|  |
| --- |
| **Name: MALAVIKA A**  **Roll No: 16**  **Batch: RMCA B**  **Date: 11-08-2022** |

**DATA SCIENCE LAB**

**Experiment No.: 1**

**Aim**

Matrix Operations

**Procedure**

1. **Identity matrix**

a = [

        [1, 0, 0],

        [0, 1, 0],

        [0, 0, 1]

    ];

flag = True;

rows = len(a);

cols = len(a[0]);

if(rows != cols):

    print("Matrix should be a square matrix");

else:

    for i in range(0, rows):

        for j in range(0, cols):

            if(i == j and a[i][j] != 1):

                flag = False;

                break;

            if(i != j and a[i][j] != 0):

                flag = False;

                break;

    if(flag):

        print("Given matrix is an identity matrix");

    else:

        print("Given matrix is not an identity matrix");

**OUTPUT**



1. **Head, tail**

import pandas as pd

csv\_url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'

col\_names = ['Sepal\_Length','Sepal\_Width','Petal\_Length','Petal\_Width','Class']

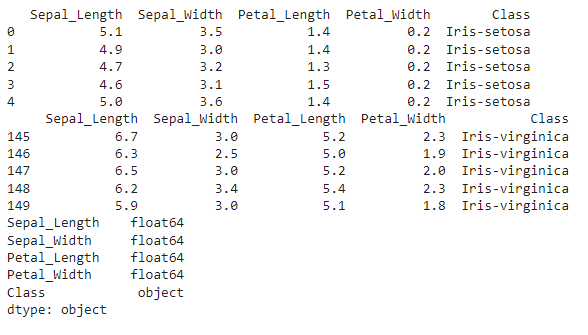
iris =  pd.read\_csv(csv\_url, names = col\_names)

print(iris.head())

print(iris.tail())

print(iris.dtypes)

**OUTPUT**



1. **Bar graph**

from random import seed

from random import randint

from matplotlib import pyplot

seed(1)

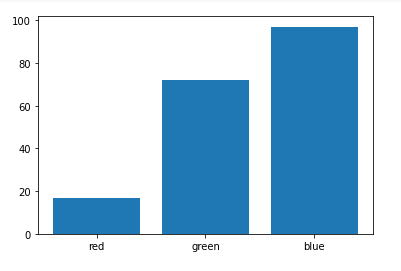
x = ['red', 'green', 'blue']

y = [randint(0, 100), randint(0, 100), randint(0, 100)]

pyplot.bar(x, y)

pyplot.show()

**OUTPUT**



1. **Graph-2**

from numpy import sin

from matplotlib import pyplot

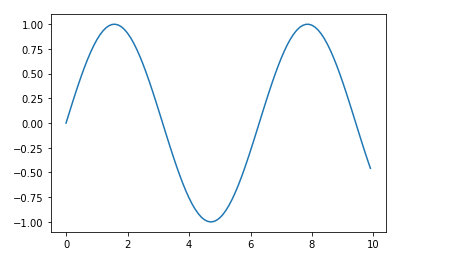
x = [x\*0.1 for x in range(100)]

y = sin(x)

pyplot.plot(x, y)

pyplot.show()

**OUTPUT**



1. **Graph-3**

from numpy.random import seed

from numpy.random import randn

from matplotlib import pyplot

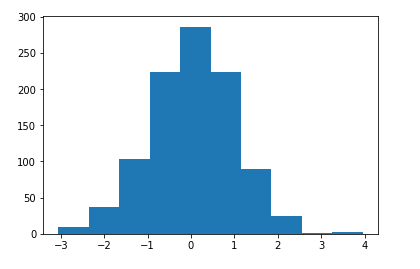
seed(1)

x = randn(1000)

pyplot.hist(x)

pyplot.show()

**OUTPUT**



1. **Dots**

from numpy.random import seed

from numpy.random import randn

from matplotlib import pyplot

seed(1)

