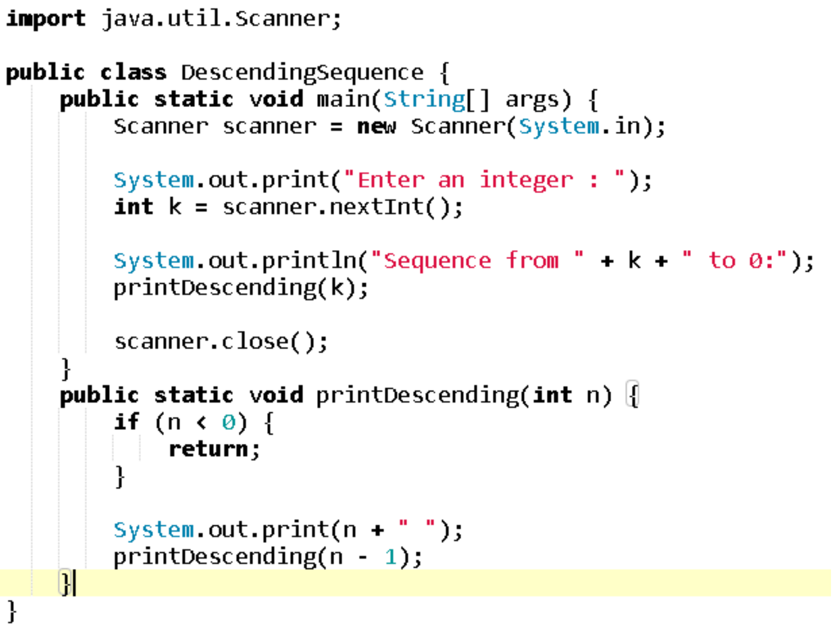
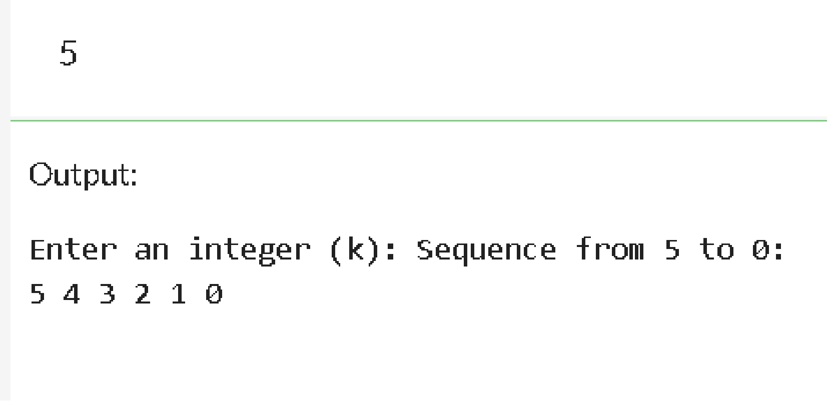
LAB # 3

LAB TASKS

1. Write a program which takes an integer value (k) as input and prints the sequence of numbers from k to 0 in descending order.



```
import java.util.Scanner;

public class DescendingSequence {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter an integer : ");
        int k = scanner.nextInt();

        System.out.println("Sequence from " + k + " to 0:");
        printDescending(k);

        scanner.close();
    }
    public static void printDescending(int n) {
        if (n < 0) {
            return;
        }

        System.out.print(n + " ");
        printDescending(n - 1);
    }
}
```

