Assignment No:-52

Name:-Suryawanshi Sangramsingh Sambhaji

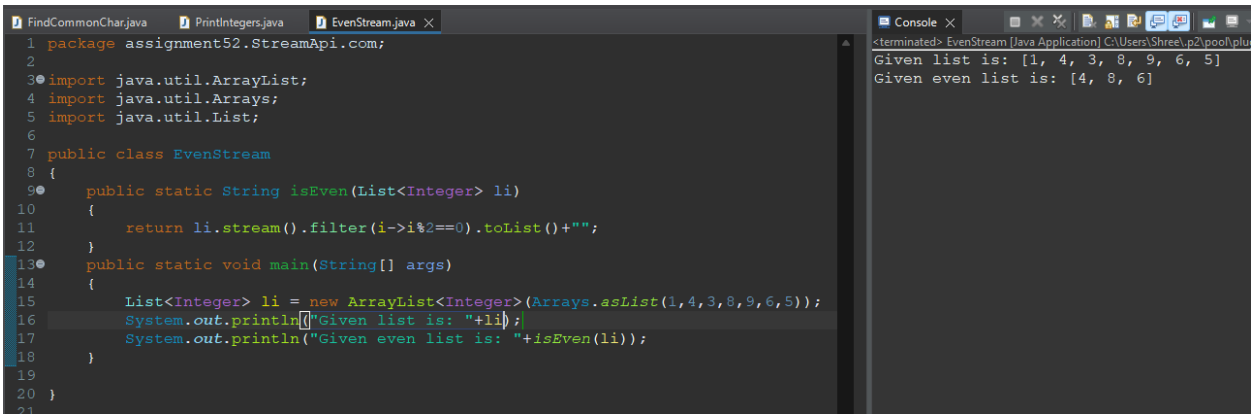
Batch: - Delta - DCA (Java) 2024 Date:-24/7/2024

Q1.Write a program to create a stream from a list of integers and print each element.

```
package assignment52.StreamApi.com;

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

public class PrintIntegers
{
    public static String isInt(List<Integer> li)
    {
        return li.stream().toList()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+isInt(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
```

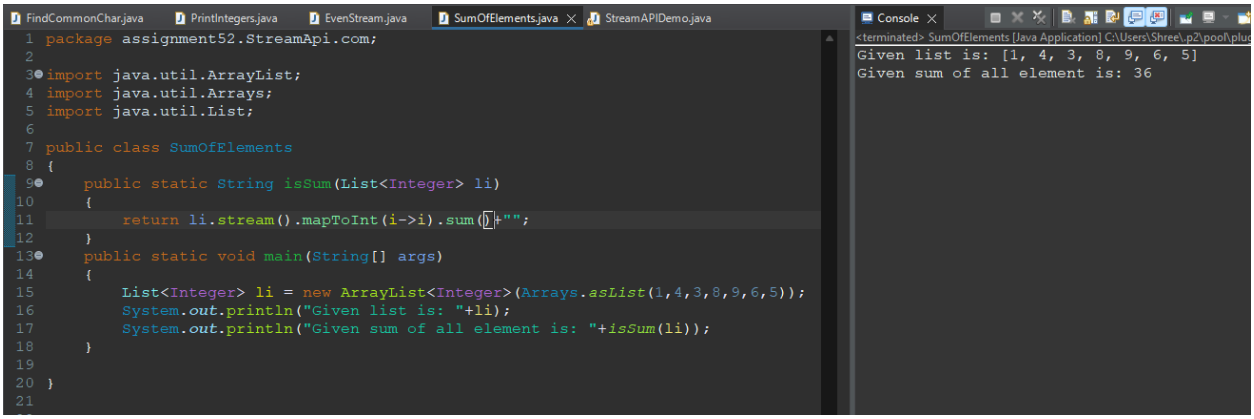
Q2.Implement a program to filter even numbers from a list using the Stream API.



```
package assignment52.StreamApi.com;

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

public class EvenStream
{
    public static String isEven(List<Integer> li)
    {
        return li.stream().filter(i->i%2==0).toList()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given even list is: "+isEven(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given even list is: [4, 8, 6]
```

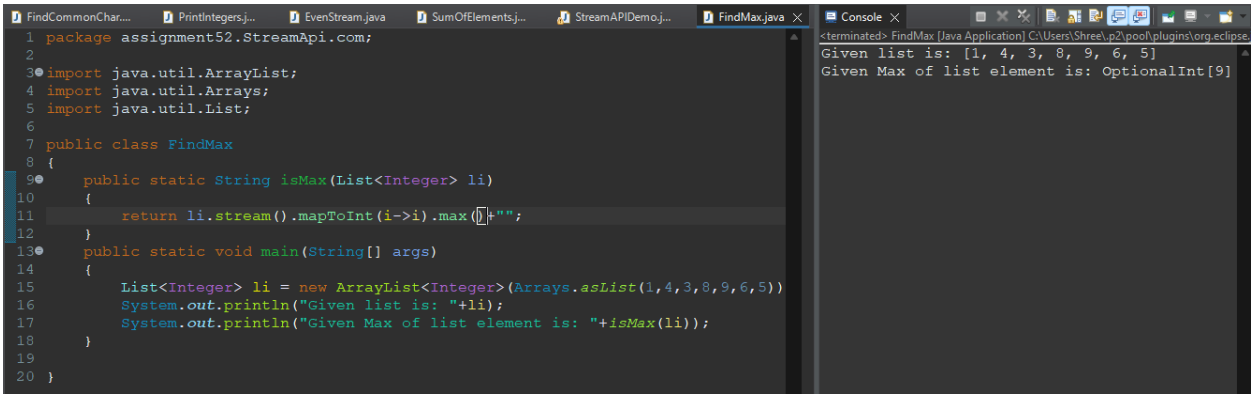
Q3. Write a program to find the sum of all elements in a list using streams.



```
package assignment52.StreamApi.com;

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

public class SumOfElements
{
    public static String isSum(List<Integer> li)
    {
        return li.stream().mapToInt(i->i).sum()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given sum of all element is: "+isSum(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given sum of all element is: 36
```

