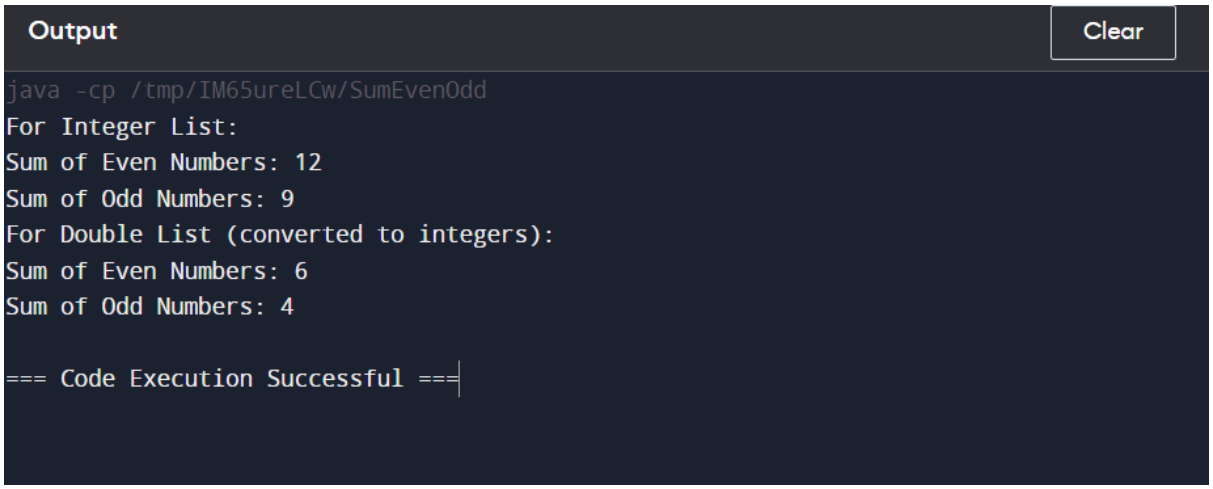
1. Java Program to Create a Generic Method that Takes a List of Numbers and Returns the Sum of All the Even and Odd Numbers

CODE:

```
import java.util.List;


public class SumEvenOdd {


   public static <T extends Number> void sumEvenOdd(List<T> numbers) {
      int sumEven = 0;
      int sumOdd = 0;


      for (T number : numbers) {
         int intValue = number.intValue();
         if (intValue % 2 == 0) {
            sumEven += intValue;
         } else {
            sumOdd += intValue;
         }
      }


      System.out.println("Sum of Even Numbers: " + sumEven);
      System.out.println("Sum of Odd Numbers: " + sumOdd);
   }


   public static void main(String[] args) {
      List<Integer> intList = List.of(1, 2, 3, 4, 5, 6);
      List<Double> doubleList = List.of(1.5, 2.0, 3.5, 4.0);


      System.out.println("For Integer List:");
      sumEvenOdd(intList);
```

```
        System.out.println("For Double List (converted to integers):");
        sumEvenOdd(doubleList);
    }
}
```

OUTPUT:



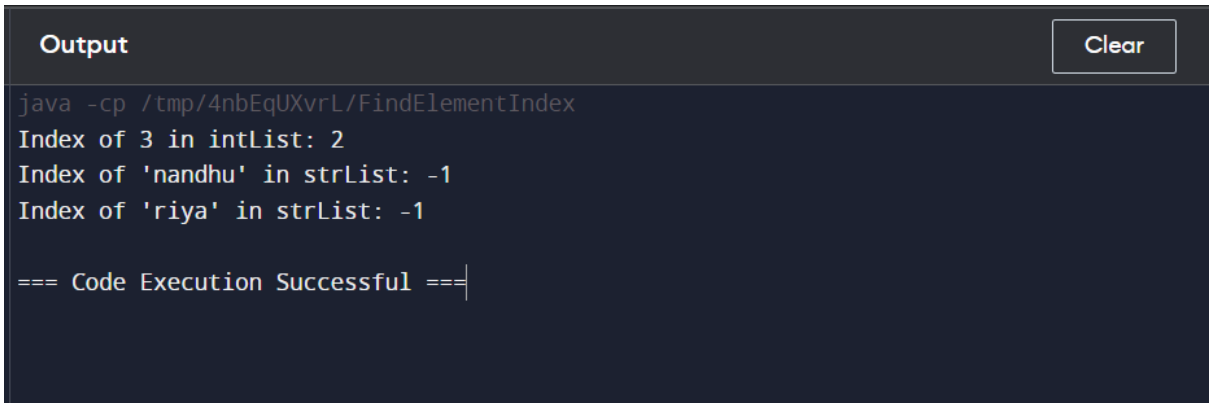
2. Java Program to Create a Generic Method that Takes a List of Any Type and a Target Element, Returns the Index of the First Occurrence of the Target Element

CODE:

```
import java.util.List;


public class FindElementIndex {


    public static <T> int findFirstOccurrence(List<T> list, T target) {
        for (int i = 0; i < list.size(); i++) {
            if (list.get(i).equals(target)) {
                return i;
            }
        }
        return -1;
    }


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

```
        List<Integer> intList = List.of(1, 2, 3, 4, 5);
        List<String> strList = List.of("dharu", "nandhu", "riya");

        System.out.println("Index of 3 in intList: " + findFirstOccurrence(intList, 3));
        System.out.println("Index of 'nandhu' in strList: " + findFirstOccurrence(strList, "banana"));
        System.out.println("Index of 'riya' in strList: " + findFirstOccurrence(strList, "grape"));
    }
}
```

OUTPUT:



3.

```
class Box<T> {
    private T item;
    public <U extends T> void setItem(U item) {
        this.item = item;
    }
    public <U extends T> U getItem() {
        return (U) item;
    }
}
public class GenericExample {
    public static void main(String[] args) {
        Box<Integer> intBox = new Box<>();
        intBox.setItem(123);
```

```
        System.out.println("Integer Value: " + intBox.<Integer>getItem());
        Box<String> strBox = new Box<>();
        strBox.setItem("Hello dharu");
        System.out.println("String Value: " + strBox.<String>getItem());
    }
}
```