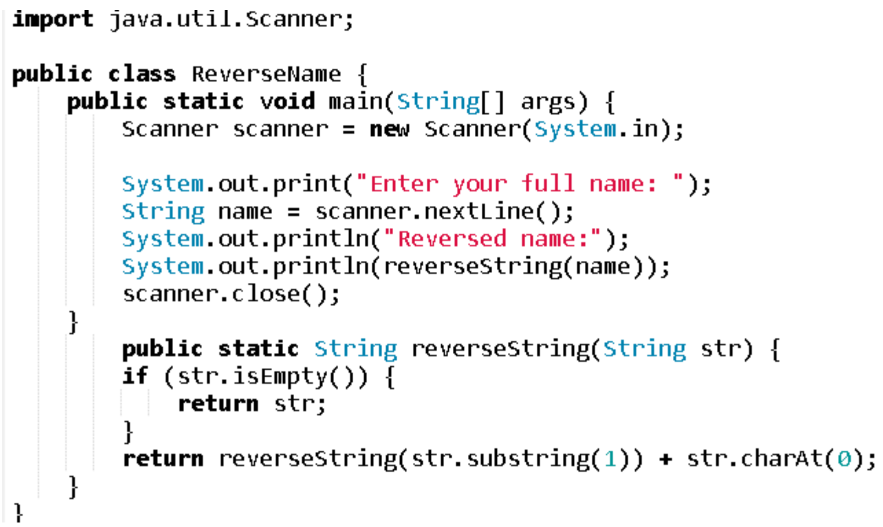


```
5
```

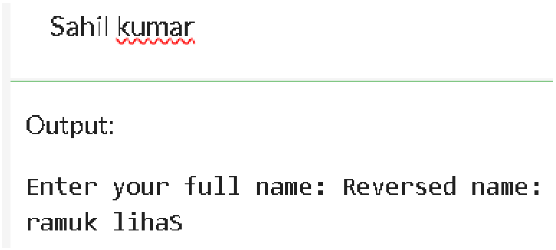
Output:

```
Enter an integer (k): Sequence from 5 to 0:
5 4 3 2 1 0
```

2. Write a program to reverse your full name using Recursion

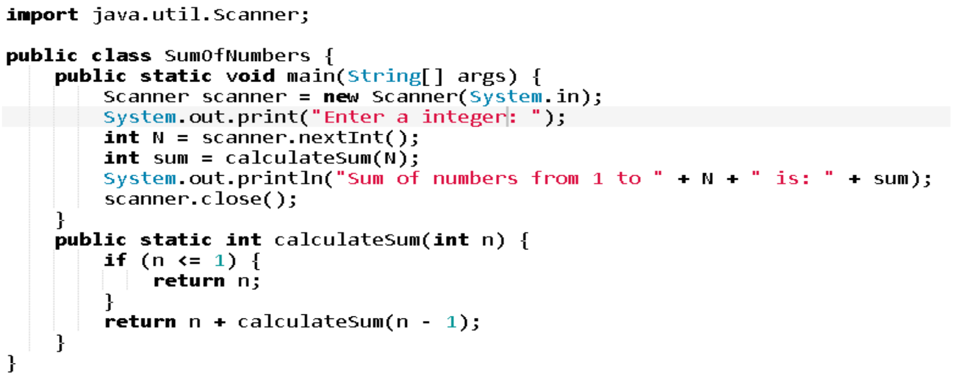


```java
import java.util.Scanner;

public class ReverseName {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter your full name: ");
        String name = scanner.nextLine();
        System.out.println("Reversed name:");
        System.out.println(reverseString(name));
        scanner.close();
    }
        public static String reverseString(String str) {
        if (str.isEmpty()) {
            return str;
        }
        return reverseString(str.substring(1)) + str.charAt(0);
    }
}
```

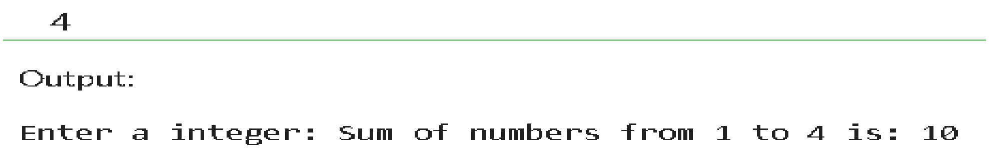


```
Sahil kumar

Output:

Enter your full name: Reversed name:
ramuk lihaS
```