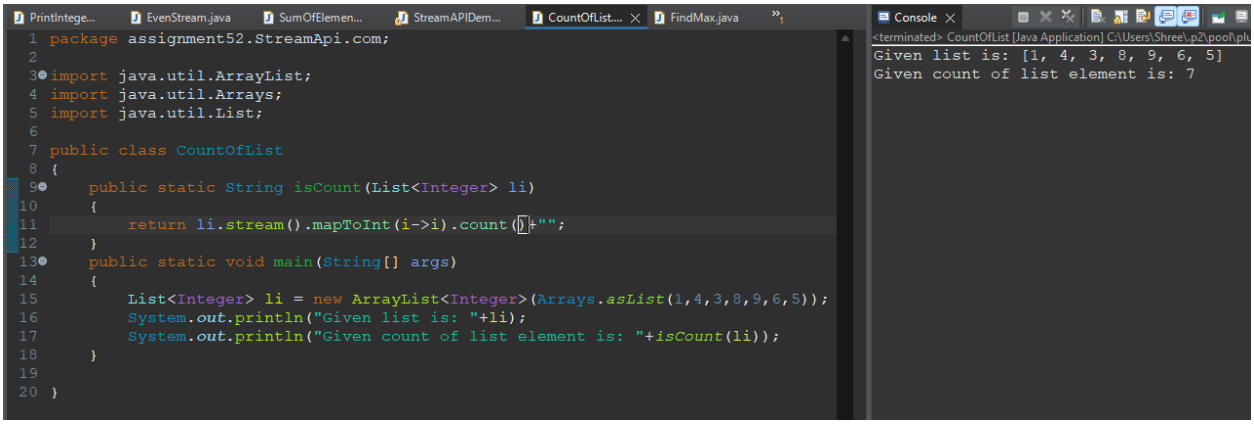
Q4. Implement a program to find the maximum element from a list using streams.



```
package assignment52.StreamApi.com;

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

public class FindMax
{
    public static String isMax(List<Integer> li)
    {
        return li.stream().mapToInt(i->i).max()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5))
        System.out.println("Given list is: "+li);
        System.out.println("Given Max of list element is: "+isMax(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given Max of list element is: OptionalInt[9]
```

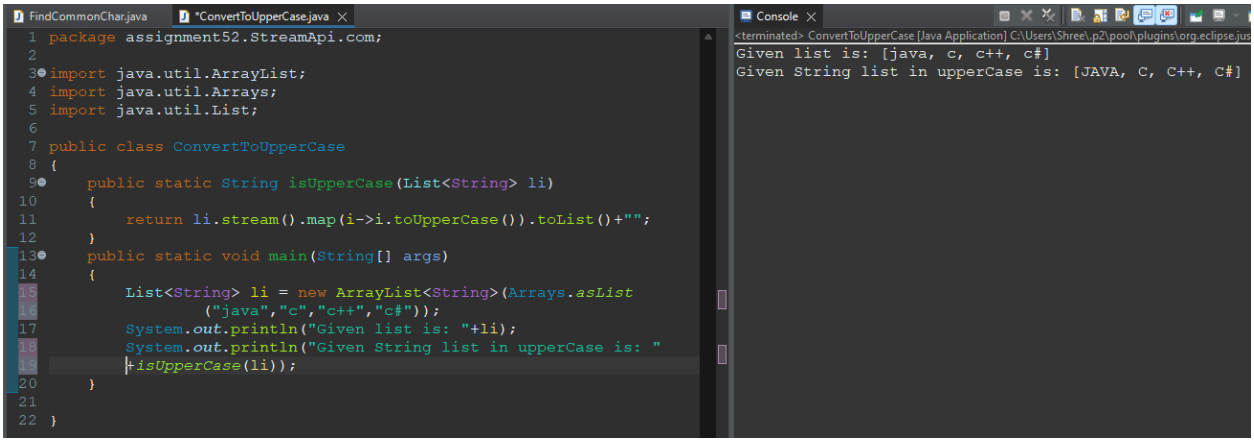
Q5. Write a program to count the number of elements in a list using streams.



```
package assignment52.StreamApi.com;

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

public class CountOfList
{
    public static String isCount(List<Integer> li)
    {
        return li.stream().mapToInt(i->i).count()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given count of list element is: "+isCount(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given count of list element is: 7
```

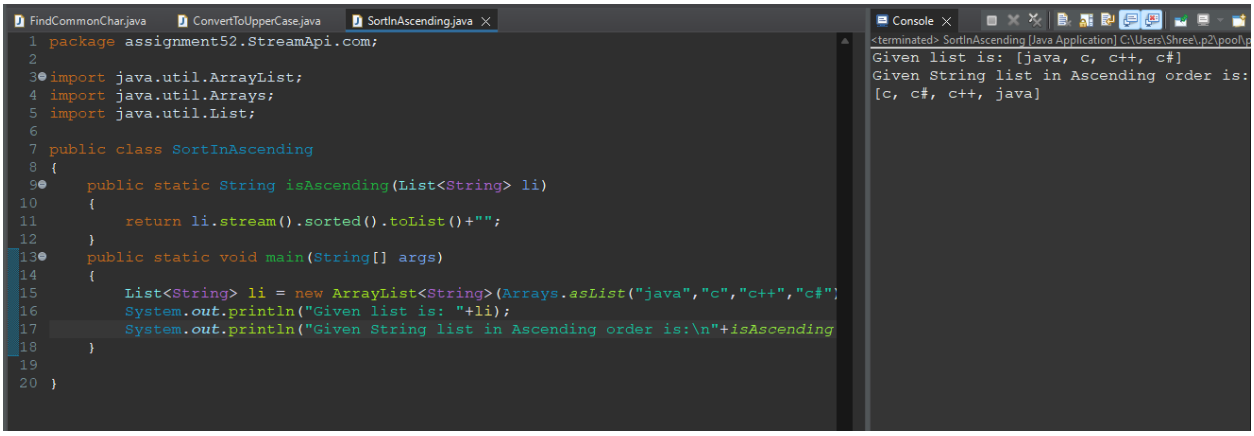
Q6. Implement a program to convert a list of strings to uppercase using streams.



```
package assignment52.StreamApi.com;

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

public class ConvertToUpperCase
{
    public static String isUpperCase(List<String> li)
    {
        return li.stream().map(i->i.toUpperCase()).toList()+"";
    }
    public static void main(String[] args)
    {
        List<String> li = new ArrayList<String>(Arrays.asList
                ("java","c","c++","c#"));
        System.out.println("Given list is: "+li);
        System.out.println("Given String list in upperCase is: "
        +isUpperCase(li));
    }

}
```

```
Given list is: [java, c, c++, c#]
Given String list in upperCase is: [JAVA, C, C++, C#]
```

Q7. Write a program to sort a list of strings in ascending order using streams.



```
package assignment52.StreamApi.com;

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

public class SortInAscending
{
    public static String isAscending(List<String> li)
    {
        return li.stream().sorted().toList()+"";
    }
    public static void main(String[] args)
    {
        List<String> li = new ArrayList<String>(Arrays.asList("java","c","c++","c#"
        System.out.println("Given list is: "+li);
        System.out.println("Given String list in Ascending order is:\n"+isAscending
    }

}
```

