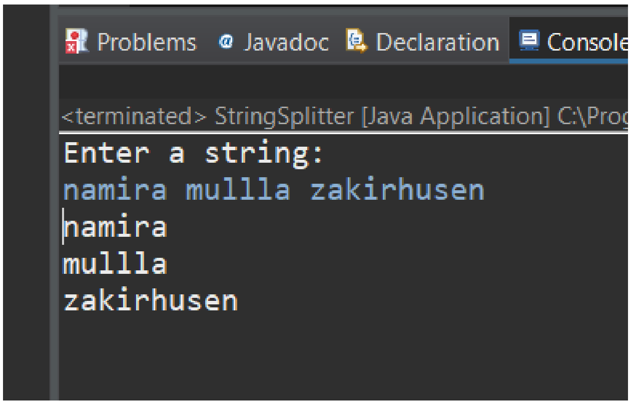
Lab5-Namira mulla

1. Write a Java program that reads a string from the user and uses StringTokenizer to split the string into individual words. Print each word on a new line.

Code:

```
package lab5;

import java.util.Scanner;

import java.util.StringTokenizer;

public class StringSplitter {

        public static void main(String[] args) {

                Scanner scanner = new Scanner(System.in);

                // Prompt the user to enter a string

                System.out.println("Enter a string:");

                String inputString = scanner.nextLine();

                // Create a StringTokenizer object to split the string into words

                StringTokenizer tokenizer = new StringTokenizer(inputString);

                while (tokenizer.hasMoreTokens()) {

                        String word = tokenizer.nextToken();

                        System.out.println(word);

                }scanner.close();

        }

}
```

Output:



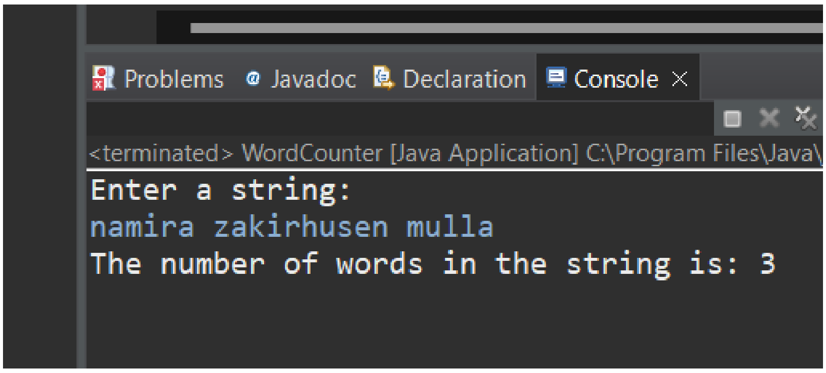
```
<terminated> StringSplitter [Java Application] C:\Proc
Enter a string:
namira mullla zakirhusen
namira
mullla
zakirhusen
```

2. Write a Java program that reads a string from the user and uses StringTokenizer to count the number of words in the string.

Code:

```
package lab5;

import java.util.Scanner;

import java.util.StringTokenizer;

public class WordCounter {

	public static void main(String[] args) {

		Scanner scanner = new Scanner(System.in);

		System.out.println("Enter a string:");

		String inputString = scanner.nextLine();

		StringTokenizer tokenizer = new StringTokenizer(inputString);

		int wordCount = 0;

		while (tokenizer.hasMoreTokens()) {

			tokenizer.nextToken();

			wordCount++;

		}

		System.out.println("The number of words in the string is: " + wordCount);

		scanner.close();

	}

}
```

Output:



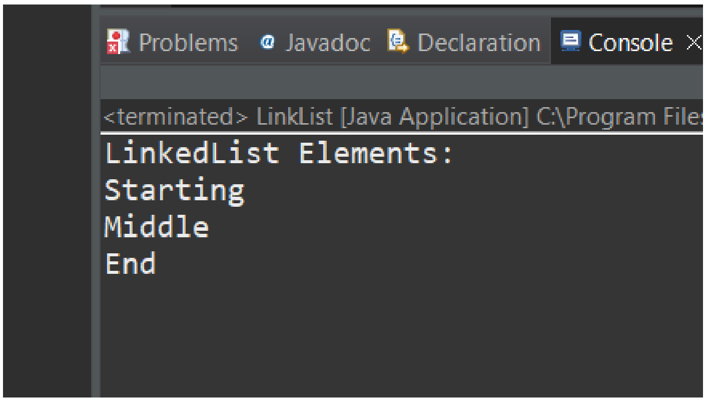
```
<terminated> WordCounter [Java Application] C:\Program Files\Java\
Enter a string:
namira zakirhusen mulla
The number of words in the string is: 3
```

3. Write a Java program to create a LinkedList of strings, add elements at specific positions (beginning, middle, end), and print the list.

