Question 1: Write a Java 8 program to filter even numbers from a list ?

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class EvenNumbersFilter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);

        List<Integer> evenNumbers = filterEvenNumbers(numbers);

        System.out.println("Even numbers: " + evenNumbers);
    }

    public static List<Integer> filterEvenNumbers(List<Integer>
numbers) {
        return numbers.stream()
                      .filter(n -> n % 2 == 0)
                      .collect(Collectors.toList());
    }
}
```

Output :- Even numbers: [2, 4]

Question 2: Write a Java 8 program to calculate the sum of integers in a list ?

```
import java.util.Arrays;
import java.util.List;

import java.util.Arrays;
import java.util.List;

public class SumCalculator {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);
        int sum = calculateSum(numbers);
        System.out.println("Sum: " + sum);
    }

    public static int calculateSum(List<Integer> numbers) {
        return numbers.stream()
                      .reduce(0, Integer::sum);
    }
}
```

Output :- Sum: 15

Question 3: Write a Java 8 program to find the maximum element from a list ?

```
import java.util.List;
import java.util.Optional;
```

```
public class MaxNumberFinder {
    public static int findMaxNumber(List<Integer> numbers) {
        Optional<Integer> max = numbers.stream()
                                       .max(Integer::compareTo);
        return max.orElse(Integer.MIN_VALUE);
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(3, 7, 2, 9, 5);
        int maxNumber = findMaxNumber(numbers);
        System.out.println("Maximum number: " + maxNumber);
    }
}
```

Output :- Maximum number: 9

Question 4: Write a Java 8 program to check if a list contains a specific element ?

```
import java.util.Arrays;
import java.util.List;

import java.util.Arrays;
import java.util.List;

public class NumberChecker {
   public static void main(String[] args) {
      List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);
      boolean containsThree = checkIfContainsNumber(numbers, 3);
      System.out.println("Contains 3: " + containsThree);
   }

   public static boolean checkIfContainsNumber(List<Integer> numbers, int targetNumber) {
      return numbers.stream()
                .anyMatch(n -> n == targetNumber);
   }
}
```

Output :- Contains 3: true

Question 5: Write a Java 8 program to find the sum of all even numbers in a list of integers ?

```
import java.util.List;

public class EvenNumberSumCalculator {
    public static int calculateEvenNumberSum(List<Integer> numbers)
{
        return numbers.stream()
                      .filter(n -> n % 2 == 0)
                      .mapToInt(n -> n)
                      .sum();
    }

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

```
        List<Integer> numbers = List.of(1, 2, 3, 4, 5, 6, 7, 8, 9,
10);
        int evenSum = calculateEvenNumberSum(numbers);
        System.out.println("Sum of even numbers: " + evenSum);
    }
}
```

Output :- Sum of even numbers: 30

Question 6: Write a Java 8 program to concatenate all strings in a list ?

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

public class StringConcatenator {
    public static String concatenateStrings(List<String> strings) {
        return strings.stream()
                      .collect(Collectors.joining());
    }

    public static void main(String[] args) {
        List<String> strings = List.of("Hello", " ", "World", "!");
        String concatenatedString = concatenateStrings(strings);
        System.out.println("Concatenated string: " +
concatenatedString);
    }
}
```

Output :- Concatenated string: Hello World!

Question 7: Write a Java 8 program to find the average length of strings in a list of strings?

```
import java.util.List;

public class AverageStringLengthCalculator {
    public static double calculateAverageStringLength(List<String>
strings) {
        return strings.stream()
                      .mapToInt(String::length)
                      .average()
                      .orElse(0.0);
    }

    public static void main(String[] args) {
        List<String> strings = List.of("apple", "banana", "orange",
"grape", "kiwi");
        double averageLength =
calculateAverageStringLength(strings);
        System.out.println("Average length of strings: " +
averageLength);
    }
}
```

Output :- Average length of strings: 5.2

Question 8: Write a Java 8 program to count the occurrences of a given character in a list of strings?

```
import java.util.List;

public class CharacterCounter {
   public static long countCharacterOccurrences(List<String> strings, char targetChar) {
      return strings.stream()
                .flatMapToInt(CharSequence::chars)
                .filter(ch -> ch == targetChar)
                .count();
   }

   public static void main(String[] args) {
      List<String> strings = List.of("apple", "banana", "cherry");
      char targetChar = 'a';
      long occurrences = countCharacterOccurrences(strings, targetChar);
      System.out.println("Occurrences of '" + targetChar + "': " + occurrences);
   }
}
```

Output :- Occurrences of 'a': 4

Question 9: Write a Java 8 program to check if all elements in a list are greater than a given value?