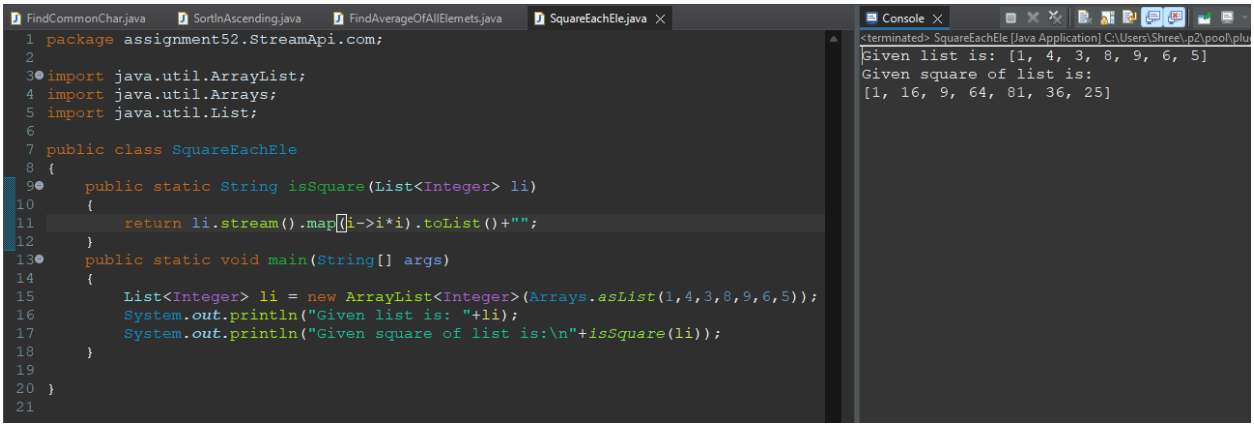
```
Given list is: [java, c, c++, c#]
Given String list in Ascending order is:
[c, c#, c++, java]
```

Q8. Implement a program to find the average of all numbers in a list using streams.

```
package assignment52.StreamApi.com;

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

public class FindAverageOfAllElemets
{
    public static String isAvg(List<Integer> li)
    {
        return li.stream().mapToDouble(i->i).average()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given average of list is:\n"+isAvg(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given average of list is:
OptionalDouble[5.142857142857143]
```

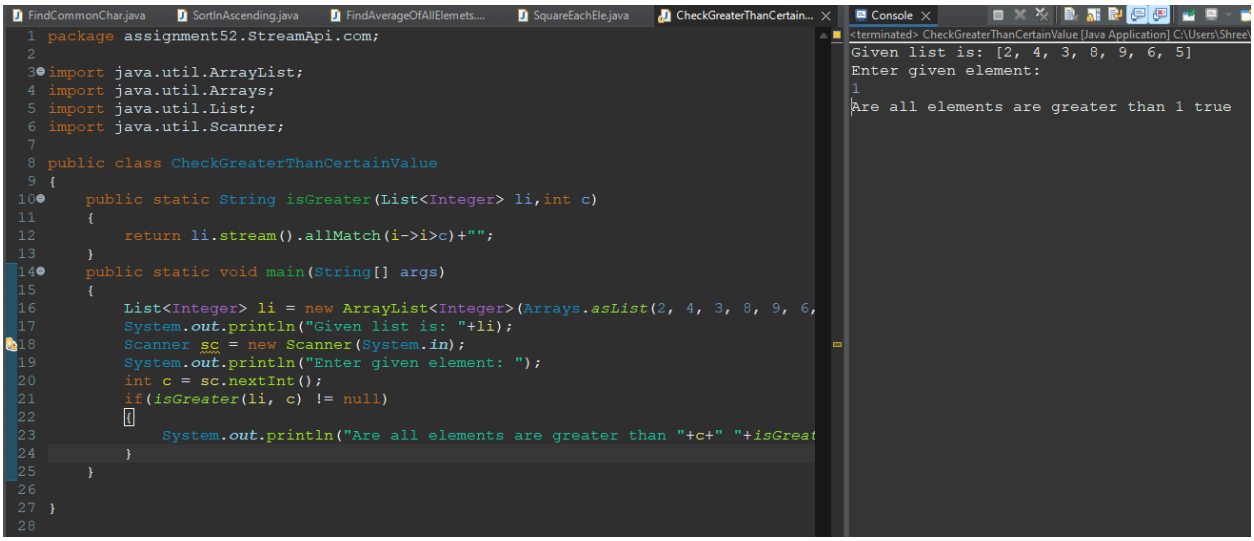
Q9. Write a program to find the square of each element in a list using streams.



```java
package assignment52.StreamApi.com;

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

public class SquareEachEle
{
    public static String isSquare(List<Integer> li)
    {
        return li.stream().map(i->i*i).toList()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given square of list is:\n"+isSquare(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given square of list is:
[1, 16, 9, 64, 81, 36, 25]
```

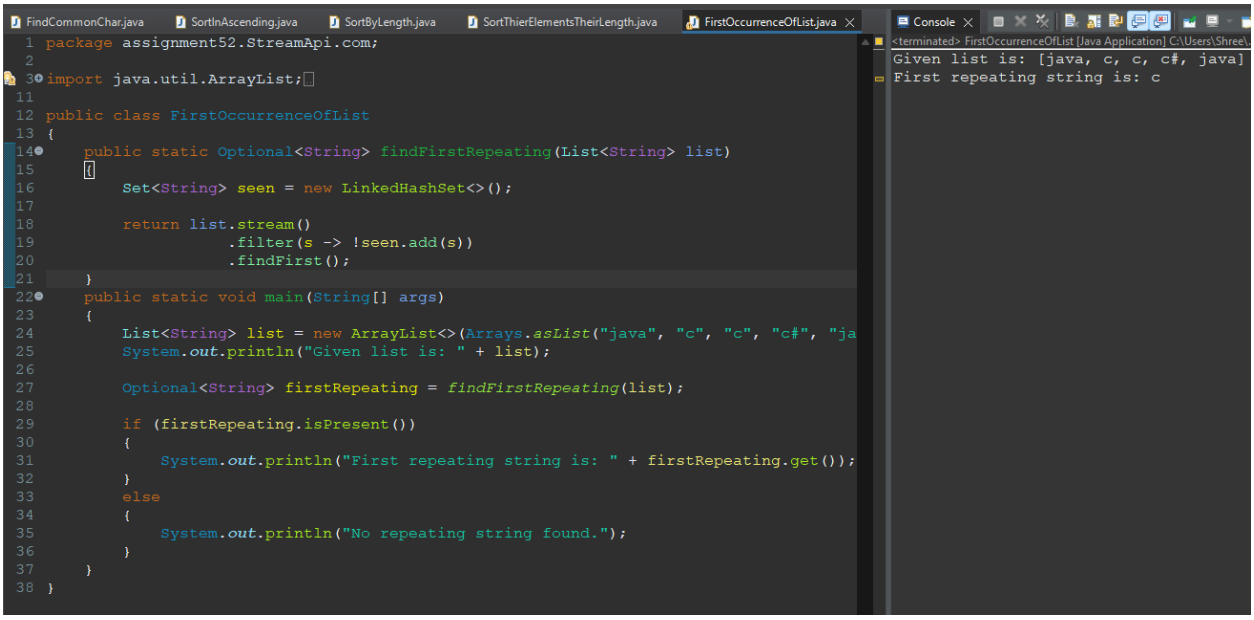
Q10. Implement a program to check if all elements in a list are greater than a certain value using streams.



```java
package assignment52.StreamApi.com;

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

public class CheckGreaterThanCertainValue
{
    public static String isGreater(List<Integer> li,int c)
    {
        return li.stream().allMatch(i->i>c)+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(2, 4, 3, 8, 9, 6,
        System.out.println("Given list is: "+li);
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter given element: ");
        int c = sc.nextInt();
        if(isGreater(li, c) != null)
        {
            System.out.println("Are all elements are greater than "+c+" "+isGreat
        }
    }

}
```

```
Given list is: [2, 4, 3, 8, 9, 6, 5]
Enter given element:
1
Are all elements are greater than 1 true
```