3. Write a program to calculate the sum of numbers from 1 to N using recursion. N should be user input. .

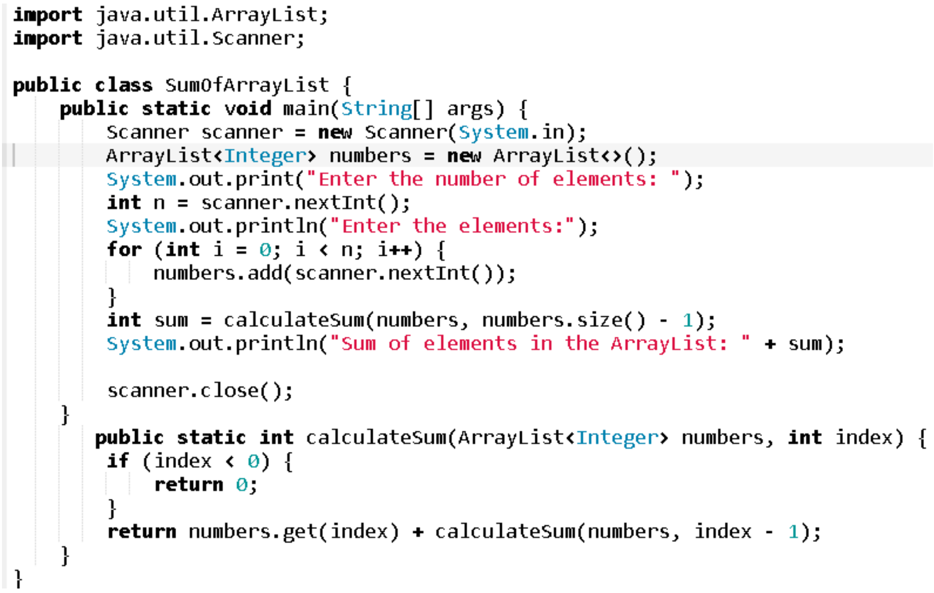


```java
import java.util.Scanner;

public class SumOfNumbers {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a integer: ");
        int N = scanner.nextInt();
        int sum = calculateSum(N);
        System.out.println("Sum of numbers from 1 to " + N + " is: " + sum);
        scanner.close();
    }
    public static int calculateSum(int n) {
        if (n <= 1) {
            return n;
        }
        return n + calculateSum(n - 1);
    }
}
```



```
4

Output:

Enter a integer: Sum of numbers from 1 to 4 is: 10
```

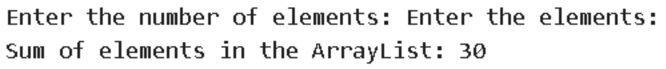
4. Write a recursive program to calculate the sum of elements in an arraylist.



```
import java.util.ArrayList;
import java.util.Scanner;

public class SumOfArrayList {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        ArrayList<Integer> numbers = new ArrayList<>();
        System.out.print("Enter the number of elements: ");
        int n = scanner.nextInt();
        System.out.println("Enter the elements:");
        for (int i = 0; i < n; i++) {
            numbers.add(scanner.nextInt());
        }
        int sum = calculateSum(numbers, numbers.size() - 1);
        System.out.println("Sum of elements in the ArrayList: " + sum);

        scanner.close();
    }
       public static int calculateSum(ArrayList<Integer> numbers, int index) {
        if (index < 0) {
            return 0;
        }
        return numbers.get(index) + calculateSum(numbers, index - 1);
    }
}
```

