```
//Methods of Collection interface:
//add(), addAll(), clear(), contains(), isEmpty(), iterator(), remove(), removeAll(),
toArray()


// import statements are mandatory
import java.util.ArrayList;
import java.util.Iterator;

public class A2
{
   public static void main(String[] args)
   {
      // Object creation using Any of the class of collection framework

      ArrayList<Integer> al1 = new ArrayList<Integer>();
      ArrayList<Integer> al2 = new ArrayList<>();
      ArrayList<Integer> al3 = new ArrayList();

   // 1. public boolean add(E e) method: It is used to insert an element
      al1.add(10);
      al1.add(20);
      al1.add(30);
      al1.add(40);
      al1.add(50);
      System.out.println(al1);//[10, 20, 30, 40, 50]
      al2.add(1);
      al2.add(2);
      al2.add(3);
      al2.add(4);
      al2.add(5);
      System.out.println(al2);//[1, 2, 3, 4, 5]
      al3.add(10);
      al3.add(30);
      al3.add(40);
      al3.add(50);
      al3.add(60);
      System.out.println(al3);//[10, 30, 40, 50, 60]

 // 2. public boolean addAll(Collection<? extends E> c) method:
//  It is used to insert the specified collection elements in the invoking collection.

      al1.addAll(al3);
      System.out.println(al1);//[10, 20, 30, 40, 50, 10, 30, 40, 50, 60]

      al1.addAll(2,al2);
      System.out.println(al1);//[10, 20, 1, 2, 3, 4, 5, 30, 40, 50, 10, 30, 40, 50, 60]
 // 3. public void clear(): It removes the total number of elements from the collection.

      System.out.println("Use of clear() method: ");
      al2.clear();
      System.out.println(al2);//[]

 // 4. public boolean contains(Object element)
 //It is used to search an element inside collection.

      System.out.println("Use of contains() method: ");
      System.out.println(al1.contains(10)); //true
      System.out.println(al1.contains(100));//false

// 5. public boolean isEmpty(): It checks if collection is empty.

      System.out.println("Use of isEmpty() method: ");
      System.out.println(al1.isEmpty());//false
      System.out.println(al2.isEmpty());//true

// 6. public int remove(int indexposition): It is used to delete an element from the collection.
//  In argument pass index position that you want to remove.

      System.out.println("Use of remove() method: ");
      System.out.println(al1);//[10, 20, 1, 2, 3, 4, 5, 30, 40, 50, 10, 30, 40, 50, 60]
      System.out.println(al1.remove(2));//1
      System.out.println(al1);//[10, 20, 2, 3, 4, 5, 30, 40, 50, 10, 30, 40, 50, 60]
      System.out.println(al1.remove(6));//30
      System.out.println(al1);//[10, 20, 2, 3, 4, 5, 40, 50, 10, 30, 40, 50, 60]
      System.out.println(al3);//[10, 30, 40, 50, 60]

      //System.out.println(al1.remove(13));
     //R.E: IndexOutOfBoundsException, as entered position is not available in ArrayList

 //7. public boolean removeAll(Collection<?> c):
//It is used to delete all the elements of the specified collection from the invoking collection.

      System.out.println(al1.removeAll(al3));//true
      System.out.println(al1);//[20, 2, 3, 4, 5]

 // 8. public Object[] toArray(): It converts collection into array.
      System.out.println("Use of toArray() method: ");

      Object[] obj = al1.toArray();
      for (Object x : obj)
      {
         System.out.println(x);
      }
```

```
    /*Use of toArray() method:
    20
    2
    3
    4
    5*/

  // 9. public Iterator iterator() - It returns an iterator.

    System.out.println("Use of iterator() method: ");
    Iterator iterator = al1.iterator();
    while(iterator.hasNext()){
       System.out.println(iterator.next());
    }
    /*Use of iterator() method:
    20
    2
    3
    4
    5*/
  }
}
```