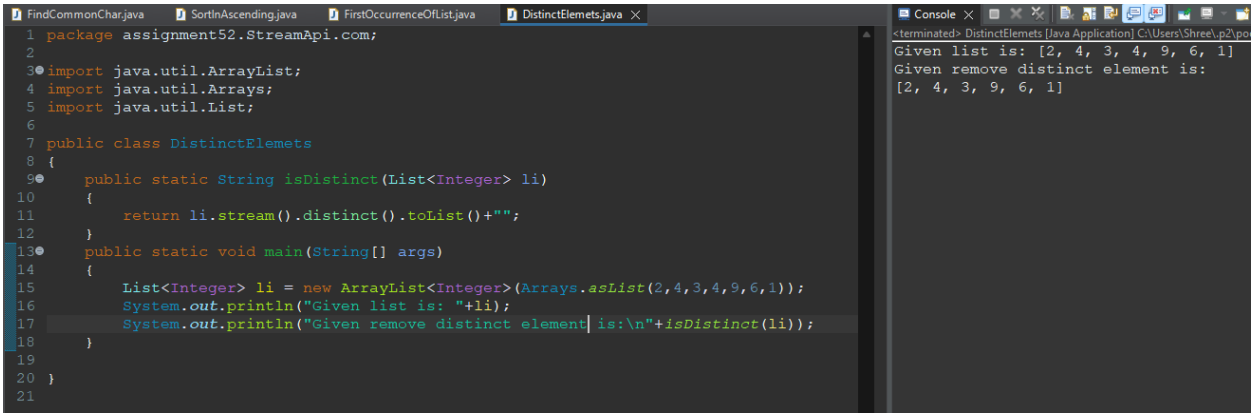
Q11. Write a program to find the first occurrence of a given element in a list using streams.



```java
package assignment52.StreamApi.com;

import java.util.ArrayList;

public class FirstOccurrenceOfList
{
    public static Optional<String> findFirstRepeating(List<String> list)
    {
        Set<String> seen = new LinkedHashSet<>();

        return list.stream()
                   .filter(s -> !seen.add(s))
                   .findFirst();
    }
    public static void main(String[] args)
    {
        List<String> list = new ArrayList<>(Arrays.asList("java", "c", "c", "c#", "ja
        System.out.println("Given list is: " + list);

        Optional<String> firstRepeating = findFirstRepeating(list);

        if (firstRepeating.isPresent())
        {
            System.out.println("First repeating string is: " + firstRepeating.get());
        }
        else
        {
            System.out.println("No repeating string found.");
        }
    }
}
```

```
Given list is: [java, c, c, c#, java]
First repeating string is: c
```

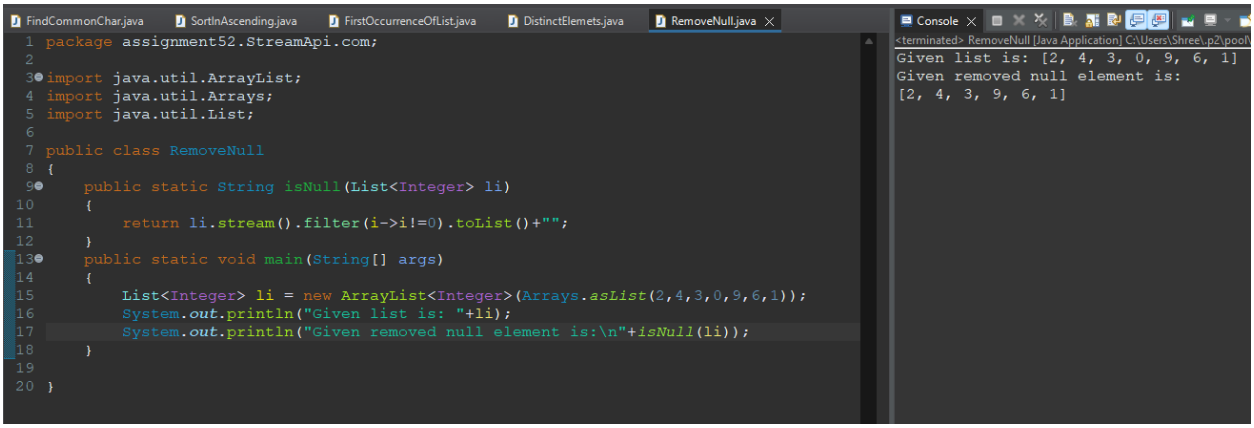
Q12. Implement a program to find the distinct elements in a list using streams.



```java
package assignment52.StreamApi.com;

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

public class DistinctElemets
{
    public static String isDistinct(List<Integer> li)
    {
        return li.stream().distinct().toList()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(2,4,3,4,9,6,1));
        System.out.println("Given list is: "+li);
        System.out.println("Given remove distinct element is:\n"+isDistinct(li));
    }

}
```

```
Given list is: [2, 4, 3, 4, 9, 6, 1]
Given remove distinct element is:
[2, 4, 3, 9, 6, 1]
```

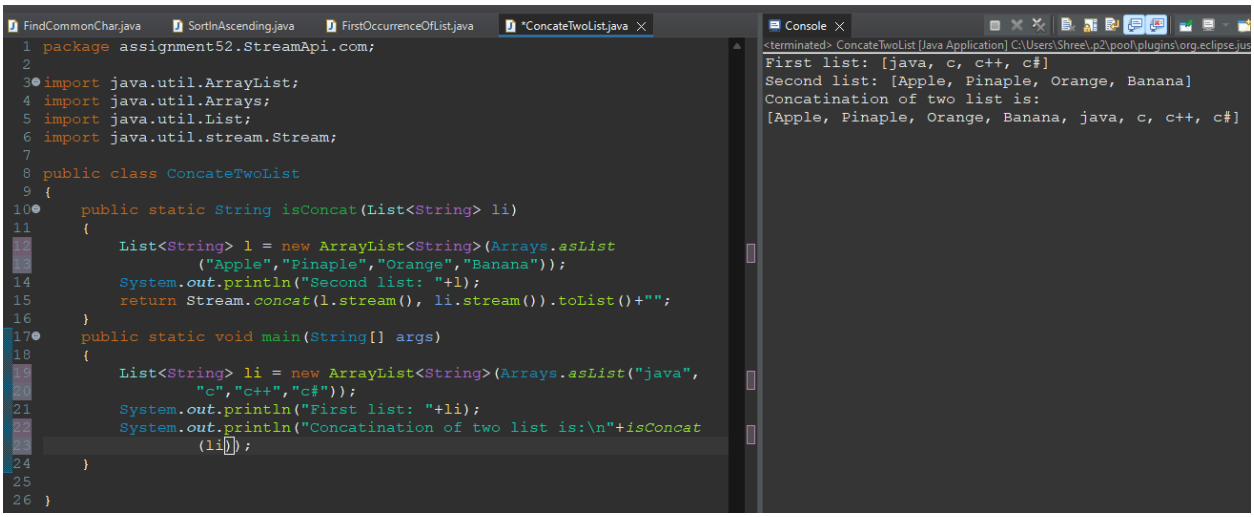
Q13. Write a program to remove null values from a list using streams.



```java
package assignment52.StreamApi.com;

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

public class RemoveNull
{
    public static String isNull(List<Integer> li)
    {
        return li.stream().filter(i->i!=0).toList()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(2,4,3,0,9,6,1));
        System.out.println("Given list is: "+li);
        System.out.println("Given removed null element is:\n"+isNull(li));
    }

}
```

```
Given list is: [2, 4, 3, 0, 9, 6, 1]
Given removed null element is:
[2, 4, 3, 9, 6, 1]
```

