1. Write a Java program to append the specified element to the end of a linked list.
2. Write a Java program to iterate through all elements in a linked list.
3. Write a Java program to iterate through all elements in a linked list starting at the specified position.
4. Write a Java program to iterate a linked list in reverse order.
5. Write a Java program to insert the specified element at the specified position in the linked list.
6. Write a Java program to insert elements into the linked list at the first and last positions.
7. Write a Java program to insert the specified element at the front of a linked list.
8. Write a Java program to insert the specified element at the end of a linked list.
9. Write a Java program to insert some elements at the specified position into a linked list.
10. Write a Java program to get the first and last occurrence of the specified elements in a linked list.
11. Write a Java program to display elements and their positions in a linked list.
12. Write a Java program to remove a specified element from a linked list.
13. Write a Java program to remove the first and last elements from a linked list.
14. Write a Java program to remove all elements from a linked list.

Hashset

1.Write a Java program to append the specified element to the end of a hash set.
2. Write a Java program to iterate through all elements in a hash list.
3. Write a Java program to get the number of elements in a hash set.
4. Write a Java program to empty an hash set.
5. Write a Java program to test if a hash set is empty or not.
6.Write a Java program to create a tree set, add some colors (strings) and print out the tree set.

6 Write a Java program to iterate through all elements in a tree set.

7.Write a Java program to add all the elements of a specified tree set to another tree set.
8.Write a Java program to create a reverse order view of the elements contained in a given tree set.
9.Write a Java program to get the first and last elements in a tree set.

Java Collection: HashMap

1. Write a Java program to associate the specified value with the specified key in a HashMap.
2. Write a Java program to count the number of key-value (size) mappings in a map.
3. Write a Java program to copy all mappings from the specified map to another map.
4. Write a Java program to remove all mappings from a map.
5. Write a Java program to check whether a map contains key-value mappings (empty) or not.

Java Collection: TreeMap Exercises [26 exercises with solution]

1. Write a Java program to associate the specified value with the specified key in a Tree Map.
2. Write a Java program to copy Tree Map's content to another Tree Map.
3. Write a Java program to search for a key in a Tree Map.
4. Write a Java program to search for a value in a Tree Map.
5. Write a Java program to get all keys from a Tree Map.