Code:

```
package lab5;

import java.util.LinkedList;

public class LinkList {

	public static void main(String[] args) {

		LinkedList<String>linkedList = new LinkedList<>();

		linkedList.add("End");

		linkedList.addFirst("Starting");

		linkedList.add(1,"Middle");

		System.out.println("LinkedList Elements:");

		for (String element:linkedList) {

			System.out.println(element);

		}

	}


}
```

Output:



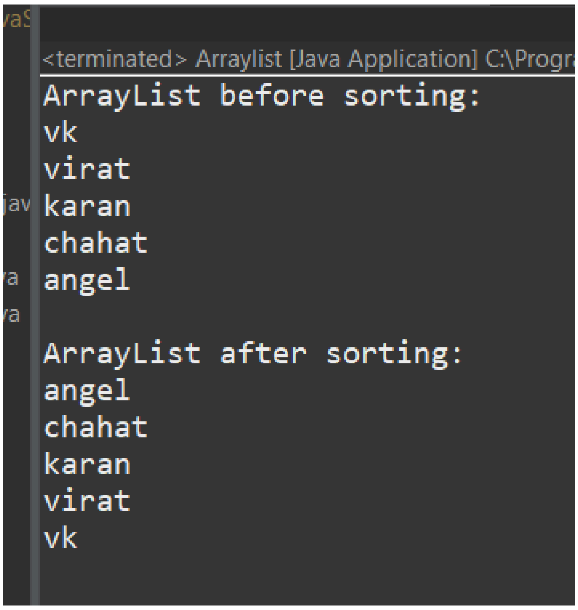
```
<terminated> LinkList [Java Application] C:\Program File
LinkedList Elements:
Starting
Middle
End
```

4. Write a Java program to sort a given array list.

Code:

```
package lab5;

import java.util.ArrayList;

import java.util.Collections;

public class Arraylist {
    public static void main(String[] args) {
        ArrayList<String> arrayList = new ArrayList<>();
        arrayList.add("vk");
        arrayList.add("virat");
        arrayList.add("karan");
        arrayList.add("chahat");
        arrayList.add("angel");
        System.out.println("ArrayList before sorting:");
        for (String element : arrayList) {
            System.out.println(element);
        }
        Collections.sort(arrayList);
        System.out.println("\nArrayList after sorting:");
        for (String element : arrayList) {
            System.out.println(element);
        }
    }
}
```

Output:



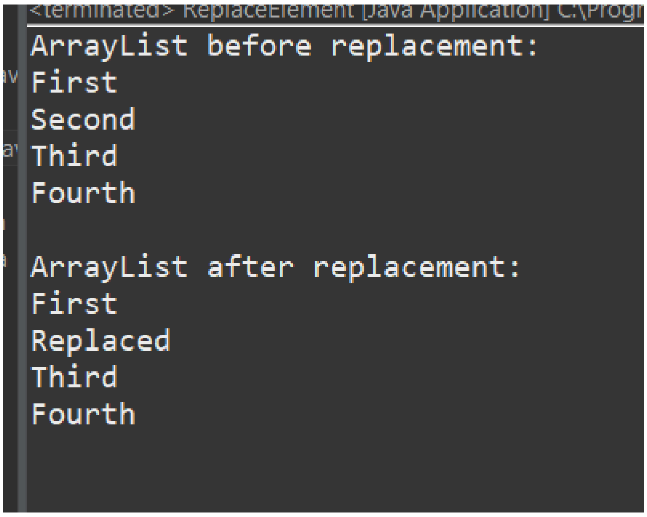
```
<terminated> Arraylist [Java Application] C:\Progr
ArrayList before sorting:
vk
virat
karan
chahat
angel

ArrayList after sorting:
angel
chahat
karan
virat
vk
```

5. Write a Java program to replace the second element of an ArrayList with the specified element.

Code:

```
package lab5;
import java.util.ArrayList;

public class ReplaceElement {
	public static void main(String[] args) {
		ArrayList<String> arrayList = new ArrayList<>();
		arrayList.add("First");
		arrayList.add("Second");
		arrayList.add("Third");
		arrayList.add("Fourth");
		System.out.println("ArrayList before replacement:");
		for (String element : arrayList) {
			System.out.println(element);
		}
		String newElement = "Replaced";
		if (arrayList.size() > 1) {
			arrayList.set(1, newElement);
		}
		System.out.println("\nArrayList after replacement:");
		for (String element : arrayList) {
			System.out.println(element);
		}
	}
}
```

Output:



```
ArrayList before replacement:
First
Second
Third
Fourth

ArrayList after replacement:
First
Replaced
Third
Fourth
```

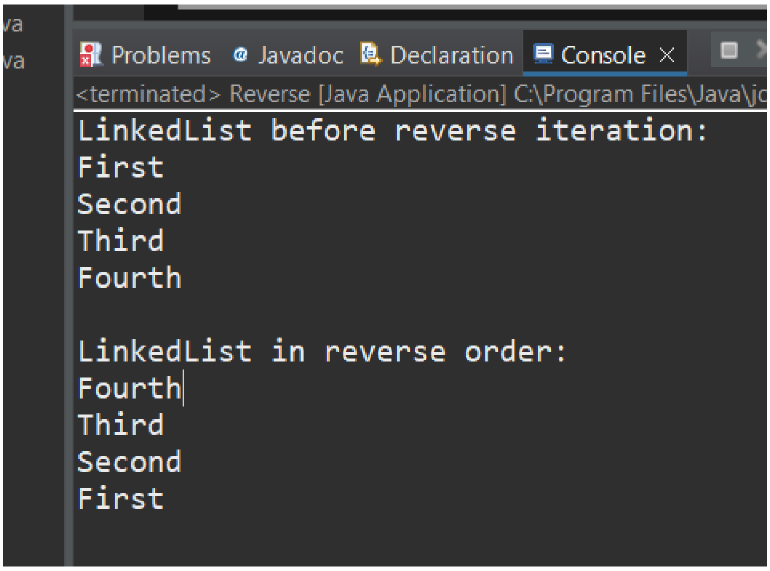
6. Write a Java program to iterate a linked list in reverse order.