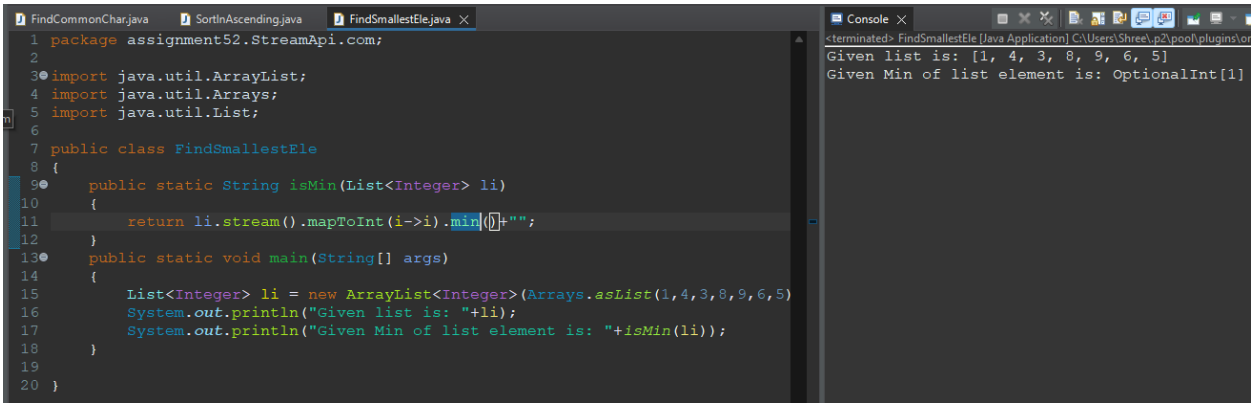
Q14. Implement a program to concatenate two lists using streams.



```
package assignment52.StreamApi.com;

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

public class ConcateTwoList
{
    public static String isConcat(List<String> li)
    {
        List<String> l = new ArrayList<String>(Arrays.asList
                ("Apple","Pinaple","Orange","Banana"));
        System.out.println("Second list: "+l);
        return Stream.concat(l.stream(), li.stream()).toList()+"";
    }
    public static void main(String[] args)
    {
        List<String> li = new ArrayList<String>(Arrays.asList("java",
                "c","c++","c#"));
        System.out.println("First list: "+li);
        System.out.println("Concatination of two list is:\n"+isConcat
                (li));
    }

}
```

```
First list: [java, c, c++, c#]
Second list: [Apple, Pinaple, Orange, Banana]
Concatination of two list is:
[Apple, Pinaple, Orange, Banana, java, c, c++, c#]
```

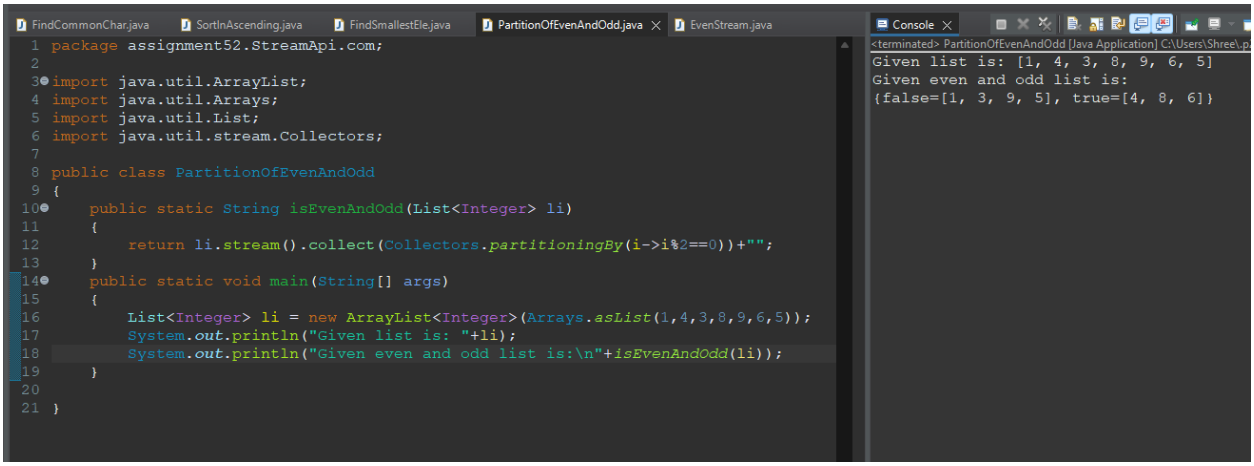
Q15. Write a program to find the second smallest element in a list using streams.



```
package assignment52.StreamApi.com;

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

public class FindSmallestEle
{
    public static String isMin(List<Integer> li)
    {
        return li.stream().mapToInt(i->i).min()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given Min of list element is: "+isMin(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given Min of list element is: OptionalInt[1]
```