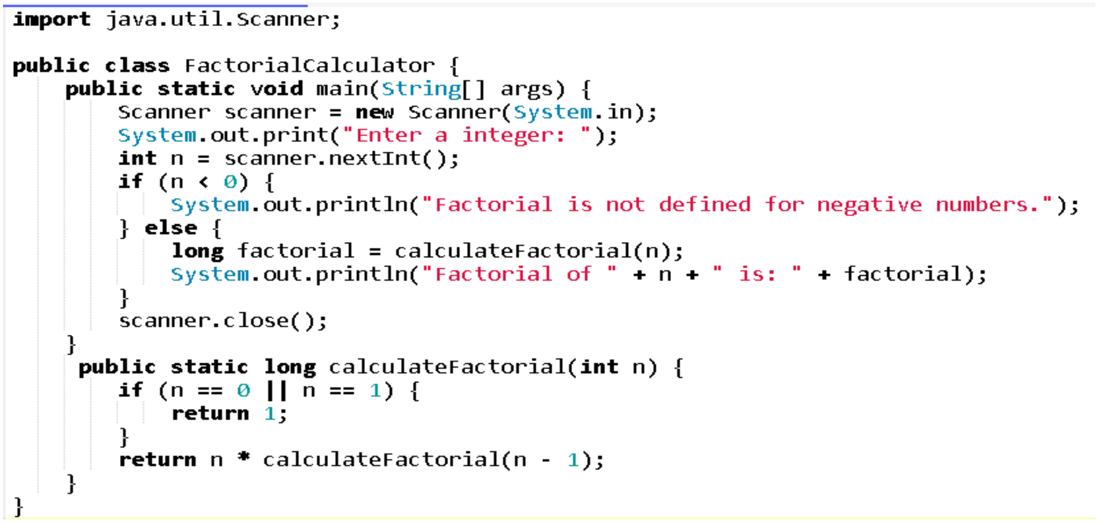
```
5
5
5
```

Output:

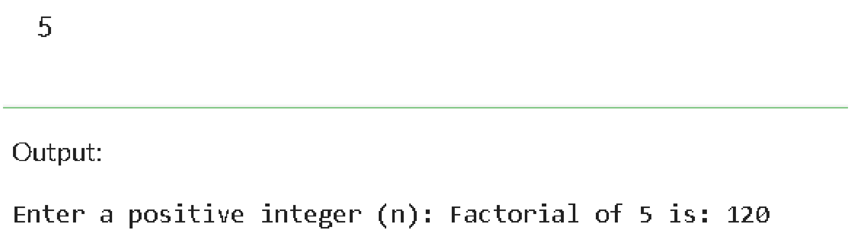


```
Enter the number of elements: Enter the elements:
Sum of elements in the ArrayList: 30
```

5. Write a recursive program to calculate the factorial of a given integer n



```
import java.util.Scanner;

public class FactorialCalculator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a integer: ");
        int n = scanner.nextInt();
        if (n < 0) {
            System.out.println("Factorial is not defined for negative numbers.");
        } else {
            long factorial = calculateFactorial(n);
            System.out.println("Factorial of " + n + " is: " + factorial);
        }
        scanner.close();
    }
     public static long calculateFactorial(int n) {
        if (n == 0 || n == 1) {
            return 1;
        }
        return n * calculateFactorial(n - 1);
    }
}
```

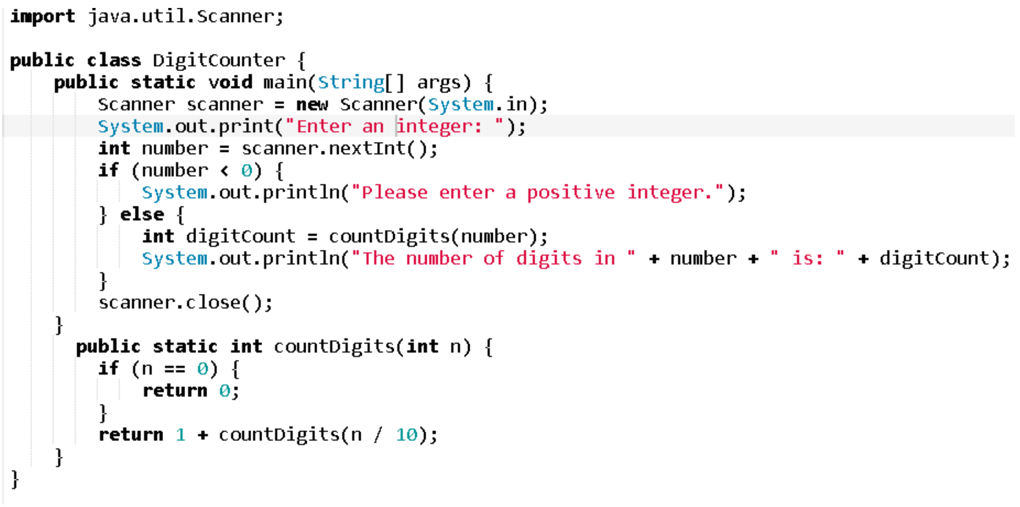


```
5
```

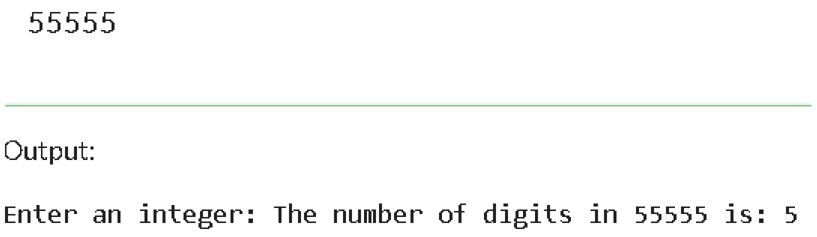
Output:

```
Enter a positive integer (n): Factorial of 5 is: 120
```

