HashSet Assignments

1. WAP to create a HashSet with Integer objects without using generics
2. WAP to create a HashSet with some colors (String) using generics
3. WAP to create a HashSet from an ArrayList
4. WAP to iterate through all elements in a HashSet and print the elements. Observe the order of elements.
5. WAP to get the number of elements in a HashSet.
6. WAP to empty a HashSet.
7. WAP to test if a HashSet is empty or not.
8. WAP to convert a HashSet to an array.
9. WAP to remove all of the elements from a HashSet.
10. WAP to add user defined objects of type Employee in a HashSet. Print the contents in the Set.
11. WAP to add user defined objects of type Employee in a HashSet using duplicate Employee object. Print the contents in the Set.
12. WAP to add user defined objects of type Employee which now implements Comparable interface in a HashSet. Print the contents in the Set. Add duplicate Employee object and print the contents in the Set. Observe the values.
13. WAP to convert a HashSet to a List/ArrayList.
14. WAP to compare the elements in two HashSets.
15. WAP to compare two sets and retain elements which are same on both sets.

LinkedHashSet Assignment

1. Do all above assignments but using LinkedHashSet

TreeSet Assignments

1. WAP to create a TreeSet from a HashSet.
2. WAP to create a new TreeSet, add Strings and print the TreeSet .
3. WAP to iterate through all elements in a TreeSet.
4. WAP to get the first and last elements in a TreeSet.
5. WAP to get the number of elements in a TreeSet.
6. WAP to create a reverse order view of the elements contained in a given TreeSet.
7. WAP to remove a given element from a TreeSet.
8. WAP to retrieve and remove the lowest element of a TreeSet using a single method call. Repeat the same using 2 different method calls.
9. WAP to retrieve and remove the highest element of a TreeSet using a single method call. Repeat the same using 2 different method calls.
10. WAP to get the element in a TreeSet which is greater than or equal to the given element.
11. WAP to get the element in a TreeSet which is less than or equal to the given element.
12. WAP to get the element in a TreeSet which is strictly greater than or equal to the given element.
13. WAP to get an element in a TreeSet which is strictly less than the given element.
14. WAP to compare two TreeSets.
15. WAP to find the numbers less than 7 in a TreeSet which contains Integer objects.
16. WAP to find the numbers greater than 7 in a TreeSet which contains Integer objects.
17. WAP to add all the elements of a specified TreeSet to another TreeSet.
18. WAP to add user defined objects of type Employee in a TreeSet. Print the contents in the Set.
19. WAP to add user defined objects of type Employee in a TreeSet using duplicate Employee object. Print the contents in the Set.
20. WAP to add user defined objects of type Employee which now implements Comparable interface in a TreeSet. Print the contents in the Set. Add duplicate Employee object and print the contents in the Set. Observe the values.
21. WAP to create a TreeSet and add user defined objects of type Employee (Employee is implementing Comparable interface in natural order). But use a Comparator which sorts in reverse order. Print the contents and observe the order of elements.