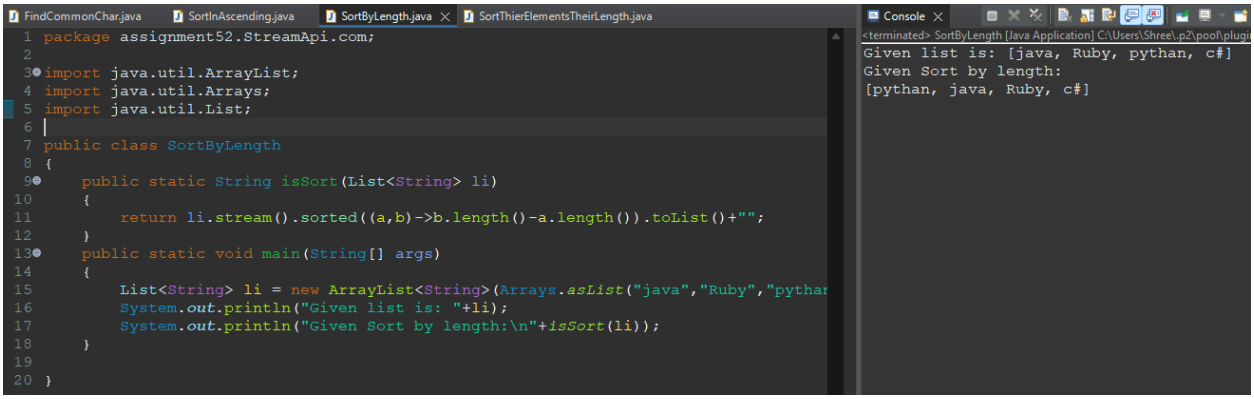
Q16. Implement a program to partition a list into even and odd numbers using streams.



```
package assignment52.StreamApi.com;

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

public class PartitionOfEvenAndOdd
{
    public static String isEvenAndOdd(List<Integer> li)
    {
        return li.stream().collect(Collectors.partitioningBy(i->i%2==0))+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
        System.out.println("Given list is: "+li);
        System.out.println("Given even and odd list is:\n"+isEvenAndOdd(li));
    }

}
```

```
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given even and odd list is:
{false=[1, 3, 9, 5], true=[4, 8, 6]}
```

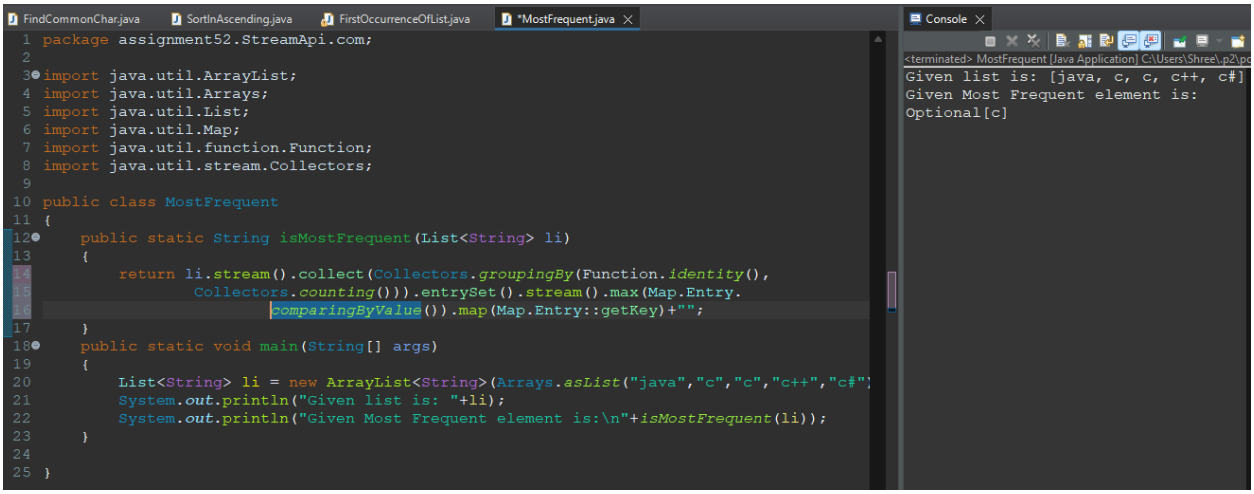
Q17. Write a program to group elements of a list by their lengths using streams.



```
package assignment52.StreamApi.com;

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

public class SortByLength
{
    public static String isSort(List<String> li)
    {
        return li.stream().sorted((a,b)->b.length()-a.length()).toList()+"";
    }
    public static void main(String[] args)
    {
        List<String> li = new ArrayList<String>(Arrays.asList("java","Ruby","pythar
        System.out.println("Given list is: "+li);
        System.out.println("Given Sort by length:\n"+isSort(li));
    }

}
```

```
Given list is: [java, Ruby, pythan, c#]
Given Sort by length:
[pythan, java, Ruby, c#]
```

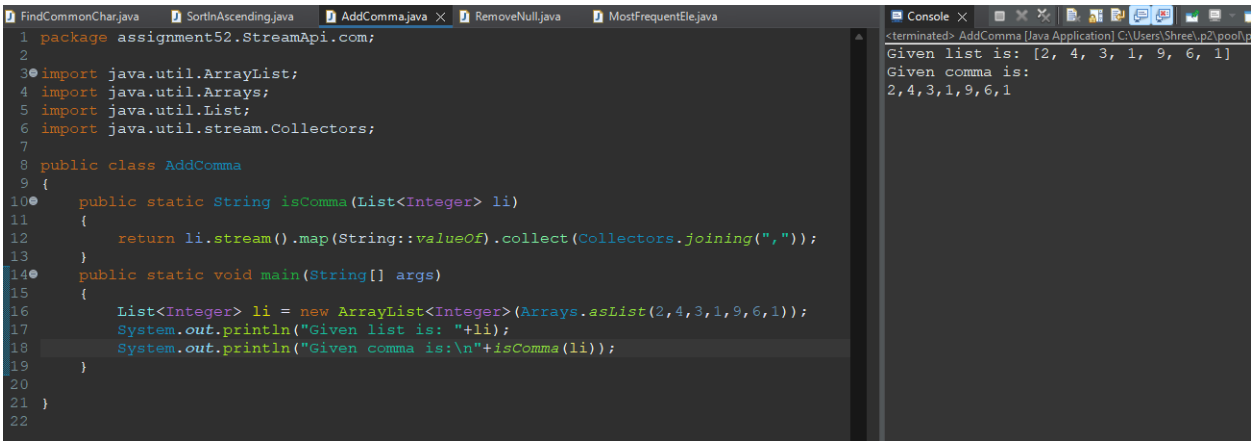
Q18. Implement a program to find the most frequently occurring element in a list using streams.



```
package assignment52.StreamApi.com;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
import java.util.Map;
import java.util.function.Function;
import java.util.stream.Collectors;

public class MostFrequent
{
    public static String isMostFrequent(List<String> li)
    {
        return li.stream().collect(Collectors.groupingBy(Function.identity(),
                Collectors.counting())).entrySet().stream().max(Map.Entry.
                        comparingByValue()).map(Map.Entry::getKey)+"";
    }
    public static void main(String[] args)
    {
        List<String> li = new ArrayList<String>(Arrays.asList("java","c","c","c++","c#"
        System.out.println("Given list is: "+li);
        System.out.println("Given Most Frequent element is:\n"+isMostFrequent(li));
    }

}
```

```
Given list is: [java, c, c, c++, c#]
Given Most Frequent element is:
Optional[c]
```

