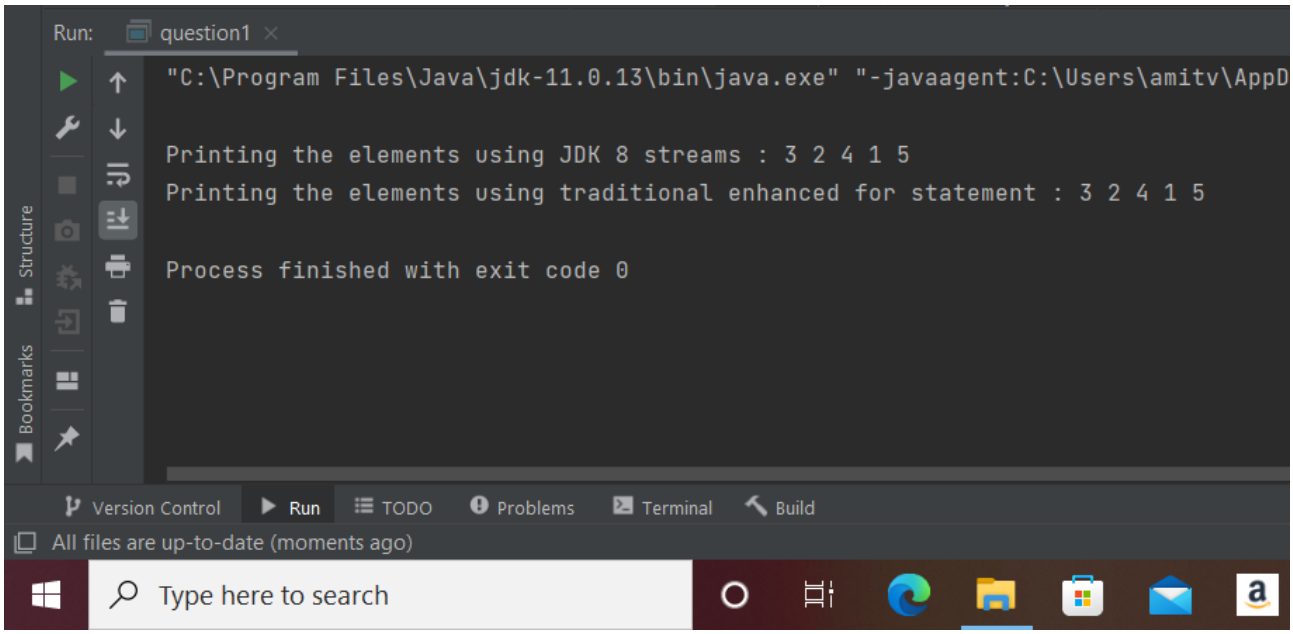
Collections Assessment

1. Write a program that prints its arguments in random order. Do not make a copy of the argument array. Demonstrate how to print out the elements using both streams and the traditional enhanced for statement.

Code : -

```
// Write a program that prints its arguments in random order. Do not make a copy of the argument array.
// Demonstrate how to print out the elements using both streams and the traditional enhanced for statement

package com.company;
import java.util.*;

public class question1 {
    public static void main(String[] args) {

        // Get and shuffle the list of arguments
        List<Integer> argList = Arrays.asList(1,2,3,4,5);
        Collections.shuffle(argList);

        // Print out the elements using JDK 8 Streams
        System.out.print("\nPrinting the elements using JDK 8 streams : ");
        argList.stream().forEach(e->System.out.format("%d ",e));

        // Print out the elements using for-each
        System.out.print("\nPrinting the elements using traditional enhanced for statement : ");
        for (Integer arg: argList) {
            System.out.format("%d ", arg);
        }
        System.out.println();
    }
}
```

Output : -


Run:
question1
"C:\Program Files\Java\jdk-11.0.13\bin\java.exe" "-javaagent:C:\Users\amitv\AppD
Printing the elements using JDK 8 streams : 3 2 4 1 5
Printing the elements using traditional enhanced for statement : 3 2 4 1 5
Process finished with exit code 0
Structure
Bookmarks
Version Control
Run
TODO
Problems
Terminal
Build
All files are up-to-date (moments ago)
Type here to search

2. Take the FindDups example and modify it to use a SortedSet instead of a Set. Specify a Comparator so that case is ignored when sorting and identifying set elements.

Code : -

```
// Take the FindDups example and modify it to use a SortedSet instead of a Set.
// Specify a Comparator so that case is ignored when sorting and identifying set
elements.

package com.company;
import java.util.*;

public class question2 {

    static final Comparator<String> IGNORE_CASE_ORDER = new
Comparator<String>() {
        public int compare(String s1, String s2) {
            return s1.compareToIgnoreCase(s2);
        }
    };

    public static void main(String[] args) {
        SortedSet<String> s = new TreeSet<String>(IGNORE_CASE_ORDER);
        String[] arr = {"Java","Python", "javascript", "react","React","java"};
        Collections.addAll(s, arr);
        System.out.println("\n"+s.size() + " distinct words from given array are : " + s);
    }
}
```

Output : -

```
Run:  question2 ×
"C:\Program Files\Java\jdk-11.0.13\bin\java.exe" "-javaagent:C:\Users\amitv\A

4 distinct words from given array are : [Java, javascript, Python, react]

Process finished with exit code 0
```

3. Write a method that takes a List<String> and applies String.trim to each element.

Code : -

```
// Write a method that takes a List<String> and applies String.trim to each element

package com.company;
import java.util.*;

public class question3 {
    static void listTrim(List<String> strings) {
        for (ListIterator<String> lit = strings.listIterator(); lit.hasNext(); ) {
            lit.set(lit.next().trim());
        }
    }

    public static void main(String[] args) {
        List<String> l = Arrays.asList(" red ", " white ", " blue ");
        listTrim(l);
        System.out.println();
        for (String s : l) {
            System.out.format("\"%s\"%n", s);
        }
    }
}
```

Output : -

```
Run:  question3
"C:\Program Files\Java\jdk-11.0.13\bin\java.exe" "-javaagent:C:\Users\amitv\A

"red"
"white"
"blue"

Process finished with exit code 0
```

4. Consider the four core interfaces, Set, List, Queue, and Map. For each of the following four assignments, specify which of the four core interfaces is best-suited, and explain how to use it to implement the assignment.

i) Whimsical Toys Inc (WTI) needs to record the names of all its employees. Every month, an employee will be chosen at random from these records to receive a free toy.

Ans) Use a **List**. Choose a random employee by picking a number between 0 and size()-1.

ii) WTI has decided that each new product will be named after an employee — but only first names will be used, and each name will be used only once. Prepare a list of unique first names.

Ans) Use a **Set**. Collections that implement this interface don't allow the same element to be entered more than once.

iii) WTI decides that it only wants to use the most popular names for its toys. Count up the number of employees who have each first name.

Ans) Use a **Map**, where the keys are first names, and each value is a count of the number of employees with that first name.

iv) WTI acquires season tickets for the local lacrosse team, to be shared by employees. Create a waiting list for this popular sport.

Ans) Use a **Queue**. Invoke add() to add employees to the waiting list, and remove() to remove them.