EXCERCSE -NUMPY

#Numpy-1

import numpy as np1

import numpy as np

import numpy

arr = numpy.array([1, 2, 3, 4, 5])

print(arr)

print("Version of numpy")

print(np1.\_\_version\_\_)

# Zero Dimension Array

print('Single number print')

a= np1.array(24)

print(a)

# 2 D. Array

print('2 D Array')

arr = np.array([[1, 2, 3], [4, 5, 6]])

print(arr)

print('3 D Array')

arr = np.array([[[1, 2, 3], [4, 5, 6]], [[1, 2, 3], [4, 5, 6]]])

print(arr)

#Check Number of Dimensions?

# NumPy Arrays provides the

# ndim attribute that returns an integer that tells us

#  how many dimensions the array have.

import numpy as np

a = np.array(42)

b = np.array([1, 2, 3, 4, 5])

c = np.array([[1, 2, 3], [4, 5, 6]])

d = np.array([[[1, 2, 3], [4, 5, 6]], [[1, 2, 3], [4, 5, 6]]])

print(a.ndim)

print(b.ndim)

print(c.ndim)

print(d.ndim)

#Create an array with 5 dimensions

# and verify that it has 5 dimensions

arr\_5 = np.array([1, 2, 3, 4], ndmin=5)

print(arr\_5)

print('number of dimensions :', arr\_5.ndim)

#syntax for checking the

# number of dimension of a NumPy array

arr = np.array([1, 2, 3, 4])

print("checking the number of dimension of a NumPy array")

print(arr.ndim)#'Run array'

#Insert the correct syntax for printing

#  the first item in the array.

arr = np.array([1, 2, 3, 4, 5])

print("First Element :",arr[0])

#Insert the correct syntax for printing the number 50

#  from the array.

arr = np.array([10, 20, 30, 40, 50])

print("Print number 50 :",arr[4])

#Insert the correct syntax for printing the number 50

#  from the array.

arr = np.array([[10, 20, 30, 40], [50, 60, 70, 80]])

print("2 D array Traversing:",arr[1,0])

#Use negative index to print the last item in the array.

arr = np.array([10, 20, 30, 40, 50])

print("LAst element with Negative number",arr[-1])

# SLICING of

#Insert the correct slicing syntax to print

#  the following selection of the array:

#Everything from (including) the second item

# to (not including) the fifth item.

arr = np.array([10, 15, 20, 25, 30, 35, 40])

print("Slicing Array Selection",arr[1:4])

#Insert the correct slicing syntax to print the following

#  selection of the array:

#Every other item from (including) the second item

# to (not including) the fifth item.

# Tip: use the step syntax.

arr = np.array([10, 15, 20, 25, 30, 35, 40])

print("Slicing  an Array with step syntax",arr[1:4:2])

#Insert the correct slicing syntax to print the following

# selection of the array:

#Every other item from the entire array.

#Tip: use the step syntax.

arr = np.array([10, 15, 20, 25, 30, 35, 40])

print("Slicing Array",arr[::2])

'''

#NumPy uses a character to represent each of the following data types, which one?

'''

'''

i = integer

b = boolean

u = unsigned integer

f = float

c = complex float

m = timedelta

M = datetime

O = object

S = string

'''

# Insert the correct NumPy syntax to print the data

#  type of an array.

arr = np.array([1, 2, 3, 4])

print(arr.dtype)

#Insert the correct argument to specify that the array should be of type STRING.

arr = np.array([1, 2, 3, 4],dtype='S')

#Insert the correct method to change the data type to integer.

arr = np.array([1.1, 2.1, 3.1])

newarr = arr.astype('i')

print("Display   Array",newarr)

# Excercise : Numpy Copy Vs View

# Use the correct method to make a copy of the array.

arr = np.array([1, 2, 3, 4, 5])

x = arr.copy()

print("Copy Value",x)

# Use the correct method to make a view of the array.

arr = np.array([1, 2, 3, 4, 5])

z = arr.view()

print("View Data",z)

# Numpy Array As a Shape of an array

arr = np.array([1, 2, 3, 4, 5])

print("Display Shape of An Array",arr.shape)

# change of shape

#Use the correct NumPy method

# to change the shape of an array from 1-D to 2-D.

arr = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12])

newarr = arr.reshape(4, 3)

print("Display Shape of old Array",arr)

print("Display Shape of new Array(4 rows\*3 columns)",newarr)

#Use a correct NumPy method to change the shape of an array

#  from 2-D to 1-D.

arr = np.array([[1, 2, 3, 4, 5, 6], [7, 8, 9, 10, 11, 12]])

newarr = arr.reshape(-1)

print("Display Shape of old Array",arr)

print("Display Shape of new Array",newarr)

# Join  Array

#Use a correct NumPy method to join two arrays into a single array.

arr1 = np.array([1, 2, 3])

arr2 = np.array([4, 5, 6])

arr = np.concatenate((arr1, arr2))

print("Display 1 Array",arr1)

print("Display 2 Array",arr2)

print("Display new Array after join",arr)

#Use the correct NumPy method to find all items with the value 4.

arr = np.array([1, 2, 3, 4, 5, 4, 4])

x = np.where(arr == 4)

print("Display location of element ==4",x)

# SORT ELEMENT ARRAY

#Use the correct NumPy method to return a sorted array.

arr = np.array([3, 2, 0, 1])

x = np.sort(arr)

print("Display Array -Before Sorted ORder",arr)

print("Display Array -Sorted ORder",x)