```
import java.util.List;

public class AllElementsChecker {
    public static boolean areAllElementsGreaterThan(List<Integer>
numbers, int threshold) {
        return numbers.stream()
                      .allMatch(n -> n > threshold);
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(10, 20, 30, 40, 50);
        int threshold = 25;
        boolean result = areAllElementsGreaterThan(numbers,
threshold);
        System.out.println("Are all elements greater than " +
threshold + "? " + result);
    }
}
```

Output :- Are all elements greater than 25? false

Question 10: Write a Java 8 program to find the factorial of a given number ?

```
import java.util.stream.IntStream;

public class FactorialCalculator {
    public static int calculateFactorial(int n) {
        return IntStream.rangeClosed(1, n)
                        .reduce(1, (a, b) -> a * b);
    }
```

```
    public static void main(String[] args) {
        int number = 5;
        int factorial = calculateFactorial(number);
        System.out.println("Factorial of " + number + ": " +
factorial);
    }
}
```

Output :- Factorial of 5: 120

Question 11: Write a Java 8 program to remove duplicate elements from a list ?

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

public class DuplicateRemover {
    public static List<Integer> removeDuplicates(List<Integer>
numbers) {
        return numbers.stream()
                      .distinct()
                      .collect(Collectors.toList());
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(1, 2, 2, 3, 4, 4, 5);
        List<Integer> uniqueNumbers = removeDuplicates(numbers);
        System.out.println("Unique numbers: " + uniqueNumbers);
    }
}
```

Output :- Unique numbers: [1, 2, 3, 4, 5]

Question 12: Write a Java 8 program to find the longest string in a list of strings ?

```
import java.util.List;
import java.util.Optional;

public class LongestStringFinder {
    public static String findLongestString(List<String> strings) {
        Optional<String> longest = strings.stream()
                                          .max((s1, s2) ->
s1.length() - s2.length());
        return longest.orElse(null);
    }

    public static void main(String[] args) {
        List<String> strings = List.of("apple", "banana", "cherry",
"grapefruit");
        String longestString = findLongestString(strings);
        System.out.println("Longest string: " + longestString);
    }
}
```

Output :- Longest string: grapefruit

Question 13: Write a Java 8 program to convert all strings to uppercase in a list ?

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

public class StringConverter {
    public static List<String> convertToUpperCase(List<String>
strings) {
        return strings.stream()
                      .map(String::toUpperCase)
                      .collect(Collectors.toList());
    }

    public static void main(String[] args) {
        List<String> strings = List.of("apple", "banana", "cherry");
        List<String> uppercaseStrings = convertToUpperCase(strings);
        System.out.println("Uppercase strings: " +
uppercaseStrings);
    }
}
```

Output :- Uppercase strings: [APPLE, BANANA, CHERRY]

Question 14: Write a Java 8 program to sort a list of strings in alphabetical order ?

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

public class StringSorter {
    public static List<String>
sortStringsAlphabetically(List<String> strings) {
        return strings.stream()
                      .sorted()
                      .collect(Collectors.toList());
    }

    public static void main(String[] args) {
        List<String> strings = List.of("banana", "apple", "cherry");
        List<String> sortedStrings =
sortStringsAlphabetically(strings);
        System.out.println("Sorted strings: " + sortedStrings);
    }
}
```

Output :- Sorted strings: [apple, banana, cherry]

Question 15: Write a Java 8 program to calculate the average of even numbers in a list of integers ?

```
import java.util.List;
import java.util.OptionalDouble;

public class EvenNumberAverageCalculator {
```

```
    public static double calculateAverageOfEvenNumbers(List<Integer>
numbers) {
        OptionalDouble average = numbers.stream()
                                        .filter(n -> n % 2 == 0)
                                        .mapToInt(n -> n)
                                        .average();
        return average.orElse(0.0);
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(1, 2, 3, 4, 5, 6, 7, 8, 9,
10);
        double average = calculateAverageOfEvenNumbers(numbers);
        System.out.println("Average of even numbers: " + average);
    }
}
```

Output :- Average of even numbers: 6.0

Question 16: Write a Java 8 program to convert a list of integers to a comma-separated string ?

