**Collection Framework Practice Exercises**

1. How would you convert an ArrayList to Array and an Array to ArrayList?
2. Write a Java program to create a new array list, add some colors (string) and print out the collection.
3. How will you reverse a List?
4. Write a Java program to iterate through all elements in an array list.
5. Write a Java program to retrieve an element (at a specified index) from a given array list.
6. Write a Java program to remove the third element from an array list.
7. Write a Java program to search an element in an array list.
8. Write a Java program to sort a given array list.
9. Write a Java program to extract a portion of an array list.
10. Write a Java program to compare two array lists.
11. Write a Java program to join two array lists.
12. Write a Java program to empty an array list.
13. Write a Java program to trim the capacity of an array list the current list size.
14. What is the difference between arraylist and linkedlist?
15. Write a Java program to append the specified element to the end of a linked list.
16. Write a Java program to iterate through all elements in a linked list.
17. Write a Java program to get the first and last occurrence of the specified elements in a linked list.
18. Write a Java program of swap two elements in a linked list
19. Write a Java program to insert the specified element at the specified position in the linked list.
20. Write a Java program to insert some elements at the specified position into a linked list.
21. Write a Java program to get the first and last occurrence of the specified elements in a linked list.
22. Write a Java program to display the elements and their positions in a linked list.
23. Write a Java program to remove a specified element from a linked list.
24. Write a Java program to shuffle the elements in a linked list.
25. Write a Java program to join two linked lists.
26. Write a Java program to clone an linked list to another linked list.
27. Write a Java program to remove and return the first element of a linked list.
28. Write a Java program to retrieve but does not remove, the first element of a linked list.
29. Write a Java program to retrieve but does not remove, the last element of a linked list.
30. Write a Java program to check if a particular element exists in a linked list.
31. Write a Java program to convert a linked list to array list.
32. Write a Java program to compare two linked lists.
33. Write a Java program to test an linked list is empty or not.
34. Write a Java program to replace an element in a linked list.
35. Write a Java program to append the specified element to the end of a hash set.
36. Write a Java program to iterate through all elements in a hash list
37. Write a Java program to get the number of elements in a hash set.
38. Write a Java program to empty a hash set.
39. Write a Java program to test if a hash set is empty or not.
40. Write a Java program to clone a hash set to another hash set.
41. Write a Java program to convert a hash set to an array.
42. Write a Java program to convert a hash set to a tree set.
43. Write a Java program to convert a hash set to a List/ArrayList.
44. Write a Java program to compare two hash set.
45. Write a Java program to compare two sets and retain elements which are the same on both sets.
46. Write a Java program to remove all of the elements from a hash set.
47. Write a Java program to create a new tree set, add some colors (string) and print out the tree set.
48. Write a Java program to iterate through all elements in a tree set.
49. Write a Java program to add all the elements of a specified tree set to another tree set.
50. Write a Java program to create a reverse order view of the elements contained in a given tree set.
51. Write a Java program to get the first and last elements in a tree set.
52. Write a Java program to get the number of elements in a tree set.
53. Write a Java program to compare two tree sets.
54. Write a Java program to find the numbers less than 7 in a tree set.
55. Write a Java program to get the element in a tree set which is greater than or equal to the given element.
56. Write a Java program to get the element in a tree set which is less than or equal to the given element.
57. Write a Java program to get the element in a tree set which is strictly greater than or equal to the given element.
58. Write a Java program to get an element in a tree set which is strictly less than the given element.
59. Write a Java program to retrieve and remove the first element of a tree set.
60. Write a Java program to retrieve and remove the last element of a tree set.
61. Write a Java program to remove a given element from a tree set.
62. Write a Java program to create a new priority queue, add some colors (string) and print out the elements of the priority queue.
63. Write a Java program to iterate through all elements in priority queue.
64. Write a Java program to add all the elements of a priority queue to another priority queue.
65. Write a Java program to insert a given element into a priority queue.
66. Write a Java program to remove all the elements from a priority queue.
67. Write a Java program to count the number of elements in a priority queue.
68. Write a Java program to compare two priority queues.
69. Write a Java program to retrieve the first element of the priority queue.
70. Write a Java program to retrieve and remove the first element of the priority queue.
71. Write a Java program to convert a priority queue to an array containing all of the elements of the queue.
72. Write a Java program to convert Priority Queue elements to a string representation.
73. Write a Java program to change priorityQueue to maximum priorityqueue.
74. Write a Java program to associate the specified value with the specified key in a HashMap.
75. Write a Java program to count the number of key-value (size) mappings in a map.
76. Write a Java program to copy all of the mappings from the specified map to another map.
77. Write a Java program to remove all of the mappings from a map.
78. Write a Java program to check whether a map contains key-value mappings (empty) or not.d
79. Write a Java program to get a shallow copy of a HashMap instance.
80. Write a Java program to test if a map contains a mapping for the specified key.
81. Write a Java program to test if a map contains a mapping for the specified value.
82. Write a Java program to create a set view of the mappings contained in a map.
83. Write a Java program to get the value of a specified key in a map.
84. Write a Java program to get a set view of the keys contained in this map.
85. Write a Java program to get a collection view of the values contained in this map.
86. Write a Java program to associate the specified value with the specified key in a Tree Map.
87. Write a Java program to copy a Tree Map content to another Tree Map.
88. Write a Java program to search a key in a Tree Map.
89. Write a Java program to search a value in a Tree Map.
90. Write a Java program to get all keys from a given Tree Map.
91. Write a Java program to delete all elements from a given Tree Map.
92. Write a Java program to sort keys in Treemap by using comparator
93. Write a Java program to get a key-value mapping associated with the greatest key and the least key in a map.
94. Write a Java program to get the first (lowest) key and the last (highest) key currently in a map.
95. Write a Java program to get a reverse order view of the keys contained in a given map.
96. Write a Java program to get a key-value mapping associated with the greatest key less than or equal to the given key.
97. Write a Java program to get the greatest key less than or equal to the given key.
98. Write a Java program to get the portion of a map whose keys are strictly less than a given key.
99. Write a Java program to get the portion of this map whose keys are less than (or equal to, if inclusive is true) a given key.
100. Write a Java program to get the least key strictly greater than the given keyReturn null if there is no such key.
101. Write a Java program to get a key-value mapping associated with the greatest key strictly less than the given keyReturn null if there is no such key
102. Write a Java program to get the greatest key strictly less than the given keyReturn null if there is no such key.
103. Write a Java program to get a NavigableSet view of the keys contained in a map.
104. Write a Java program to remove and get a key-value mapping associated with the least key in a map.
105. Write a Java program to remove and get a key-value mapping associated with the greatest key in this map.
106. Write a Java program to get the portion of a map whose keys range from a given key (inclusive), to another key (exclusive).
107. Write a Java program to get the portion of a map whose keys range from a given key to another key.
108. Write a Java program to get a portion of a map whose keys are greater than or equal to a given key.
109. Write a Java program to get a portion of a map whose keys are greater than to a given key.
110. Write a Java program to get a key-value mapping associated with the least key greater than or equal to the given key. Return null if there is no such key.
111. Write a Java program to get the least key greater than or equal to the given key. Returns null if there is no such key.