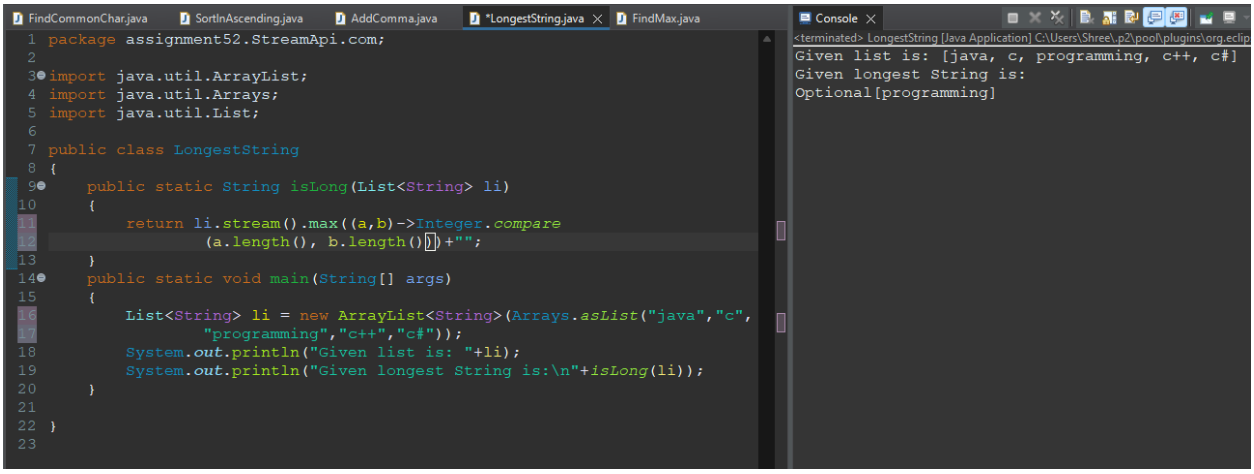
Q19. Write a program to convert a list of integers to a comma-separated string using streams.



```
package assignment52.StreamApi.com;

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

public class AddComma
{
    public static String isComma(List<Integer> li)
    {
        return li.stream().map(String::valueOf).collect(Collectors.joining(","));
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(2,4,3,1,9,6,1));
        System.out.println("Given list is: "+li);
        System.out.println("Given comma is:\n"+isComma(li));
    }

}
```

```
Given list is: [2, 4, 3, 1, 9, 6, 1]
Given comma is:
2,4,3,1,9,6,1
```

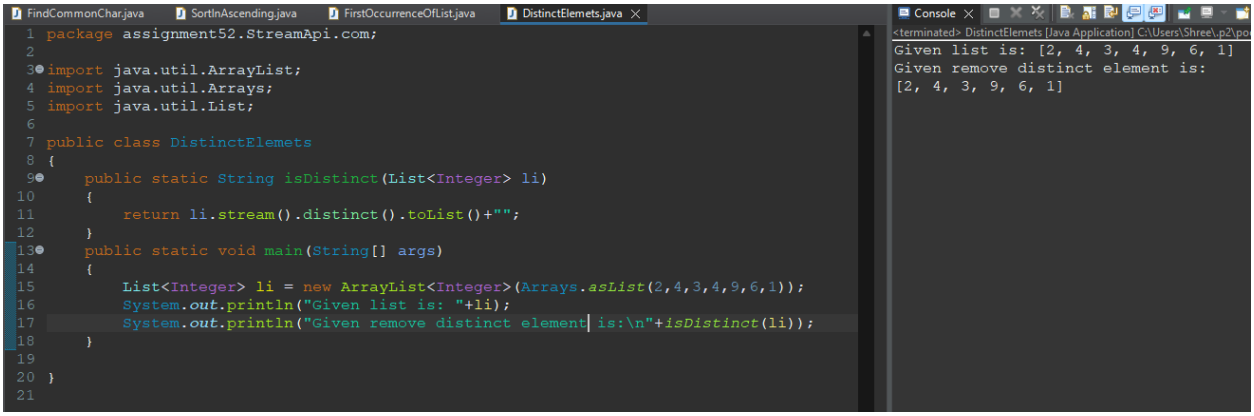
Q20. Implement a program to find the longest string in a list using streams.



```
package assignment52.StreamApi.com;

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

public class LongestString
{
    public static String isLong(List<String> li)
    {
        return li.stream().max((a,b)->Integer.compare
                (a.length(), b.length()))+"";
    }
    public static void main(String[] args)
    {
        List<String> li = new ArrayList<String>(Arrays.asList("java","c",
                "programming","c++","c#"));
        System.out.println("Given list is: "+li);
        System.out.println("Given longest String is:\n"+isLong(li));
    }

}
```

```
<terminated> LongestString [Java Application] C:\Users\Shree\.p2\pool\plugins\org.eclip
Given list is: [java, c, programming, c++, c#]
Given longest String is:
Optional[programming]
```

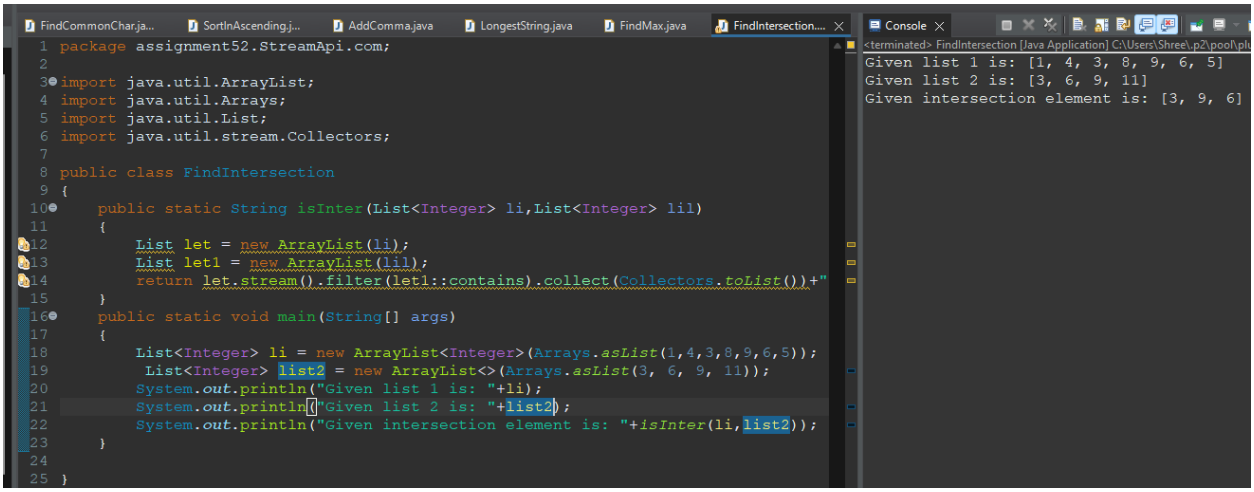
Q21. Write a program to remove duplicate elements from a list using streams.



```
package assignment52.StreamApi.com;

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

public class DistinctElemets
{
    public static String isDistinct(List<Integer> li)
    {
        return li.stream().distinct().toList()+"";
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(2,4,3,4,9,6,1));
        System.out.println("Given list is: "+li);
        System.out.println("Given remove distinct element is:\n"+isDistinct(li));
    }

}
```

```
<terminated> DistinctElemets [Java Application] C:\Users\Shree\.p2\po
Given list is: [2, 4, 3, 4, 9, 6, 1]
Given remove distinct element is:
[2, 4, 3, 9, 6, 1]
```