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

public class IntegerListConverter {
    public static String
convertToCommaSeparatedString(List<Integer> numbers) {
        return numbers.stream()
                      .map(Object::toString)
                      .collect(Collectors.joining(", "));
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(1, 2, 3, 4, 5);
        String commaSeparatedString =
convertToCommaSeparatedString(numbers);
        System.out.println("Comma-separated string: " +
commaSeparatedString);
    }
}
```

Output :- Comma-separated string: 1, 2, 3, 4, 5

Question 17: Write a Java 8 program to find the last element in a list ?

```
import java.util.Arrays;
import java.util.List;
import java.util.Optional;

public class LastElementFinder {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);
        Optional<Integer> lastElement = findLastElement(numbers);
```

```
        if (lastElement.isPresent()) {
            System.out.println("Last element: " +
lastElement.get());
        } else {
            System.out.println("List is empty");
        }
    }

    public static <T> Optional<T> findLastElement(List<T> list) {
        return list.stream()
                   .reduce((first, second) -> second);
    }
}
```

Output :- Last element: 5

Question 18: Write a Java 8 program to find the second smallest element in a list of integers ?

```
import java.util.List;
import java.util.Optional;

public class SecondSmallestFinder {
    public static int findSecondSmallestElement(List<Integer>
numbers) {
        Optional<Integer> secondSmallest = numbers.stream()
                                                  .distinct()
                                                  .sorted()
                                                  .skip(1)
                                                  .findFirst();
        return secondSmallest.orElse(Integer.MIN_VALUE);
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(3, 1, 4, 2, 5);
        int secondSmallest = findSecondSmallestElement(numbers);
        System.out.println("Second smallest element: " +
secondSmallest);
    }
}
```

Output :- Second smallest element: 2

Question 19: Write a Java 8 program to find the frequency of each word in a list of strings?

```
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;

public class WordFrequencyCounter {
    public static Map<String, Long>
countWordFrequency(List<String> words) {
        return words.stream()
```

```
                    .flatMap(line -> List.of(line.split("\
\s+")).stream())
                    .collect(Collectors.groupingBy(String::toL
owerCase, Collectors.counting()));
    }

    public static void main(String[] args) {
        List<String> words = List.of("apple banana apple",
"banana cherry", "apple banana cherry");
        Map<String, Long> frequencyMap =
countWordFrequency(words);
        System.out.println("Word frequency: " + frequencyMap);
    }
}
```

Output :- Word frequency: {banana=3, cherry=2, apple=3}

Question 20: Write a Java 8 program to find the sum of digits of a list of integers?

```
import java.util.List;

public class DigitSumCalculator {
    public static int calculateDigitSum(List<Integer> numbers) {
        return numbers.stream()
                    .mapToInt(n ->
String.valueOf(n).chars().map(Character::getNumericValue).sum())
                    .sum();
    }

    public static void main(String[] args) {
        List<Integer> numbers = List.of(123, 456, 789);
        int digitSum = calculateDigitSum(numbers);
        System.out.println("Sum of digits: " + digitSum);
    }
}
```

Output :- Sum of digits: 45

Question 21: Write a Java 8 program to find the distinct characters in a list of strings ?

```
import java.util.List;
import java.util.Set;
import java.util.stream.Collectors;

public class DistinctCharacterFinder {
    public static Set<Character>
findDistinctCharacters(List<String> strings) {
        return strings.stream()
                    .flatMapToInt(String::chars)
                    .mapToObj(ch -> (char) ch)
                    .collect(Collectors.toSet());
```

```
    }

    public static void main(String[] args) {
        List<String> strings = List.of("apple", "banana",
"cherry");
        Set<Character> distinctChars =
findDistinctCharacters(strings);
        System.out.println("Distinct characters: " +
distinctChars);
    }
}
```

Output :- Distinct characters: [p, a, b, r, c, e, h, y, l, n]

Question 22: Write Java 8 program to find all the numbers starting with 2 in given list ?

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class NumberFilter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(223, 234, 145, 367,
289, 2001, 2289);
        List<Integer> numbersStartingWithOne =
filterNumbersStartingWithOne(numbers);
        System.out.println("Numbers starting with 1: " +
numbersStartingWithOne);
    }

    public static List<Integer>
filterNumbersStartingWithOne(List<Integer> numbers) {
        return numbers.stream()
                      .filter(number ->
String.valueOf(number).startsWith("2"))
                      .collect(Collectors.toList());
    }
}
```

