Import NumPy package as np In [3]: import numpy as np Create an empty array of 20 0's and replace the 4th object with the number 5 In [4]: a = np.zeros(20)a[3] = 50., 0., 0.]) Create an array of 20 1's and store it as a variable named array\_master. Copy the same array into another variable named array\_copy In [5]: array\_master = np.ones(20) array\_copy = array\_master.copy() array\_copy Out[5]: 1., 1., 1.]) Create an array containing 30 1's and broadcast all the one's to the value 100 In [6]: array = np.ones(30)array[:] = 100array array([100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., Out[6]: 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100., 100.]) Create an array of integers starting from 21 until 31 and name it as array1 Create an array of integers starting from 11 until 21 and name it array2 Calculate the difference between array1 and array2 In [7]: array1 = np.arange(21,32)array2 = np.arange(11, 22)array1-array2 array([10, 10, 10, 10, 10, 10, 10, 10, 10, 10]) Out[7]: Create an array of all even integers from 2 to 10 and name it a1 Create an array of all even integers from 22 to 30 and name it a2 a) Use the 2 arrays as rows and create a matrix [ Hint - Use stack function from numpy ] In [8]: a1 = np.arange(2, 11, 2)a2 = np.arange(22, 31, 2)np.stack((a1,a2)) array([[ 2, 4, 6, 8, 10], [22, 24, 26, 28, 30]]) b) Use the 2 arrays as columns and create a matrix [ Hint - Use column\_stack function from numpy ] In [9]: np.column\_stack((a1,a2)) array([[ 2, 22], Out[9]: [ 4, 24], [ 6, 26], [ 8, 28], [10, 30]]) Create a 5x6 matrix with values ranging from 0 to 29 and retrieve the value intersecting at 2nd row and 3rd column In [10]: array\_matrix = np.arange(30).reshape(5,6) array\_matrix[1:2] array([[ 6, 7, 8, 9, 10, 11]]) Create an identity matrix of shape 10x10 and replace the 0's with the value 21 In [11]: arr = np.eye(10)arr[arr==0] = 21Use NumPy to generate a random set of 10 numbers between 0 and 1 Display a boolean array output where all values > 0.2 are True, rest are marked as False In [12]: ar = np.random.rand(10)ar>0.2 array([False, True, True, True, True, True, True, True, True, Use NumPy to generate an array matrix of 5x2 random numbers sampled from a standard normal distribution In [13]: np.random.randn(5,2) array([[ 0.07135728, -0.99640898], 0.71415077, 0.27860674], 0.90151723, -0.46561746], 0.144217 , -0.12849378], [ 1.34373202, -1.14202742]]) Create an array of 30 linearly spaced points between 0 and 100 In [15]: np.linspace(0,100,30) array([ 0. 3.44827586, 6.89655172, 10.34482759, 13.79310345, 17.24137931, 20.68965517, 24.13793103, 27.5862069 , 31.03448276, 34.48275862, 37.93103448, 41.37931034, 44.82758621, 48.27586207, 51.72413793, 65.51724138, 55.17241379, 58.62068966, 62.06896552, 68.96551724, 72.4137931 , 75.86206897, 79.31034483, 82.75862069, 86.20689655, 89.65517241, 93.10344828, 96.55172414, 100. ]) Numpy Indexing and Selections Using the below given Matrix, generate the output for the below questions In [16]: simple\_matrix = np.arange(1,101).reshape(10,10) simple\_matrix 2, 3, 4, 5, 6, 7, 8, 9, 10], array([[ 1, Out[16]: 16, 17, 11, 12, 13, 15, 18, 19, 20], 14, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30], 31, 32, 33, 34, 35, 36, 37, 38, 39, 40], 45, 42, 43, 41, 44, 46, 47, 48, 49, 50], 51, 52, 53, 54, 55, 56, 57, 58, 59, 60], 61, 62, 63, 64, 65, 66, 67, 68, 69, 70], [ 71, 72, 73, 74, 75, 76, 77, 78, 79, 80], [ 81, 82, 83, 84, 85, 86, 87, 88, 89, 90], [ 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]]) a) Retrieve the last 2 rows and first 3 column values of the above matrix using index & selection technique In [20]: simple\_matrix[8:,0:3] array([[81, 82, 83], Out[20]: [91, 92, 93]]) b) Retrieve the value 55 from the above matrix using index & selection technique simple\_matrix[5,4] Out[23]: c) Retrieve the values from the 3rd column in the above matrix In [24]: simple\_matrix[:,2] array([ 3, 13, 23, 33, 43, 53, 63, 73, 83, 93]) Out[24]: d) Retrieve the values from the 4th row in the above matrix In [25]: simple\_matrix[3,:] array([31, 32, 33, 34, 35, 36, 37, 38, 39, 40]) Out[25]: e) Retrieve values from the 2nd & 4th rows in the above matrix In [31]:  $simple_matrix[(2,4),:]$ array([[21, 22, 23, 24, 25, 26, 27, 28, 29, 30], Out[31]: [41, 42, 43, 44, 45, 46, 47, 48, 49, 50]]) Calculate the following values for the given matrix a) Calculate sum of all the values in the matrix In [32]: simple\_matrix.sum() 5050 Out[32]: b) Calculate standard deviation of all the values in the matrix In [33]: simple\_matrix.std() 28.86607004772212 Out[33]: c) Calculate the variance of all values in the matrix In [35]: simple\_matrix.var() 833.25 Out[35]: d) Calculate the mean of all values in the matrix In [37]: simple\_matrix.mean() Out[37]: e) Retrieve the largest number from the matrix In [38]: simple\_matrix.max() 100 Out[38]: f) Retrieve the smallest number from the matrix In [39]: simple\_matrix.min() Out[39]: 1