Date

Experiment No.

1. Write a python program to do the following operation

a) Create multidimensional arrays and find its shape and dimension.

```
import numpy as np
a = np.array([[1,2,3],[2,3,4],[3,4,5]])
b = a.shape
print ("Shape :", a.shape)
c = a.ndim
print ("dimension:", a.ndim)
```

b) Create a matrix full of zeros and ones

```
# matrix full of zeros
z = np.zeros([2,2])
print ("zeros:", z)

# matrix full of ones
o = np.ones([2,2])
print ("Ones:", o)
```

c) Reshape and flatten data in the array

```
a = np.array([[1,2,3,4],[2,3,4,5],[3,4,5,6],
              [4,5,6,7]])
```

```
b = a.reshape(1,2,2)
print("reshape:", b)

c = a.flatten()
print("flatten:", c)
```

d) Append data vertically and horizontally

```
# Appending data vertically

x = np.array([[10,20],[30,90]])
y = np.array([[30,10],[60,70]])
v = np.vstack((x,y))
print("vertically:", v)

# Appending data horizontally

h = np.hstack((x,y))
print("horizontally:", h)
```

e) Apply indexing and slicing an array

```
# Indexing

a = np.array([[1,2,3,4],[2,3,4,5],[3,4,5,6],[4,5,6,7]])
temp = a[[0,1,2,3],[1,1,1,1]]
print("Indexing:", temp)
```

Date

Experiment No.

```
# slicing

f = a[:1, ::2]
print ("String:", f)
```

f) Use statistical function an array to perform min, max, mean, median and standard deviation.

```
# min for finding minimum of an array

a = np.array([[1,3,-1,4],[3,-2,1,4]])
b = a.min()
print ("minimum:", b)

# max for finding maximum of an array

c = a.max()
print ("maximum:", c)

# mean

d = np.array([[1,2,3,1,5]])
d = d.mean()

# median

e = np.median(d)
print ("median:", e)
```

```
# Standard deviation

f = a.std()
print ("Standard deviation:",f)
```

2. Write a python program to compute.

a) Dot and matrix product of two arrays.

```
import numpy as np
a = np.array([1,2,3])
b = np.array([2,3,4])
print("dot product of one dimension is:" np.dot(a,b))

a = np.array([[1,2],[3,4]])
b = np.array([[1,2],[3,4]])
print("matrix multiplication :", np.matmul(a,b))
print("element multiplication of matrix :" np.multiply
      (a,b))
```

b) Eigen values and Eigen vectors of a matrix.

```
# eigen values of a matrix

import numpy as np
a = np.array([[1,2],[3,4]])
eigvalues, eigvectors = np.linalg.eig(a)
print("eigen values :", eigvalues, "eigen vectors :",
      eigenvector)
```

Date

Experiment No

c) Solve linear equation.

```
import numpy as np
a = np.array([[-2,6],[1,-2]])
b = np.array([[9],[8]])
a_inv = np.linalg.inv(a)
e = np.matmul(a_inv,b)
print("linear equation:",e)
```

d) Multiplicative inverse, rank and determinant of a matrix.

```
# multiplicative inverse

import numpy as np
a = np.array([[-2,6],[1,-2]])
a_inv = np.linalg.inv(a)
print("a. inverse:",a_inv)

# matrix determinant

import numpy as np
a = np.array([[-2,7],[5,-8]])
b = np.linalg.det(a)
print("determinant:",b)

import numpy as np
a = np.array([[-2,3],[6,-7]])
b = np.linalg.matrix_rank(a)
print("rank:",b)
```

Date

Experiment No.

3. Write Python program to

a) Display first ten records.

```
import pandas as pd
dfa_data = pd.read_csv(r"C:\users\ISE 2019\Desktop
                \dm dataset\data1.csv")
records = len(dfa_data)
print("number of records =", records)
dfa_data.head(6)
```

b) Display statistical summary of data frame.

```
import pandas as pd
import numpy as np
a1 = pd.Series([2,3,4])
print(a1.describe())
```

c) Drop a column having NAN.

```
import pandas as pd
import numpy as np
dfa_data = pd.read_csv(r"C:\users\ISE 2019\Desktop
                \dm dataset\data1.csv")
print("before use of dropna()", dfa_data.shape)
dfa_data.dropna(axis=1, how='all', inplace=True)
print("After use of dropna()", dfa_data.shape)
dfa_data.head(5)
```

d) Delete a row containing NAN.

```
import pandas as pd
import numpy as np
dfa_data = pd.read_csv(r"C:\users\ISE 2019\
            Desktop\dm dataset\data1.csv")
print(dfa_data.shape)
dfa_data.dropna(inplace=True)
print(dfa_data.shape)
```

e) To check for missing values of a data set.

```
import pandas as pd
import numpy as np
dict = pd.read_csv(r"C:\users\ISE 2019\Desktop\
        dm dataset\data1.csv")
df = pd.DataFrame(dict)
df = isnull()
```