Output :- Numbers starting with 1: [223, 234, 289, 2001, 2289]

Question 23: Write Java 8 program to find the first element of the given integers list ?

```
import java.util.Arrays;
import java.util.List;
import java.util.Optional;

public class FirstElementFinder {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(10, 20, 30, 40, 50);

        Optional<Integer> firstElement = findFirstElement(numbers);
```

```
        if (firstElement.isPresent()) {
            System.out.println("First element of the list: " +
firstElement.get());
        } else {
            System.out.println("The list is empty.");
        }
    }

    public static Optional<Integer> findFirstElement(List<Integer>
numbers) {
        return numbers.stream().findFirst();
    }
}
```

Output :- First element of the list: 10

Question 24: Write Java 8 program to count the total numbers of elements in the given integers list ?

```
import java.util.Arrays;
import java.util.List;

public class ElementCounter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(10, 20, 30, 40, 50);

        long count = countElements(numbers);

        System.out.println("Total number of elements in the list: "
+ count);
    }

    public static long countElements(List<Integer> numbers) {
        return numbers.stream().count();
    }
}
```

Output :- Total number of elements in the list: 5

Question 25 : Write Java 8 program to sort all the values of the list in ascending order?

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class IntegersSorter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(5, 2, 8, 1, 9, 3);

        List<Integer> sortedNumbers = sortIntegers(numbers);

        System.out.println("Sorted numbers: " + sortedNumbers);
    }
```

```
    public static List<Integer> sortIntegers(List<Integer> numbers)
{
        return numbers.stream()
                      .sorted()
                      .collect(Collectors.toList());
    }
}
```

Output :- Sorted numbers: [1, 2, 3, 5, 8, 9]

Question 26 : Write Java 8 program to sort all the values of the list in descending order?

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class IntegersSorter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(5, 2, 8, 1, 9, 3);

        List<Integer> sortedNumbers =
sortIntegersDescending(numbers);

        System.out.println("Sorted numbers in descending order: " +
sortedNumbers);
    }

    public static List<Integer> sortIntegersDescending(List<Integer>
numbers) {
        return numbers.stream()
                      .sorted(java.util.Comparator.reverseOrder()) /
/ Sort in descending order
                      .collect(Collectors.toList());
    }
}
```

Output :- Sorted numbers in descending order: [9, 8, 5, 3, 2, 1]

Question 27 : Write Java 8 program to check if given integer array contains duplicate or not. Return true if it contains duplicate character.

```
import java.util.Arrays;

public class DuplicateChecker {
    public static void main(String[] args) {
        int[] nums = {1, 2, 3, 4, 5};
        System.out.println("Contains duplicates: " +
containsDuplicates(nums));
    }

    public static boolean containsDuplicates(int[] nums) {
        return Arrays.stream(nums)
                     .boxed() // Convert int[] to Stream<Integer>
                     .distinct() // Remove duplicates
                     .count() != nums.length; // Check if count is
different from array length
```

```
    }
}
```

Output :- Contains duplicates: false

Question 28 : Write a Java 8 program to concatenate two Streams?

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

public class StreamConcatenator {
    public static void main(String[] args) {
        Stream<Integer> stream1 = Stream.of(1, 2, 3);
        Stream<Integer> stream2 = Stream.of(4, 5, 6);

        Stream<Integer> concatenatedStream =
concatenateStreams(stream1, stream2);

        concatenatedStream.forEach(System.out::print);
    }

    public static <T> Stream<T> concatenateStreams(Stream<T>
stream1, Stream<T> stream2) {
        return Stream.concat(stream1, stream2);
    }
}
```

Output :- 123456

Question 29 : Write Java 8 program to perform square on list elements and filter numbers greater than 1000.

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class SquareAndFilter {
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class Main {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(10, 20, 30, 40, 50,
110, 120);

        List<Integer> squaredAndFilteredNumbers =
squareAndFilterNumbers(numbers);

        System.out.println("Numbers squared and filtered: " +
squaredAndFilteredNumbers);
    }

    public static List<Integer> squareAndFilterNumbers(List<Integer>
numbers) {
        return numbers.stream()
                      .map(n -> n * n) // Square each element
```

```
                        .filter(n -> n > 1000) // Filter numbers
greater than 1000
                        .collect(Collectors.toList());
    }
}
```