6. Write a program to count the digits of a given number using recursion.



```java
import java.util.Scanner;

public class DigitCounter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        int number = scanner.nextInt();
        if (number < 0) {
            System.out.println("Please enter a positive integer.");
        } else {
            int digitCount = countDigits(number);
            System.out.println("The number of digits in " + number + " is: " + digitCount);
        }
        scanner.close();
    }
      public static int countDigits(int n) {
        if (n == 0) {
            return 0;
        }
        return 1 + countDigits(n / 10);
    }
}
```



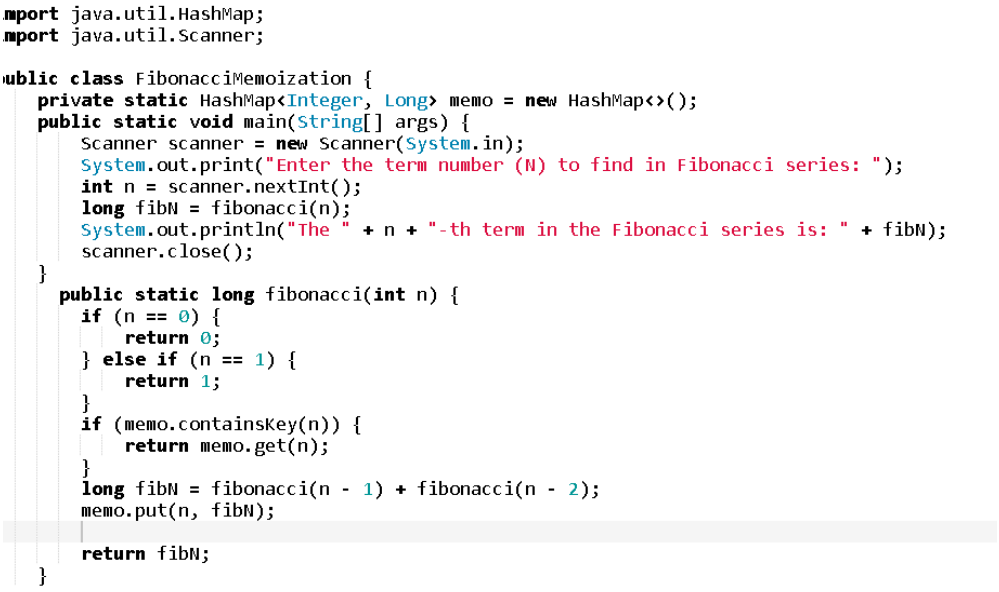
```
55555
```

Output:

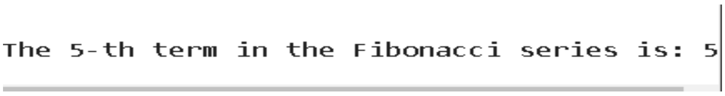
```
Enter an integer: The number of digits in 55555 is: 5
```

