# Aim: Demonstrate the use of Numpy for matrix operations

#Addition of matrix

import numpy as np

a = np.array([[2, 4], [5, -6]])

b = np.array([[9, -3], [3, 6]])

c = a + b

print(c)

#Multiplication of matrix

a = np.array([[3, 6, 7], [5, -3, 0]])

b = np.array([[1, 1], [2, 1], [3, -3]])

c = a.dot(b)

print(c)

#Transpose of matrix

a = np.array([[1, 1], [2, 1], [3, -3]])

print(a.transpose())

#Accessing matrix elements

a = np.array([2, 4, 6, 8, 10])

print("a[0] =", a[0])

print("a[2] =", a[2])

print("a[-1] =", a[-1])

a = np.array([[1, 4, 5, 12],

[-5, 8, 9, 0],

[-6, 7, 11, 19]])

print("a[0][0] =", a[0][0])

print("a[1][2] =", a[1][2])

print("a[-1][-1] =", a[-1][-1])

#Accessing rows of the matrix

a = np.array([[1, 4, 5, 12],

[-5, 8, 9, 0],

[-6, 7, 11, 19]])

print("a[0] =", a[0])

print("a[2] =", a[2])

print("a[-1] =", a[-1])

#Accessing columns of the matrix

a = np.array([[1, 4, 5, 12],

[-5, 8, 9, 0],

[-6, 7, 11, 19]])

print("a[:,0] =",a[:,0])

print("a[:,3] =", a[:,3])

print("a[:,-1] =", a[:,-1])

#silicing matrix

a = np.array([1, 3, 5, 7, 9, 7, 5])

print(a[2:5])

print(a[:-5])

print(a[5:])

print(a[:])

print(a[::-1])