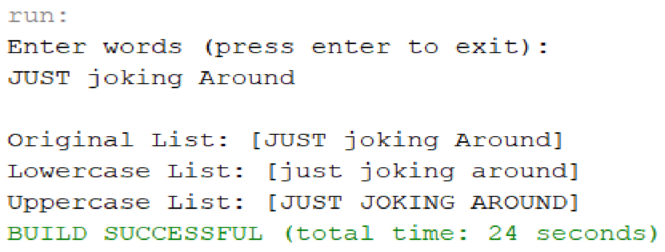
WEEK 8 STREAMS:

Convert a list of strings to uppercase or lowercase using array streams:

CODE:

```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import java.util.stream.Collectors;

public class StringCaseConverter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<String> words = new ArrayList<>();

        System.out.println("Enter words (press enter to exit):");

        while (true) {
            String input = scanner.nextLine();
            if (input.equalsIgnoreCase("")) {
                break;
            }
        List<String> lowercaseWords = words.stream()//list string
                .map(String::toLowerCase)
                .collect(Collectors.toList());

        List<String> uppercaseWords = words.stream()
                .map(String::toUpperCase)
                .collect(Collectors.toList());

        System.out.println("Original List: " + words);
        System.out.println("Lowercase List: " + lowercaseWords);
        System.out.println("Uppercase List: " + uppercaseWords);
       scanner.close();
    }
}
```

OUTPUT

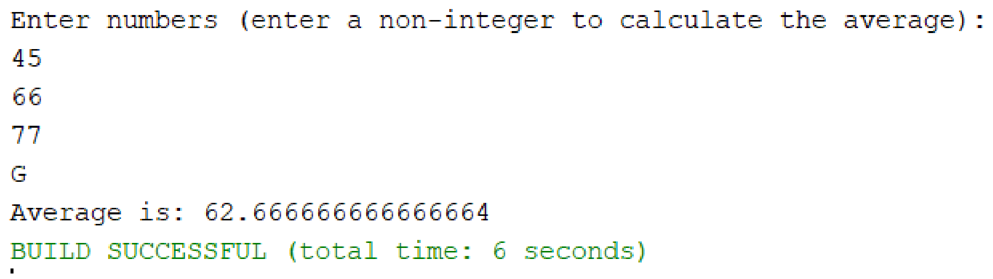


```
run:
Enter words (press enter to exit):
JUST joking Around

Original List: [JUST joking Around]
Lowercase List: [just joking around]
Uppercase List: [JUST JOKING AROUND]
BUILD SUCCESSFUL (total time: 24 seconds)
```

Write a Java program about to calculate average of numbers:

CODE:

```
package Appletlab;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class AvgOfNumbers {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Integer> numbers = new ArrayList<>();

        System.out.println("Enter numbers (enter a non-integer to calculate the average):");
        while (scanner.hasNextInt()) {
            int number = scanner.nextInt();
            numbers.add(number);
        }

        if (numbers.isEmpty()) {
            System.out.println("No valid numbers entered.");
        } else {
            double avg =
numbers.stream().mapToDouble(Integer::intValue).average().getAsDouble();//--sequential
            System.out.println("Average is: " + avg);
        }
    }
}
```



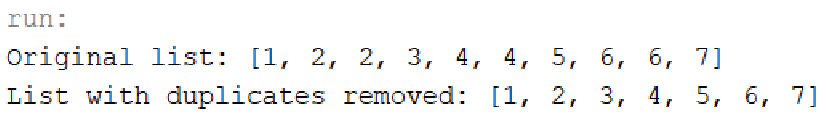
```
Enter numbers (enter a non-integer to calculate the average):
45
66
77
G
Average is: 62.666666666666664
BUILD SUCCESSFUL (total time: 6 seconds)
```

Write a Java program to remove all duplicate elements from a list using stream builder.