HOME TASK

1. Write a java program to find the N-th term in the Fibonacci series using Memoization.

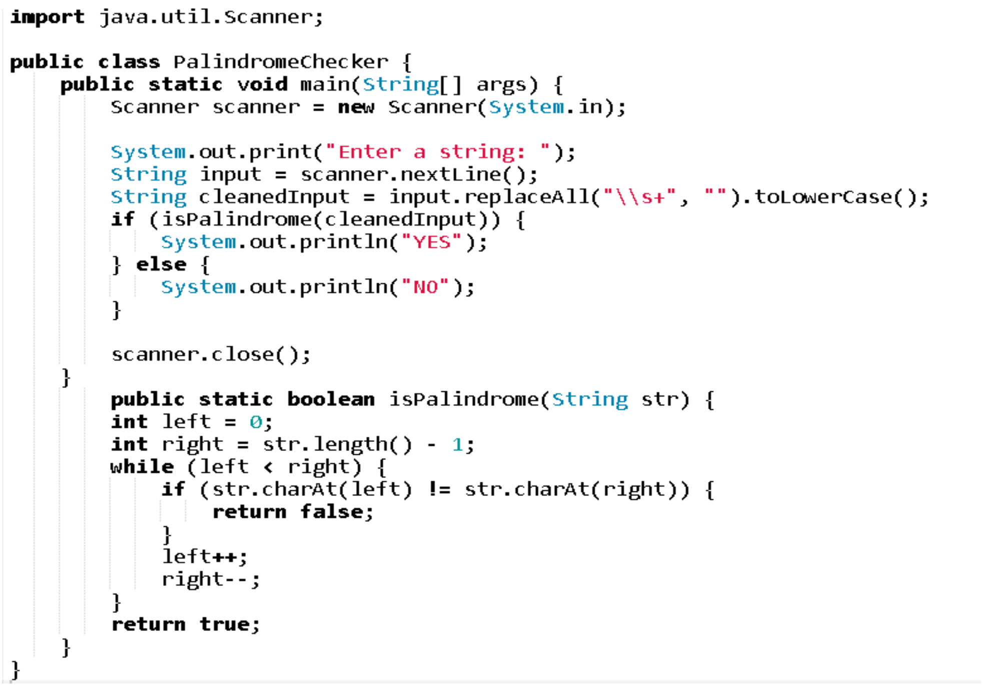


```
mport java.util.HashMap;
mport java.util.Scanner;

ublic class FibonacciMemoization {
    private static HashMap<Integer, Long> memo = new HashMap<>();
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the term number (N) to find in Fibonacci series: ");
        int n = scanner.nextInt();
        long fibN = fibonacci(n);
        System.out.println("The " + n + "-th term in the Fibonacci series is: " + fibN);
        scanner.close();
    }
      public static long fibonacci(int n) {
        if (n == 0) {
            return 0;
        } else if (n == 1) {
            return 1;
        }
        if (memo.containsKey(n)) {
            return memo.get(n);
        }
        long fibN = fibonacci(n - 1) + fibonacci(n - 2);
        memo.put(n, fibN);

        return fibN;
    }
```

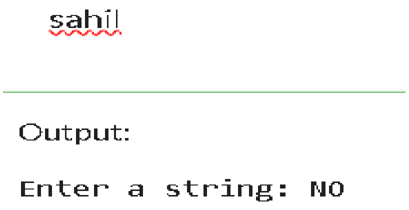


```
The 5-th term in the Fibonacci series is: 5
```

2. Write a java program to check whether a given string is a palindrome or not. A palindrome is a string that reads the same forwards and backwards.Print "YES" if the string is a palindrome, otherwise print "NO".



```
import java.util.Scanner;

public class PalindromeChecker {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a string: ");
        String input = scanner.nextLine();
        String cleanedInput = input.replaceAll("\\s+", "").toLowerCase();
        if (isPalindrome(cleanedInput)) {
            System.out.println("YES");
        } else {
            System.out.println("NO");
        }

        scanner.close();
    }
        public static boolean isPalindrome(String str) {
        int left = 0;
        int right = str.length() - 1;
        while (left < right) {
            if (str.charAt(left) != str.charAt(right)) {
                return false;
            }
            left++;
            right--;
        }
        return true;
    }
}
```