Output :- Numbers squared and filtered: [1600, 2500, 12100, 14400]

Question 30 : Write Java 8 program to separate odd and even numbers from the given list of integers?

```
import java.util.Arrays;
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;

public class OddEvenSeparator {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5, 6, 7,
8, 9, 10);

        Map<Boolean, List<Integer>> oddEvenMap =
separateOddEven(numbers);

        System.out.println("Odd numbers: " + oddEvenMap.get(false));
        System.out.println("Even numbers: " + oddEvenMap.get(true));
    }

    public static Map<Boolean, List<Integer>>
separateOddEven(List<Integer> numbers) {
        return numbers.stream()
                        .collect(Collectors.partitioningBy(n -> n % 2
== 0));
    }
}
```

Output :- Odd numbers: [1, 3, 5, 7, 9], Even numbers: [2, 4, 6, 8, 10]

Question 31 : Write Java 8 program to print the numbers which are multiples of 3?

```
import java.util.Arrays;
import java.util.List;

public class MultiplesOfThreePrinter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(3, 6, 9, 20, 25, 40);
        printMultiplesOfThree(numbers);
    }

    public static void printMultiplesOfThree(List<Integer> numbers)
{
        System.out.println("Multiples of 3:");
        numbers.stream()
               .filter(n -> n % 3 == 0) // Filter numbers that are
multiples of 3
```

```
                .forEach(System.out::println); // Print each multiple
of 3
    }
}
```

Output :- 3 6 and 9

Question 32 : Write Java 8 program to merge two unsorted arrays into single sorted array?

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

public class ArrayMerger {
    public static void main(String[] args) {
        int[] arr1 = {3, 6, 8, 10, 10};
        int[] arr2 = {1, 2, 4, 5};

        int[] mergedArray = mergeAndSortArrays(arr1, arr2);

        System.out.println("Merged and sorted array: " +
Arrays.toString(mergedArray));
    }

    public static int[] mergeAndSortArrays(int[] arr1, int[] arr2) {
            return IntStream.concat(Arrays.stream(arr1),
Arrays.stream(arr2))
                .sorted()
                .toArray();
    }
}
```

Output :- Merged sorted array: [1, 2, 3, 4, 5, 6, 8, 10, 10]

Question 33 : Write Java 8 program to merge two unsorted arrays into single sorted array by removing duplicates?

```
import java.util.Arrays;

public class ArrayMerger {
    public static void main(String[] args) {
        int[] arr1 = {4, 2, 6, 1};
        int[] arr2 = {7, 3, 5, 8};

        int[] mergedArray =
mergeAndSortArraysWithoutDuplicates(arr1, arr2);

        System.out.println("Merged and sorted array without
duplicates: " + Arrays.toString(mergedArray));
    }

    public static int[] mergeAndSortArraysWithoutDuplicates(int[]
arr1, int[] arr2) {
        return IntStream.concat(Arrays.stream(arr1),
Arrays.stream(arr2))
// Concatenate both arrays into a single stream
                .distinct() // Remove duplicates
```

```
                .sorted() // Sort the elements of the stream
                .toArray(); // Convert the sorted stream to an array
    }
}
```

Output :- Merged and sorted array without duplicates: [1, 2, 3, 4, 5, 6, 7, 8]

Question 34 : Java program to get first three maximum numbers and three minimum numbers from the given list of integers?

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class MaxMinNumbersFinder {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(10, 5, 20, 15, 25, 3,
30, 1, 8);

        List<Integer> maxNumbers = findMaxNumbers(numbers, 3);
        List<Integer> minNumbers = findMinNumbers(numbers, 3);

        System.out.println("Three maximum numbers: " + maxNumbers);
        System.out.println("Three minimum numbers: " + minNumbers);
    }

    public static List<Integer> findMaxNumbers(List<Integer>
numbers, int count) {
        return numbers.stream()
                .sorted((a, b) -> Integer.compare(b, a)) //
Sort in descending order
                .limit(count) // Limit to count of maximum
numbers
                .collect(Collectors.toList());
    }

    public static List<Integer> findMinNumbers(List<Integer>
numbers, int count) {
        return numbers.stream()
                .sorted() // Sort in ascending order (default)
                .limit(count) // Limit to count of minimum
numbers
                .collect(Collectors.toList());
    }
}
```