CODE:

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
import java.util.stream.Stream;

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

        List<Integer> numbers = new ArrayList<>(Arrays.asList(1, 2, 2, 3, 4, 4, 5, 6, 6, 7));

        StreamBuilder<Integer> streamBuilder = new StreamBuilder<>(numbers);//--

        List<Integer> uniqueNumbers = new ArrayList<>();
        while (streamBuilder.hasNext()) {
            Integer element = streamBuilder.next();
            if (!uniqueNumbers.contains(element)) {
                uniqueNumbers.add(element);
            }
        }

        System.out.println("Original list: " + numbers);
        System.out.println("List with duplicates removed: " + uniqueNumbers);
    }
        StreamBuilder(List<T> source) {
            this.source = source;
        }

        boolean hasNext() {
            return index < source.size();
        }

        T next() {
            return source.get(index++);
        }
    }
}
```

OUTPUT



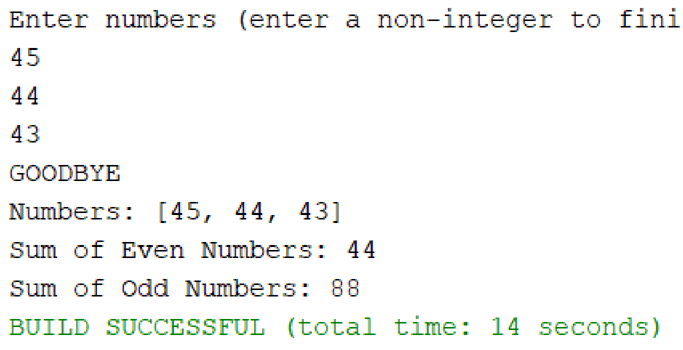
```
run:
Original list: [1, 2, 2, 3, 4, 4, 5, 6, 6, 7]
List with duplicates removed: [1, 2, 3, 4, 5, 6, 7]
```

Write a Java program to calculate the sum of all even, odd numbers in a list using specified streams :

CODE:

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

public class SumEvenOddNumbers {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Integer> numbers = new ArrayList<>();

        System.out.println("Enter numbers (enter a non-integer to finish):");
        while (scanner.hasNextInt()) {
            int number = scanner.nextInt();
            numbers.add(number);
        }
        int sumEven = numbers.stream().filter(n -> n % 2 ==
0).mapToInt(Integer::intValue).sum();

        int sumOdd = numbers.stream().filter(n -> n % 2 !=
0).mapToInt(Integer::intValue).sum();

        // Display the results
        System.out.println("Numbers: " + numbers);
        System.out.println("Sum of Even Numbers: " + sumEven);
        System.out.println("Sum of Odd Numbers: " + sumOdd);

        scanner.close();
    }
}
```

OUTPUT



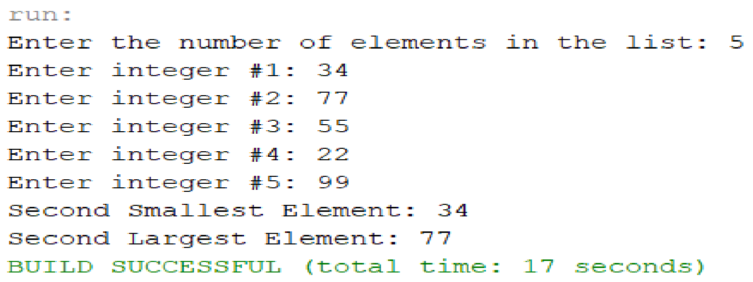
```
Enter numbers (enter a non-integer to finish):
45
44
43
GOODBYE
Numbers: [45, 44, 43]
Sum of Even Numbers: 44
Sum of Odd Numbers: 88
BUILD SUCCESSFUL (total time: 14 seconds)
```

Write a Java program to find the second smallest and largest elements in a list of integers using iterator streams :

CODE:

```
import java.util.*;
import java.util.stream.Collectors;

public class SecondSmallestLargest {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of elements in the list: ");
        int n = scanner.nextInt();

        List<Integer> numbers = new ArrayList<>();
        for (int i = 0; i < n; i++) {
            System.out.print("Enter integer #" + (i + 1) + ": ");
            int num = scanner.nextInt();
            numbers.add(num);
        }
        List<Integer> distinctSortedNumbers = numbers.stream()
                .distinct() // Remove duplicates
                .sorted()   // Sort in ascending order
                .collect(Collectors.toList());

        if (distinctSortedNumbers.size() >= 2) {
            int secondSmallest = distinctSortedNumbers.get(1);
            int secondLargest = distinctSortedNumbers.get(distinctSortedNumbers.size() - 2);
            System.out.println("Second Smallest Element: " + secondSmallest);
            System.out.println("Second Largest Element: " + secondLargest);
        } else {
            System.out.println("Not enough distinct elements to find the second smallest and
largest.");
        }

        scanner.close();
    }
}
```

