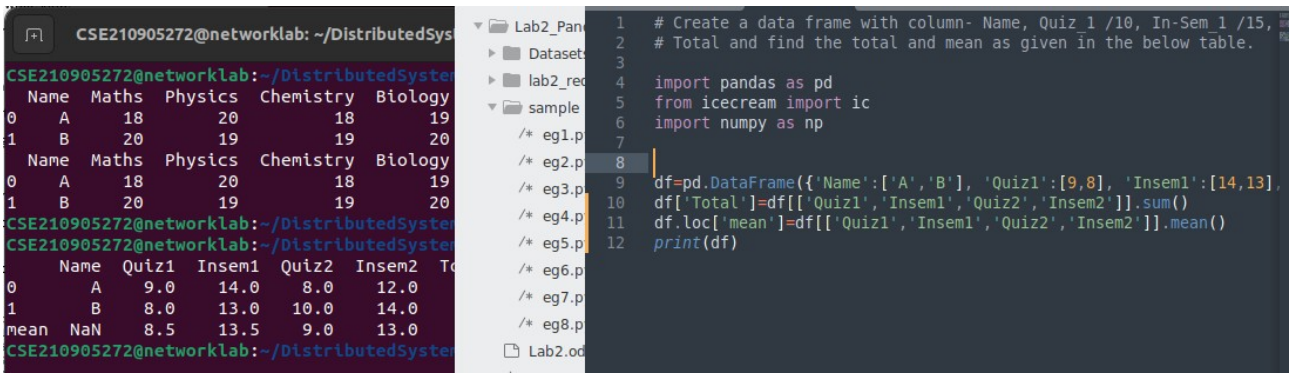


.

Q6) 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
from icecream import ic
import numpy as np

df=pd.DataFrame({'Name':['A','B'], 'Quiz1':[9,8], 'Insem1':[14,13], 'Quiz2':[8,10], 'Insem2':[12,14]})
df['Total']=df[['Quiz1','Insem1','Quiz2','Insem2']].sum()
df.loc['mean']=df[['Quiz1','Insem1','Quiz2','Insem2']].mean()
print(df)


CSE210905272@networklab: ~/DistributedSys
CSE210905272@networklab:~/DistributedSyster
Name Maths Physics Chemistry Biology
0 A 18 20 18 19
1 B 20 19 19 20
Name Maths Physics Chemistry Biology
0 A 18 20 18 19
1 B 20 19 19 20
CSE210905272@networklab:~/DistributedSyster
CSE210905272@networklab:~/DistributedSyster
Name Quiz1 Insem1 Quiz2 Insem2 To
0 A 9.0 14.0 8.0 12.0
1 B 8.0 13.0 10.0 14.0
mean NaN 8.5 13.5 9.0 13.0
CSE210905272@networklab:~/DistributedSyster
Lab2_Pan
Datasets
lab2_rec
sample
eg1.p
eg2.p
eg3.p
eg4.p
eg5.p
eg6.p
eg7.p
eg8.p
Lab2.od
# Create a data frame with column- Name, Quiz_1 /10, In-Sem_1 /15,
# Total and find the total and mean as given in the below table.
import pandas as pd
from icecream import ic
import numpy as np
df=pd.DataFrame({'Name':['A','B'], 'Quiz1':[9,8], 'Insem1':[14,13],
df['Total']=df[['Quiz1','Insem1','Quiz2','Insem2']].sum()
df.loc['mean']=df[['Quiz1','Insem1','Quiz2','Insem2']].mean()
print(df)