Output :- Three maximum numbers: [30, 25, 20], Three minimum numbers: [1, 3, 5]

Question 35 : Java 8 program to check if two strings are anagrams or not?

```
import java.util.Arrays;

public class AnagramChecker {
    public static void main(String[] args) {
        String str1 = "listen";
        String str2 = "silent";
```

```
        boolean areAnagrams = areAnagrams(str1, str2);

        if (areAnagrams) {
            System.out.println("Strings '" + str1 + "' and '" + str2
+ "' are anagrams.");
        } else {
            System.out.println("Strings '" + str1 + "' and '" + str2
+ "' are not anagrams.");
        }
    }

    public static boolean areAnagrams(String str1, String str2) {
        // Convert strings to lowercase to ignore case sensitivity
        str1 = str1.toLowerCase();
        str2 = str2.toLowerCase();

        // Sort the characters of both strings
        String sortedStr1 = sortString(str1);
        String sortedStr2 = sortString(str2);

        // Compare the sorted strings
        return sortedStr1.equals(sortedStr2);
    }

    private static String sortString(String str) {
        // Convert string to char array, sort the array, and then
convert it back to string
        return Arrays.stream(str.split(""))
                     .sorted()
                     .reduce("", (s1, s2) -> s1 + s2);
    }
}
```

Output :- Strings 'listen' and 'silent' are anagrams.

Question 36 : Write Java 8 program to find sum of all digits of a number ?

```
import java.util.Arrays;

public class DigitSumCalculator {
    public static void main(String[] args) {
        int number = 12345;

        int digitSum = calculateDigitSum(number);

        System.out.println("Sum of digits of " + number + " is: " +
digitSum);
    }

    public static int calculateDigitSum(int number) {
        return Arrays.stream(String.valueOf(number).split("")) //
Convert number to string and split into individual digits
                     .mapToInt(Integer::parseInt) // Map each digit
to an integer
                     .sum(); // Sum all the digits
    }
}
```

Output :- Sum of digits of 12345 is: 15

Question 37 : Write Java 8 program to sort given list of strings according to decreasing order of their length?

```
import java.util.Arrays;
import java.util.Comparator;
import java.util.List;

public class StringLengthSorter {
    public static void main(String[] args) {
        List<String> strings = Arrays.asList("banana", "apple",
"orange", "grape", "kiwi");

        List<String> sortedStrings = sortStringsByLength(strings);

        System.out.println("Sorted strings by length (decreasing
order): " + sortedStrings);
    }

    public static List<String> sortStringsByLength(List<String>
strings) {
        return strings.stream()
                      .sorted((s1, s2) ->
Integer.compare(s2.length(), s1.length())) // Sort strings by length
in decreasing order
                      .toList(); // Convert the stream to a list
    }
}
```

Output :- Sorted strings by length (decreasing order): [banana, orange, apple, grape, kiwi]

Question 38 : Write Java 8 program to find common elements between two lists?

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

public class CommonElementsFinder {
    public static void main(String[] args) {
        List<Integer> list1 = new ArrayList<>(List.of(1, 2, 3, 4,
5));
        List<Integer> list2 = new ArrayList<>(List.of(4, 5, 6, 7,
8));

        Set<Integer> commonElements = findCommonElements(list1,
list2);

        System.out.println("Common elements between the two lists: "
+ commonElements);
    }
```

```
    public static <T> Set<T> findCommonElements(List<T> list1,
List<T> list2) {
        // Convert lists to sets to remove duplicates
        Set<T> set1 = new HashSet<>(list1);
        Set<T> set2 = new HashSet<>(list2);

        // Retain only the elements that are present in both sets
        set1.retainAll(set2);

        return set1;
    }
}
```

Output :- Common elements between the two lists: [4, 5]

Question 39 : Write Java 8 program to prints the first 5 odd numbers?

```
import java.util.stream.IntStream;

public class FirstTenOddNumbers {
    public static void main(String[] args) {
        System.out.println("First 10 odd numbers:");
        printFirstTenOddNumbers();
    }

    public static void printFirstTenOddNumbers() {
        IntStream.iterate(1, n -> n + 2) // Start from 1 and
increment by 2 to get odd numbers
                .limit(5) // Limit to the first 5 odd numbers
                .forEach(System.out::println); // Print each odd
number
    }
}
```

Output :- First 5 odd numbers:1 3 5 7 and 9

Question 40 : Write Java 8 program to print the most repeated element in an array?