OUTPUT



```
run:
Enter the number of elements in the list: 5
Enter integer #1: 34
Enter integer #2: 77
Enter integer #3: 55
Enter integer #4: 22
Enter integer #5: 99
Second Smallest Element: 34
Second Largest Element: 77
BUILD SUCCESSFUL (total time: 17 seconds)
```

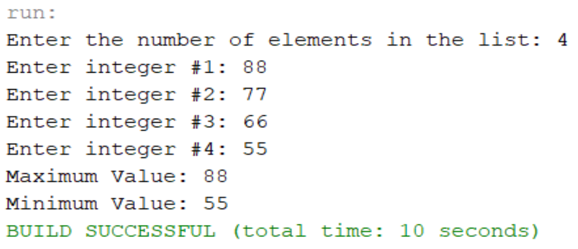
TASK -2

1. **Write a Java program to find the maximum and minimum values in a list of integers using iterable streams:**

CODE:

```
import java.util.*;
import java.util.stream.Stream;

public class MaxMinValues {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
       System.out.print("Enter the number of elements in the list: ");
        int n = scanner.nextInt();

        if (n <= 0) {
            System.out.println("Please enter a positive number of integers.");
            return;
        }

        // Input: Enter the list of integers
        List<Integer> numbers = new ArrayList<>();
        for (int i = 0; i < n; i++) {
            System.out.print("Enter integer #" + (i + 1) + ": ");
            int num = scanner.nextInt();
            numbers.add(num);
        }

        if (max.isPresent() && min.isPresent()) {
            System.out.println("Maximum Value: " + max.get());
            System.out.println("Minimum Value: " + min.get());
        } else {
            System.out.println("No integers provided to find the maximum and minimum.");
        }

        scanner.close();
    }
}
```

OUTPUT



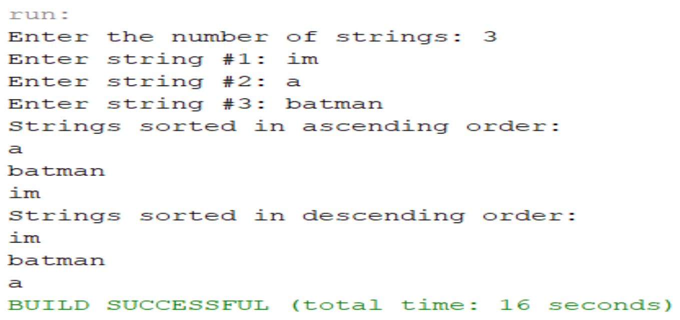
```
run:
Enter the number of elements in the list: 4
Enter integer #1: 88
Enter integer #2: 77
Enter integer #3: 66
Enter integer #4: 55
Maximum Value: 88
Minimum Value: 55
BUILD SUCCESSFUL (total time: 10 seconds)
```

2. Write a Java program to sort a list of strings in alphabetical order, ascending and descending using iterate streams:

CODE:

```
import java.util.*;
import java.util.stream.Collectors;
import java.util.stream.Stream;
public class StringSorter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of strings: ");
        int n = scanner.nextInt();
        scanner.nextLine(); // Consume newline
        List<String> strings = new ArrayList<>();
        for (int i = 0; i < n; i++) {
            System.out.print("Enter string #" + (i + 1) + ": ");
            String inputString = scanner.nextLine();
            strings.add(inputString);
        }
      .collect(Collectors.toList());
     System.out.println("Strings sorted in ascending order:");
        printList(ascendingSortedStrings);

        System.out.println("Strings sorted in descending order:");
        printList(descendingSortedStrings);
       scanner.close();  }
    private static void printList(List<String> list) {
        for (String str : list) {
            System.out.println(str);  }    }}
```

OUTPUT:



```
run:
Enter the number of strings: 3
Enter string #1: im
Enter string #2: a
Enter string #3: batman
Strings sorted in ascending order:
a
batman
im
Strings sorted in descending order:
im
batman
a
BUILD SUCCESSFUL (total time: 16 seconds)
```