**Name: Varsha JJ**

**Roll No: 48**

**Batch: MCA-B**

**Date: 01/09/2022**

**DATA SCIENCE LAB**

**Experiment No.: 1**

**Aim**

How to create Series with nd array.

**Procedure**

import pandas as pd

import numpy as np

arr=np.array([10,15,18,22])

s = pd.Series(arr)

print(s)

**Output**

0 10

1 15

2 18

3 22

dtype: int64

**Experiment No.: 2**

**Aim**

How to create Series with Mutable index

**Procedure**

import pandas as pd

import numpy as np

arr=np.array(['a','b','c','d'])

s=pd.Series(arr,

index=['first','second','third','fourth'])

print(s)

**Output**

first a

second b

third c

fourth d

dtype: object

**Experiment No.: 3**

**Aim**

Creating a series from a Dictionary.

**Procedure**

import pandas as pd

s={'name':'hardik','iplteam':'mi','runs':100}

p=pd.Series(s)

print(p)

**Output**

name hardik

iplteam mi

runs 100

dtype: object

**Experiment No.: 4**

**Aim**

Print all the values of the Series by multiplying them by 2.

**Procedure**

import pandas as pd

p=pd.Series([1,2,3,4,5])

print(p)

print("multlipling all values in series by 2")

print(p\*2)

**Output**

0 1

1 2

2 3

3 4

4 5

dtype: int64

multlipling all values in series by 2

0 2

1 4

2 6

3 8

4 10

dtype: int64

**Experiment No.: 5**

**Aim**

Print Square of all the values of the series.

**Procedure**

import pandas as pd

p=pd.Series([1,2,3,4,5])

print('..............................................')

print("square of all values")

print(p\*\*2)

**Output**

0 1

1 2

2 3

3 4

4 5

dtype: int64

square of all values

0 1

1 4

2 9

3 16

4 25

dtype: int64

**Experiment No.: 6**

**Aim**

Print all the values of the Series that are greater than 2.

**Procedure**

import pandas as pd

p=pd.Series([1,2,3,4,5])

print("when the value greater than 2")

print(p[p>2])

print('..............................................')

**Output**

when the value greater than 2

2 3

3 4

4 5

dtype: int64

**Experiment No.: 7**

**Aim**

Addition of two series.

**Procedure**

import pandas as pd

s1=pd.Series([1,2,3,4,5],index=['a','b','c','d','e'])

s2=pd.Series([1,2,3,4,5],index=['a','b','c','d','e'])

print(s1)

print(s2)

print(s1+s2)

**Output**

a 1

b 2

c 3

d 4

e 5

dtype: int64

a 1

b 2

c 3

d 4

e 5

dtype: int64

a 2

b 4

c 6

d 8

e 10

dtype: int64

**Experiment No.: 8**

**Aim**

Print the first and last 5 elements of a series.

**Procedure**

import pandas as pd

import numpy as np

arr=np.array([10,12,23,3,4,56,57,6,7])

s=pd.Series(arr)

print(s.head(5))

**Output**

0 10

1 12

2 23

3 3

4 4

dtype: int64

**Experiment No.: 9**

**Aim**

Print the values from index 0 to 5.

**Procedure**

import pandas as pd

import numpy as np

arr=np.array([10,12,23,3,4,56,57,6,7])

s=pd.Series(arr)

print(s.head(6))

**Output**

0 10

1 12

2 23

3 3

4 4

5 56

dtype: int64

**Experiment No.: 10**

**Aim**

Selection Using loc, iloc index label.

**Procedure**

import pandas as pd

import numpy as np

arr=np.array([10,12,23,3,4,56,57,6,7])

s=pd.Series(arr)

print(s)

print(s.loc[:2])

print(s.iloc[3:4])

**Output**

0 10

1 12

2 23

3 3

4 4

5 56

6 57

7 6

8 7

dtype: int64

0 10

1 12

2 23

dtype: int64

3 3

dtype: int64

**Experiment No.: 11**

**Aim**

Retrieve subsets of data using slicing.

**Procedure**

import pandas as pd

import numpy as np

arr=np.array([10,12,23,3,4])

s=pd.Series(arr,index=['A','B','C','D','E'])

print(s)

print(s[::-1])

**Output**

A 10

B 12

C 23

D 3

E 4

dtype: int64

E 4

D 3

C 23

B 12

A 10

dtype: int64

**Dataframe**

**Experiment No.: 1**

**Aim**

create Dataframe From Series

**Procedure**

import pandas as pd

s = pd.Series(['a','b','c','d'])

df=pd.DataFrame(s)

print(df)

**Output**

0

0 a

1 b

2 c

3 d

**Experiment No.: 2**

**Aim**

Data Frame from List of Dictionaries

**Procedure**

import pandas as pd

l=[{'Name':'sachin','city':'kerala'},

{'Name':'virat','city':'tamilnadu'}]

d=pd.DataFrame(l)

print(d)

**Output**

Name city

0 sachin kerala

1. virat tamilnadu

**Experiment No.: 3**

**Aim**

Display the first 5 rows of data frame.

