LIST

```
import java.util.ArrayList;
import java.util.List;

public class ListOperationsDemo {
    public static void main(String[] args) {
        // Creating a new ArrayList of String type
        List<String> myList = new ArrayList<>();

        // Adding elements to the list
        myList.add("Hello"); // Adds to the end
        myList.add("World"); // Adds to the end
        myList.add(1, "Java"); // Inserts "Java" at index 1

        // Printing all elements in the list
        System.out.println("List after adding elements:");
        printList(myList);

        // Removing elements from the list
        myList.remove("World"); // Removes "World"
        myList.remove(0); // Removes the first item (now "Hello")
        System.out.println("List after removing elements:");
        printList(myList);

        // Replacing an element
        myList.set(0, "New Element"); // Replaces "Java" with "New Element"
        System.out.println("List after replacing an element:");
```

```
        printList(myList);

        // Checking if list contains an element
        boolean containsElement = myList.contains("New Element");
        System.out.println("List contains 'New Element': " + containsElement);

        // Getting the size of the list
        int size = myList.size();
        System.out.println("List size: " + size);

        // Checking if the list is empty
        boolean isEmpty = myList.isEmpty();
        System.out.println("List is empty: " + isEmpty);

        // Clearing the list
        myList.clear();
        System.out.println("List after clearing:");
        printList(myList);
    }

    // Utility method to print all elements in the list
    private static void printList(List<String> list) {
        if (list.isEmpty()) {
            System.out.println("The list is empty.");
        } else {
            for (String s : list) {
                System.out.println(s);
            }
        }
```

```
    }
}
```

Output

```
List after adding elements:
Hello
Java
World

List after removing elements:
Java

List after replacing an element:
New Element

List contains 'New Element': True
List size: 1
List is empty: False

List after clearing:
The list is empty.
```

Arraylist,LinkedList(Using a List interface)

```
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;

public class ListInterfaceDemo {
    public static void main(String[] args) {
```

```
    // Using List with ArrayList
    List<String> arrayList = new ArrayList<>();
    arrayList.add("Apple");
    arrayList.add("Banana");
    arrayList.add("Cherry");

    System.out.println("ArrayList Elements:");
    printList(arrayList);

    // Using List with LinkedList
    List<String> linkedList = new LinkedList<>();
    linkedList.add("Ford");
    linkedList.add("Chevrolet");
    linkedList.add("Toyota");

    System.out.println("LinkedList Elements:");
    printList(linkedList);

    // Demonstrating adding and removing elements
    arrayList.add(1, "Blueberry"); // Insert at index 1
    linkedList.remove("Chevrolet"); // Remove element

    System.out.println("Updated ArrayList Elements:");
    printList(arrayList);

    System.out.println("Updated LinkedList Elements:");
    printList(linkedList);
}
```

```
    // Utility method to print all elements in the list
    private static void printList(List<String> list) {
        for (String item : list) {
            System.out.println(item);
        }
    }
}
```

Output

ArrayList Elements:

Apple

Banana

Cherry

LinkedList Elements:

Ford

Chevrolet

Toyota

Updated ArrayList Elements:

Apple

Blueberry

Banana

Cherry

Updated LinkedList Elements:

Ford

Toyota