

**1)Min and Max in a List in Java**

```
import java.util.Arrays;

import java.util.Collections;

import java.util.List;

public class MinMax {

    public static void main(String[] args) {

        List<Integer> numbers = Arrays.asList(4, 2, 9, 1, 7);

        int minValue = Collections.min(numbers);

        int maxValue = Collections.max(numbers);

        System.out.println("Minimum Value: " + minValue);

        System.out.println("Maximum Value: " + maxValue);

    }

}
```

**Output**

Minimum Value: 1

Maximum Value: 9

**2)Split a List into Two Halves in Java**

```
import java.util.Arrays;

import java.util.List;

public class SplitList {

    public static void main(String[] args) {

        List<String> items = Arrays.asList("one", "two", "three", "four", "five", "six");

        int midpoint = items.size() / 2;
```

```

List<String> firstHalf = items.subList(0, midpoint);

List<String> secondHalf = items.subList(midpoint, items.size());

System.out.println("First Half: " + firstHalf);

System.out.println("Second Half: " + secondHalf);

}

}

```

### **Output**

First Half: [one, two, three]

Second Half: [four, five, six]

### **3)Remove Duplicates from ArrayList in Java**

```

import java.util.ArrayList;

import java.util.Arrays;

import java.util.LinkedHashSet;

import java.util.List;

import java.util.Set;

public class RemoveDuplicates {

    public static void main(String[] args) {

        List<String> names = new ArrayList<>();

        names.put("Gayathri");

        names .put("gayu");

        names.put("gaya3");

        names.put("gayu");

        Set<String> uniqueNames = new LinkedHashSet<>(names); names.clear();

        names.addAll(uniqueNames);

        System.out.println( names);
    }
}

```

```
}
```

```
}
```

### **Output**

[Gayathri,gayu,gaya3]

### **4)Add Element at First and Last Position of LinkedList in Java**

```
import java.util.LinkedList;
```

```
public class LinkedList {
```

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

```
LinkedList<Integer> linkedList = new LinkedList<>();
```

```
linkedList.addFirst(1);
```

```
linkedList.addLast(5);
```

```
System.out.println("Linked List: " + linkedList);
```

```
}
```

```
}
```

### **Output**

Linked List: [1, 5]