```
import java.util.Arrays;
import java.util.Map;
import java.util.function.Function;
import java.util.stream.Collectors;

public class MostRepeatedElementFinder {
    public static void main(String[] args) {
        int[] array = {1, 2, 3, 4, 2, 2, 3, 4, 4, 4, 5, 5, 4};

        Map.Entry<Integer, Long> mostRepeatedElementEntry =
findMostRepeatedElement(array);

        System.out.println("Most repeated number: " +
mostRepeatedElementEntry.getKey());
        System.out.println("Number of occurrences: " +
mostRepeatedElementEntry.getValue());
```

```
    }

    public static Map.Entry<Integer, Long>
findMostRepeatedElement(int[] array) {
        return Arrays.stream(array)
                .boxed()
                .collect(Collectors.groupingBy(Function.identity(),
Collectors.counting()))
                .entrySet().stream()
                .max(Map.Entry.comparingByValue())
                .orElseThrow(() -> new
IllegalArgumentException("Array is empty"));
    }
}
```

Output :- Most repeated number: 4, Number of occurrences: 5

Question 41 : Write Java 8 program to print duplicate elements from an array?

```
import java.util.Arrays;
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;

public class DuplicateElementExtractor {
    public static void main(String[] args) {
        Integer[] array = {1, 2, 3, 4, 2, 3, 5, 6, 7, 8, 9, 1};

        List<Integer> duplicates = findDuplicateElements(array);

        System.out.println("Duplicate elements: " + duplicates);
    }

    public static List<Integer> findDuplicateElements(Integer[]
array) {
        Map<Integer, Long> countMap = Arrays.stream(array)
                .collect(Collectors.groupingBy(e -> e,
Collectors.counting()));

        return countMap.entrySet().stream()
                .filter(entry -> entry.getValue() > 1)
                .map(Map.Entry::getKey)
                .collect(Collectors.toList());
    }
}
```

Output :- Duplicate elements: [1, 2, 3]

Question 42 : Write Java 8 program to find first repeated character in the given string?

```
import java.util.HashSet;

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

```
        String str = "hello world";

        char firstRepeatedChar = findFirstRepeatedCharacter(str);

        if (firstRepeatedChar != '\0') {
            System.out.println("First repeated character: " +
firstRepeatedChar);
        } else {
            System.out.println("No repeated characters found");
        }
    }

    public static char findFirstRepeatedCharacter(String str) {
        HashSet<Character> set = new HashSet<>();

        return str.chars() // Convert the string to an IntStream of
characters
                .mapToObj(ch -> (char) ch) // Convert each character
code to its corresponding character
                .filter(ch -> !set.add(ch)) // Filter out characters
that are already in the set (i.e., repeated characters)
                .findFirst() // Find the first repeated character
                .orElse('\0'); // Return '\0' if no repeated
characters found
    }
}
```

Output :- First repeated character:

Question 43 : Write a Java 8 program to check if a list contains a specific string ?

```
import java.util.Arrays;
import java.util.List;

public class ListContainsStringChecker {
    public static void main(String[] args) {
        List<String> list = Arrays.asList("apple", "banana",
"orange", "grape");

        String searchString = "orange";

        boolean containsString = containsString(list, searchString);

        if (containsString) {
            System.out.println("List contains the string: " +
searchString);
        } else {
            System.out.println("List does not contain the string: "
+ searchString);
        }
    }

    public static boolean containsString(List<String> list, String
searchString) {
        return list.stream()
```

```
                    .anyMatch(searchString::equals); // Using method
reference
    }
}
```

Output :- List contains the string: orange

Question 44 : Write a Java 8 program print all the strings of given length ?

```
import java.util.Arrays;
import java.util.List;

public class ElementsOfCertainLengthPrinter {
    public static void main(String[] args) {
        List<String> list = Arrays.asList("apple", "banana",
"orange", "grape", "kiwi");

        int targetLength = 6;

        printElementsOfCertainLength(list, targetLength);
    }

    public static void printElementsOfCertainLength(List<String>
list, int targetLength) {
        list.stream()
            .filter(str -> str.length() == targetLength)
            .forEach(System.out::println);
    }
}
```