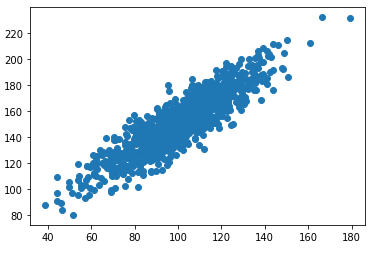
x = 20 \* randn(1000) + 100

y = x + (10 \* randn(1000) + 50)

pyplot.scatter(x, y)

pyplot.show()

**OUTPUT**



1. **Range index**

import pandas as p

index = pd.RangeIndex(start=10, stop=30, step=2, name="data")

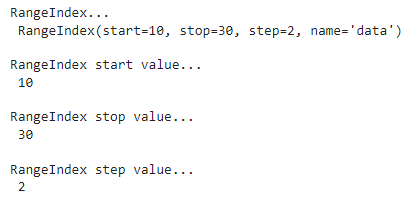
print("RangeIndex...\n",index)

print("\nRangeIndex start value...\n",index.start)

print("\nRangeIndex stop value...\n",index.stop)

print("\nRangeIndex step value...\n",index.step)

**OUTPUT**



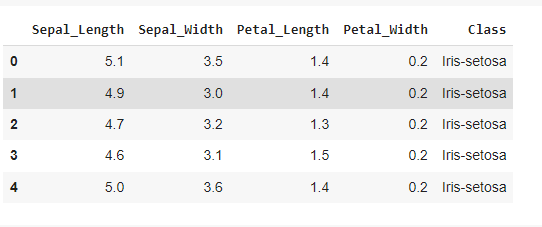
1. **Iris[]**

import pandas as p

iris['Sepal\_Length'].head()

iris[0:5]

**OUTPUT**



1. **Iris.loc**

import pandas as pd

iris.loc[0:10, ['Sepal\_Length', 'Petal\_Length']]

iris.loc[0, ['Sepal\_Length', 'Petal\_Length']]

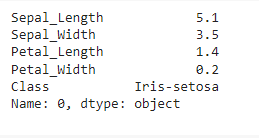
iris.loc[0, 'Petal\_Length']

iris['Petal\_Length'].head()

iris.Sepal\_Length.head()

iris.loc[0]

**OUTPUT**



1. **Iris.iat[]**

import pandas as p

iris.iloc[0:3, 0:4]

iris.iat[0,0]

**OUTPUT**



1. **Matrix add**

X = [[12,7,3],

    [4 ,5,6],

    [7 ,8,9]]

Y = [[5,8,1],

    [6,7,3],

    [4,5,9]]

result = [[0,0,0],

         [0,0,0],

         [0,0,0]]

for i in range(len(X)):

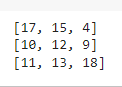
   for j in range(len(X[0])):

       result[i][j] = X[i][j] + Y[i][j]

for r in result:

   print(r)

**OUTPUT**



1. **Matrix multiplication**

X = [[12,7,3],

    [4 ,5,6],

    [7 ,8,9]]

Y = [[5,8,1,2],

    [6,7,3,0],

    [4,5,9,1]]

result = [[0,0,0,0],

         [0,0,0,0],

         [0,0,0,0]]

for i in range(len(X)):

   for j in range(len(Y[0])):

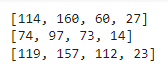
       for k in range(len(Y)):

           result[i][j] += X[i][k] \* Y[k][j]

for r in result:

   print(r)

**OUTPUT**



1. **Determinant matrix**

import numpy as np

arr = np.array([[32, 22, 11],

                [12, 5, 2],

                [21, 32, 7]])

print("Numpy Matrix =")

print(arr)

determinant = np.linalg.det(arr)

print("\nThe Determinant of 3 \* 3 Matrix =")

print(int(determinant))

**OUTPUT**