**Procedure**

import pandas as pd

empdata = {'empid':[1,2,3,4,5,6],'ename':['Vimal','Sachin','Bav','Kumar','Ravy','Sunil']}

df=pd.DataFrame(empdata)

print(df)

print(df.head(5))

**Output**

empid ename

0 1 Vimal

1 2 Sachin

2 3 Bav

3 4 Kumar

4 5 Ravy

5 6 Sunil

empid ename

0 1 Vimal

1 2 Sachin

2 3 Bav

3 4 Kumar

1. 5 Ravy

**Experiment No.: 4**

**Aim**

Select the last two columns of the data frame

**Procedure**

import pandas as pd

empdata = {'empid':[1,2,3,4,5,6], 'ename':['Vimal','Sachin','Bav','Kumar','Ravy','Sunil']}

df=pd.DataFrame(empdata)

print(df)

df.loc[0:5]

print(df.tail(2))

**Output**

empid ename

0 1 Vimal

1 2 Sachin

2 3 Bav

3 4 Kumar

4 5 Ravy

5 6 Sunil

empid ename

4 5 Ravy

5 6 Sunil

**Experiment No.: 5**

**Aim**

Demonstrate deletion, and renaming of columns.

**Procedure**

import pandas as pd dic1= {'id':['1','2','3','4','5'],'value1':['A','C','E','G','I'],'value2':['B','D','F','H','J']} dic2= {'id':['2','3','6','7','8'],'value1':['K','M','O','Q','S'],'value2':['L','N','P','R','T']} dic3= {'id':['1','2','3','4','5','7','8','9','10','11'],'value3':[12,13,14,15,16,17,15,12,13,23]} df1=pd.DataFrame(dic1) df2=pd.DataFrame(dic2) df3=pd.concat([df1,df2]) df4=pd.DataFrame(dic3) df5=pd.merge(df3,df4,on='id') print(df5)

**Output**

id value1 value2 value3 0 1 A B 12 1 2 C D 13 2 2 K L 13 3 3 E F 14 4 3 M N 14 5 4 G H 15 6 5 I J 16 7 7 Q R 17 8 8 S T 15

**Experiment No.: 6**

**Aim**

Demonstrate concat, Merge operations in data frame

**Procedure**

import pandas as pd

s= pd.Series([10,20,30,40])

df=pd.DataFrame(s)

df.columns=['List1']

df['List2']=40

df1=df.drop('List2',axis=1)

df2=df.drop(index=[2,3],axis=0)

print(df)

print(" After deletion::")

print(df1)

print (" After row deletion::")

print(df2)

**Output**

List1 List2

0 10 40

1 20 40

2 30 40

3 40 40

After deletion::

List1

0 10

1 20

2 30

3 40

After row deletion::

List1 List2

0 10 40

1. 20 40

**Experiment No.: 7**

**Aim**

Write a Pandas program to join the two given data frames along rows and assign all data.

**Procedure**

**Test Data:**

student\_data1:

student\_id name marks

0 S1 Danniella Fenton 200

1 S2 Ryder Storey 210

2 S3 Bryce Jensen 190

3 S4 Ed Bernal 222

4 S5 Kwame Morin 199

student\_data2:

student\_id name marks

0 S4 Scarlette Fisher 201

1 S5 Carla Williamson 200

2 S6 Dante Morse 198

3 S7 Kaiser William 219

4 S8 Madeeha Preston 201

**Output**

Original DataFrames:

student\_id name marks

0 S1 Danniella Fenton 200

1 S2 Ryder Storey 210

2 S3 Bryce Jensen 190

3 S4 Ed Bernal 222

4 S5 Kwame Morin 199

-------------------------------------

student\_id name marks

0 S4 Scarlette Fisher 201

1 S5 Carla Williamson 200

2 S6 Dante Morse 198

3 S7 Kaiser William 219

4 S8 Madeeha Preston 201

Join the said two dataframes along rows:

student\_id name marks

0 S1 Danniella Fenton 200

1 S2 Ryder Storey 210

2 S3 Bryce Jensen 190

3 S4 Ed Bernal 222

4 S5 Kwame Morin 199

0 S4 Scarlette Fisher 201

1 S5 Carla Williamson 200

2 S6 Dante Morse 198

3 S7 Kaiser William 219

4 S8 Madeeha Preston 201