**Numpy Sorting**

Sorting refers to arranging data in a particular format. Sorting algorithm specifies the way to arrange data in a particular order. Most common orders are in numerical or lexicographical order. In Numpy, we can perform various sorting operations using the various functions that are provided in the library like sort, lexsort, argsort etc.

|  |
| --- |
| # importing libraries  import numpy as np    # sort along the first axis  a = np.array([[12, 15], [10, 1]])  arr1 = np.sort(a, axis = 0)  print ("Along first axis : \n", arr1)      # sort along the last axis  a = np.array([[10, 15], [12, 1]])  arr2 = np.sort(a, axis = -1)  print ("\nAlong first axis : \n", arr2)      a = np.array([[12, 15], [10, 1]])  arr1 = np.sort(a, axis = None)  print ("\nAlong none axis : \n", arr1) |

[**numpy.argsort()**](https://www.geeksforgeeks.org/sort-sorteda-np-argsorta-np-lexsortb-python/) **:**This function returns the indices that would sort an array.

|  |
| --- |
| # Python code to demonstrate  # working of  numpy.argsort  import numpy as np    # Numpy array created  a = np.array([9, 3, 1, 7, 4, 3, 6])    # unsorted array print  print('Original array:\n', a)    # Sort array indices  b = np.argsort(a)  print('Sorted indices of original array->', b)    # To get sorted array using sorted indices  # c is temp array created of same len as of b  c = np.zeros(len(b), dtype = int)  for i in range(0, len(b)):      c[i]= a[b[i]]  print('Sorted array->', c) |

[**numpy.lexsort()**](https://www.geeksforgeeks.org/sort-sorteda-np-argsorta-np-lexsortb-python/) **:** This function returns an indirect stable sort using a sequence of keys.

|  |
| --- |
| # Python code to demonstrate working of import numpy as np    # Numpy array created  # First column  a = np.array([9, 3, 1, 3, 4, 3, 6])    # Second column  b = np.array([4, 6, 9, 2, 1, 8, 7])  print('column a, column b')  for (i, j) in zip(a, b):      print(i, ' ', j)    # Sort by a then by b  ind = np.lexsort((b, a))  print('Sorted indices->', ind) |