Output :-banana, orange

Question 45 : Write a Java 8 program print first non repetitive character in the string ?

```
import java.util.function.Function;
import java.util.stream.Collectors;
import java.util.Map;

public class FirstNonRepeatedCharacterFinder {
    public static void main(String[] args) {
        String str = "hello world";

        Character firstNonRepeatedChar =
findFirstNonRepeatedChar(str);

        System.out.println("First non-repeated character: " +
firstNonRepeatedChar);
    }

    public static Character findFirstNonRepeatedChar(String str) {
                return str.chars()     // IntStream
                .mapToObj(i -> Character.toLowerCase((char) i))  //
convert to lowercase & then to Character object Stream
                .collect(Collectors.groupingBy(Function.identity(),
LinkedHashMap::new, Collectors.counting())) // store in a
LinkedHashMap with the count
```

```
                .entrySet().stream()                        //
EntrySet stream
                .filter(entry -> entry.getValue() == 1L)    //
extracts characters with a count of 1
                .map(Map.Entry::getKey)              // get the keys
of EntrySet
                .findFirst().get();                  // get the
first entry from the keys
    }
}
```

Output :- First non-repeated character: h

Question 46 : Write a Java 8 program to find the product of all elements in a list ?

```
import java.util.Arrays;
import java.util.List;

public class ProductOfListElements {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);

        int product = calculateProduct(numbers);

        System.out.println("Product of all elements: " + product);
    }

    public static int calculateProduct(List<Integer> numbers) {
        return numbers.stream()
                      .reduce(1, (a, b) -> a * b);
    }
}
```

Output :- Product of all elements: 120

Question 47 : Write a Java 8 program to check if all elements in a list are unique ?

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

public class UniqueElementsChecker {
    public static void main(String[] args) {
        List<Integer> numbers = List.of(1, 2, 3, 4, 5);
        boolean allUnique = areAllElementsUnique(numbers);
        if (allUnique) {
            System.out.println("All elements in the list are
unique.");
        } else {
            System.out.println("List contains duplicate elements.");
        }
    }
```

```
    public static boolean areAllElementsUnique(List<Integer> list) {
        return list.stream()
                   .collect(Collectors.toSet())
                   .size() == list.size();
    }
}
```

Output :- All elements in the list are unique.

Question 48 : Write a Java 8 program to find the first word in a list that starts with given letter ?

```
import java.util.List;
import java.util.Optional;

public class FirstWordStartsWithLetterFinder {
    public static void main(String[] args) {
        List<String> words = List.of("apple", "banana", "orange",
"grape");
        char targetLetter = 'o';

        Optional<String> firstWord =
findFirstWordStartsWithLetter(words, targetLetter);

        if (firstWord.isPresent()) {
            System.out.println("First word starting with '" +
targetLetter + "': " + firstWord.get());
        } else {
            System.out.println("No word starting with '" +
targetLetter + "' found.");
        }
    }

    public static Optional<String>
findFirstWordStartsWithLetter(List<String> words, char targetLetter)
{
        return words.stream()
                    .filter(word -> !word.isEmpty() &&
word.charAt(0) == targetLetter)
                    .findFirst();
    }
}
```

Output :- First word starting with 'o': orange

Question 49 : Write a Java 8 program to find the sum of the first 10 natural numbers ?

```
import java.util.stream.IntStream;

public class SumOfFirstTenNaturalNumbers {
    public static void main(String[] args) {
        int sum = sumOfFirstNNaturalNumbers(10);
        System.out.println("Sum of the first 10 natural numbers: " +
sum);
```

```
    }

    public static int sumOfFirstNNaturalNumbers(int n) {
        return IntStream.rangeClosed(1, n)
                        .sum();
    }
}
```

Output :- Sum of the first 10 natural numbers: 55

Question 50 : Write a Java 8 program to find the product of the first 10 natural numbers ?

```
import java.util.stream.LongStream;

public class ProductCalculator {
    public static void main(String[] args) {
        long product = productOfFirstNNaturalNumbers(10);
        System.out.println("Product of the first 10 natural numbers:
" + product);
    }

    public static long productOfFirstNNaturalNumbers(int n) {
        return LongStream.rangeClosed(1, n)
                         .reduce(1, (a, b) -> a * b);
    }
}
```

Output :- Product of the first 10 natural numbers: 3628800