Code:

```
package lab5;

import java.util.LinkedList;

import java.util.ListIterator;


public class Reverse {

            public static void main(String[] args) {

                LinkedList<String> linkedList = new LinkedList<>();

                linkedList.add("First");

                linkedList.add("Second");

                linkedList.add("Third");

                linkedList.add("Fourth");

                System.out.println("LinkedList before reverse iteration:");

                for (String element : linkedList) {

                    System.out.println(element);

                }

                System.out.println("\nLinkedList in reverse order:");

                ListIterator<String> iterator = linkedList.listIterator(linkedList.size());

                while (iterator.hasPrevious()) {

                    System.out.println(iterator.previous());

                }
```

```
        }
}
```

Output:



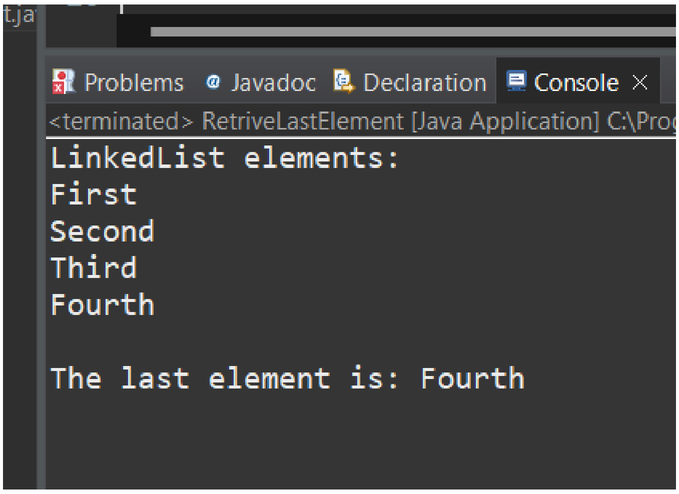
```
<terminated> Reverse [Java Application] C:\Program Files\Java\jc
LinkedList before reverse iteration:
First
Second
Third
Fourth

LinkedList in reverse order:
Fourth
Third
Second
First
```

7. Write a Java program to retrieve, but not remove, the last element of a linked list.

Code:

```
package lab5;

import java.util.LinkedList;


public class RetriveLastElement {

    public static void main(String[] args) {

        LinkedList<String> linkedList = new LinkedList<>();

        linkedList.add("First");

        linkedList.add("Second");

        linkedList.add("Third");

        linkedList.add("Fourth");

        System.out.println("LinkedList elements:");

        for (String element : linkedList) {

            System.out.println(element);

        }

        String lastElement = linkedList.peekLast();

        System.out.println("\nThe last element is: " + lastElement);
    }
}
```

Output:



<terminated> RetriveLastElement [Java Application] C:\Prog

```
LinkedList elements:
First
Second
Third
Fourth

The last element is: Fourth
```

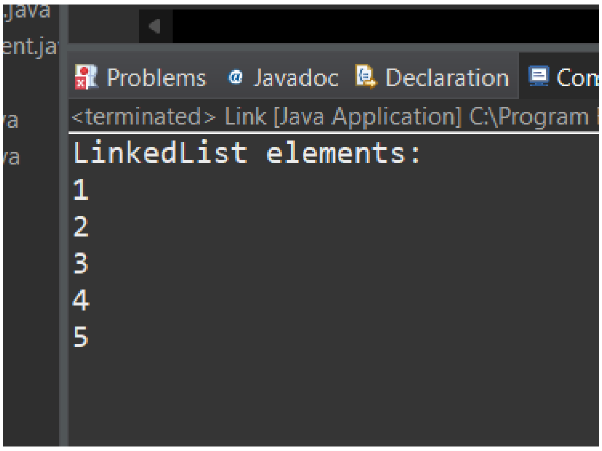
8. Write a Java program to create a LinkedList of integers and print all the elements.

Code:

```
package lab5;
import java.util.LinkedList;

public class Link {
    public static void main(String[] args) {
        LinkedList<Integer> linkedList = new LinkedList<>();
        linkedList.add(1);
        linkedList.add(2);
        linkedList.add(3);
        linkedList.add(4);
        linkedList.add(5);
        System.out.println("LinkedList elements:");
        for (Integer element : linkedList) {
            System.out.println(element);
        }
    }
```

}

Output:



<terminated> Link [Java Application] C:\Program

```
LinkedList elements:
1
2
3
4
5
```