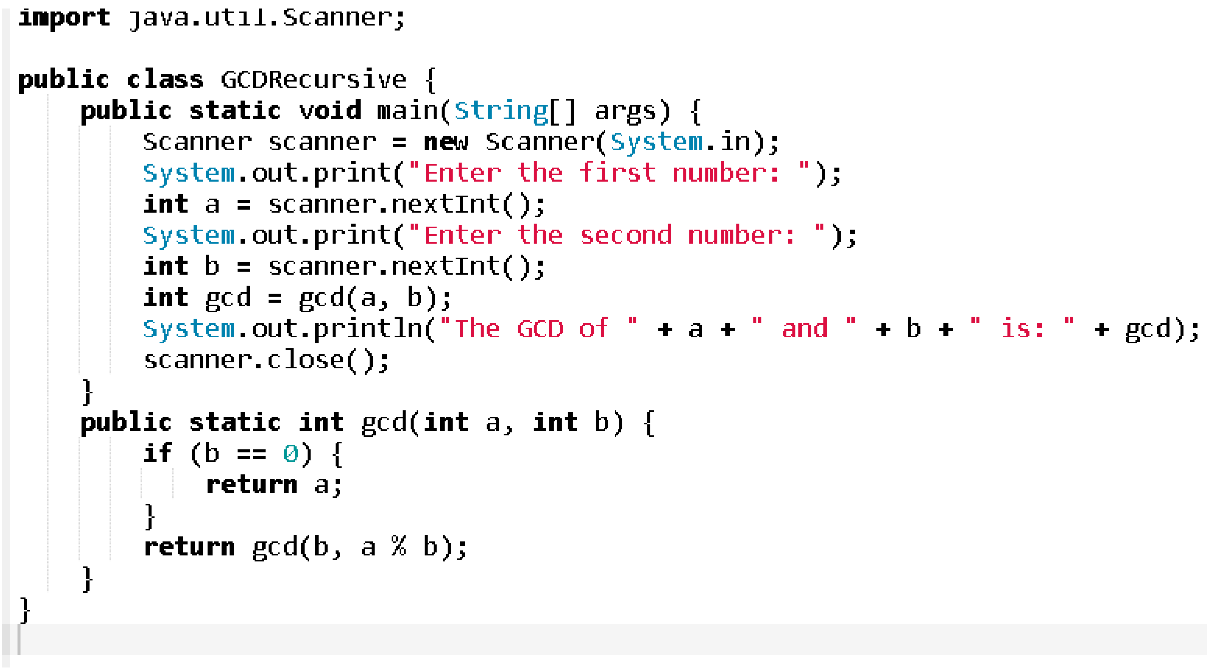


```
sahil
```

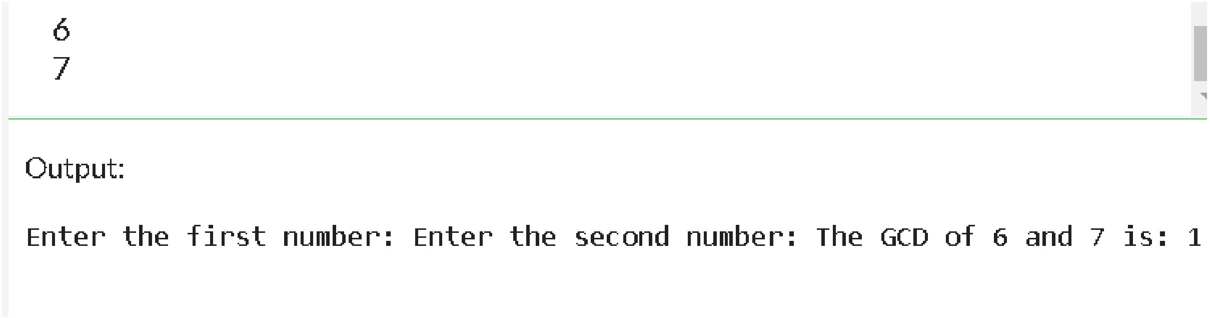
Output:

```
Enter a string: NO
```

3. Write a recursive program to find the greatest common divisor (GCD) of two numbers using Euclid's algorithm.



```java
import java.util.Scanner;

public class GCDRecursive {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the first number: ");
        int a = scanner.nextInt();
        System.out.print("Enter the second number: ");
        int b = scanner.nextInt();
        int gcd = gcd(a, b);
        System.out.println("The GCD of " + a + " and " + b + " is: " + gcd);
        scanner.close();
    }
    public static int gcd(int a, int b) {
        if (b == 0) {
            return a;
        }
        return gcd(b, a % b);
    }
}
```



```
6
7
```

Output:

```
Enter the first number: Enter the second number: The GCD of 6 and 7 is: 1
```