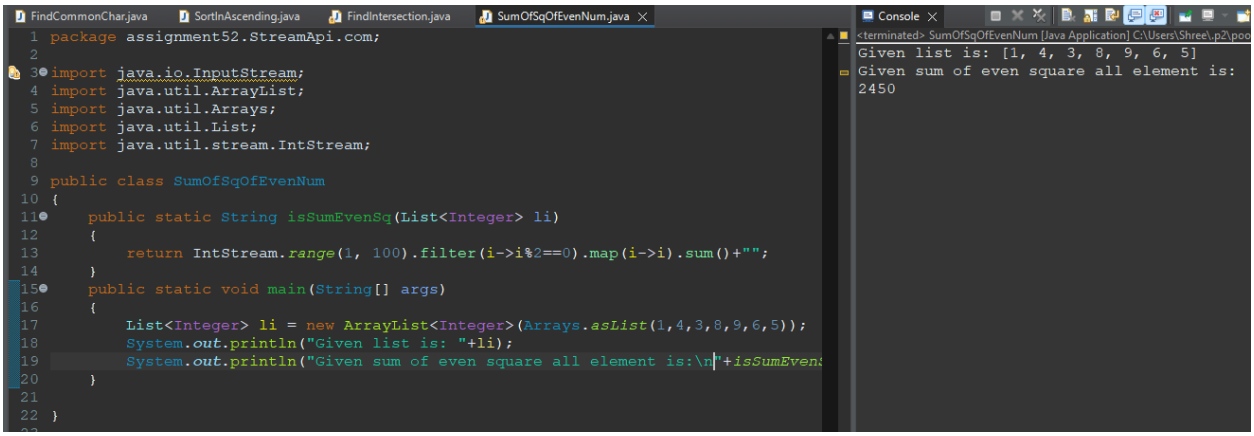
Q22. Implement a program to find the intersection of two lists using streams.



```
package assignment52.StreamApi.com;

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

public class FindIntersection
{
    public static String isInter(List<Integer> li,List<Integer> lil)
    {
        List let = new ArrayList(li);
        List let1 = new ArrayList(lil);
        return let.stream().filter(let1::contains).collect(Collectors.toList())+"
    }
    public static void main(String[] args)
    {
        List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
         List<Integer> list2 = new ArrayList<>(Arrays.asList(3, 6, 9, 11));
        System.out.println("Given list 1 is: "+li);
        System.out.println("Given list 2 is: "+list2);
        System.out.println("Given intersection element is: "+isInter(li,list2));
    }

}
```

```
<terminated> FindIntersection [Java Application] C:\Users\Shree\.p2\pool\plu
Given list 1 is: [1, 4, 3, 8, 9, 6, 5]
Given list 2 is: [3, 6, 9, 11]
Given intersection element is: [3, 9, 6]
```

Q23. Write a program to create a stream of numbers from 1 to 100 and find the sum of squares of all even numbers.

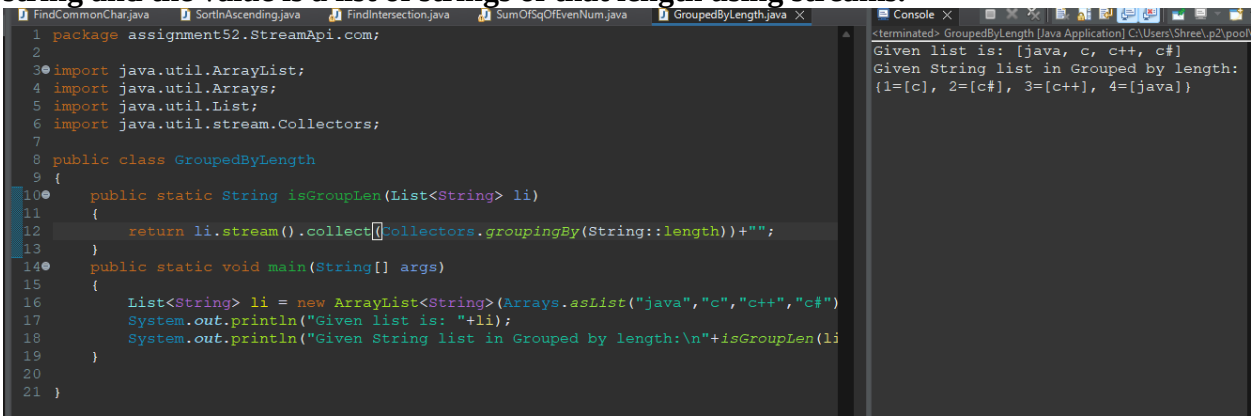


```
FindCommonChar.java  SortInAscending.java  FindIntersection.java  SumOfSqOfEvenNum.java

1  package assignment52.StreamApi.com;
2
3  import java.io.InputStream;
4  import java.util.ArrayList;
5  import java.util.Arrays;
6  import java.util.List;
7  import java.util.stream.IntStream;
8
9  public class SumOfSqOfEvenNum
10 {
11     public static String isSumEvenSq(List<Integer> li)
12     {
13         return IntStream.range(1, 100).filter(i->i%2==0).map(i->i).sum()+"";
14     }
15     public static void main(String[] args)
16     {
17         List<Integer> li = new ArrayList<Integer>(Arrays.asList(1,4,3,8,9,6,5));
18         System.out.println("Given list is: "+li);
19         System.out.println("Given sum of even square all element is:\n"+isSumEvens
20     }
21
22 }
```

```
Console
<terminated> SumOfSqOfEvenNum [Java Application] C:\Users\Shree\.p2\poo
Given list is: [1, 4, 3, 8, 9, 6, 5]
Given sum of even square all element is:
2450
```

Q24. Implement a program to find the cumulative sum of elements in a list using streams.

Q25. Write a program to convert a list of strings to a map where the key is the length of the string and the value is a list of strings of that length using streams.



```
FindCommonChar.java  SortInAscending.java  FindIntersection.java  SumOfSqOfEvenNum.java  GroupedByLength.java

1  package assignment52.StreamApi.com;
2
3  import java.util.ArrayList;
4  import java.util.Arrays;
5  import java.util.List;
6  import java.util.stream.Collectors;
7
8  public class GroupedByLength
9  {
10     public static String isGroupLen(List<String> li)
11     {
12         return li.stream().collect(Collectors.groupingBy(String::length))+"";
13     }
14     public static void main(String[] args)
15     {
16         List<String> li = new ArrayList<String>(Arrays.asList("java","c","c++","c#")
17         System.out.println("Given list is: "+li);
18         System.out.println("Given String list in Grouped by length:\n"+isGroupLen(li
19     }
20
21 }
```

```
Console
<terminated> GroupedByLength [Java Application] C:\Users\Shree\.p2\pool
Given list is: [java, c, c++, c#]
Given String list in Grouped by length:
{1=[c], 2=[c#], 3=[c++], 4=[java]}
```