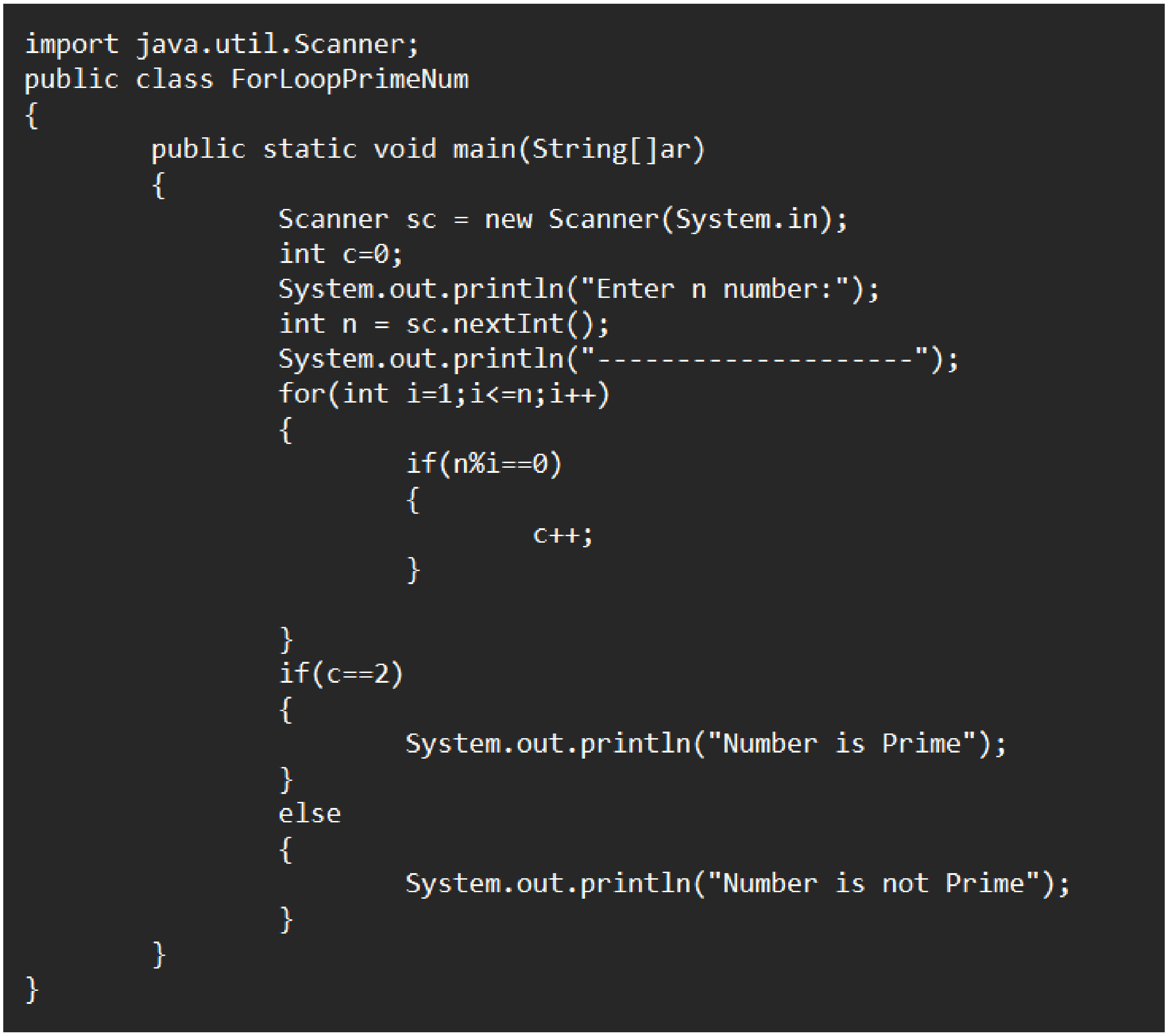
Assignment No:-5

Name:-Suryawanshi Sangramsingh Sambhaji

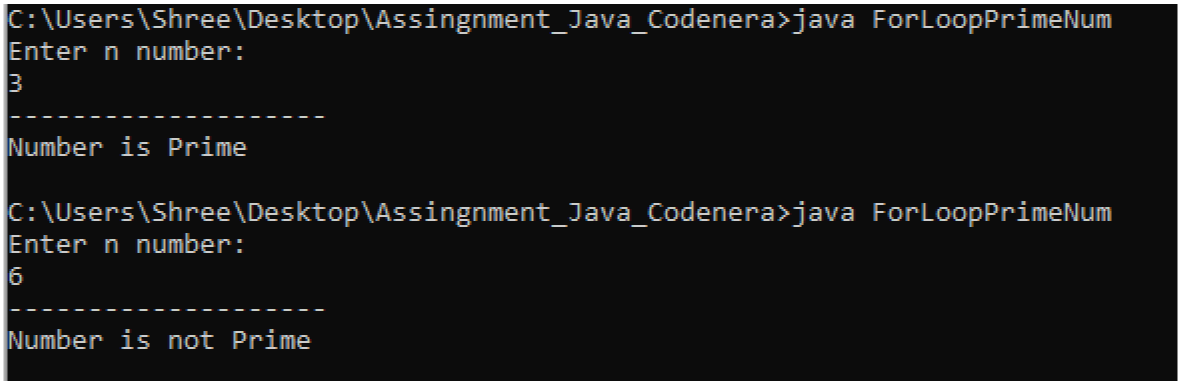
Batch:-April   Date:-6/5/2024

1. Write a program to check if a number enter from keyboard is prime or not.



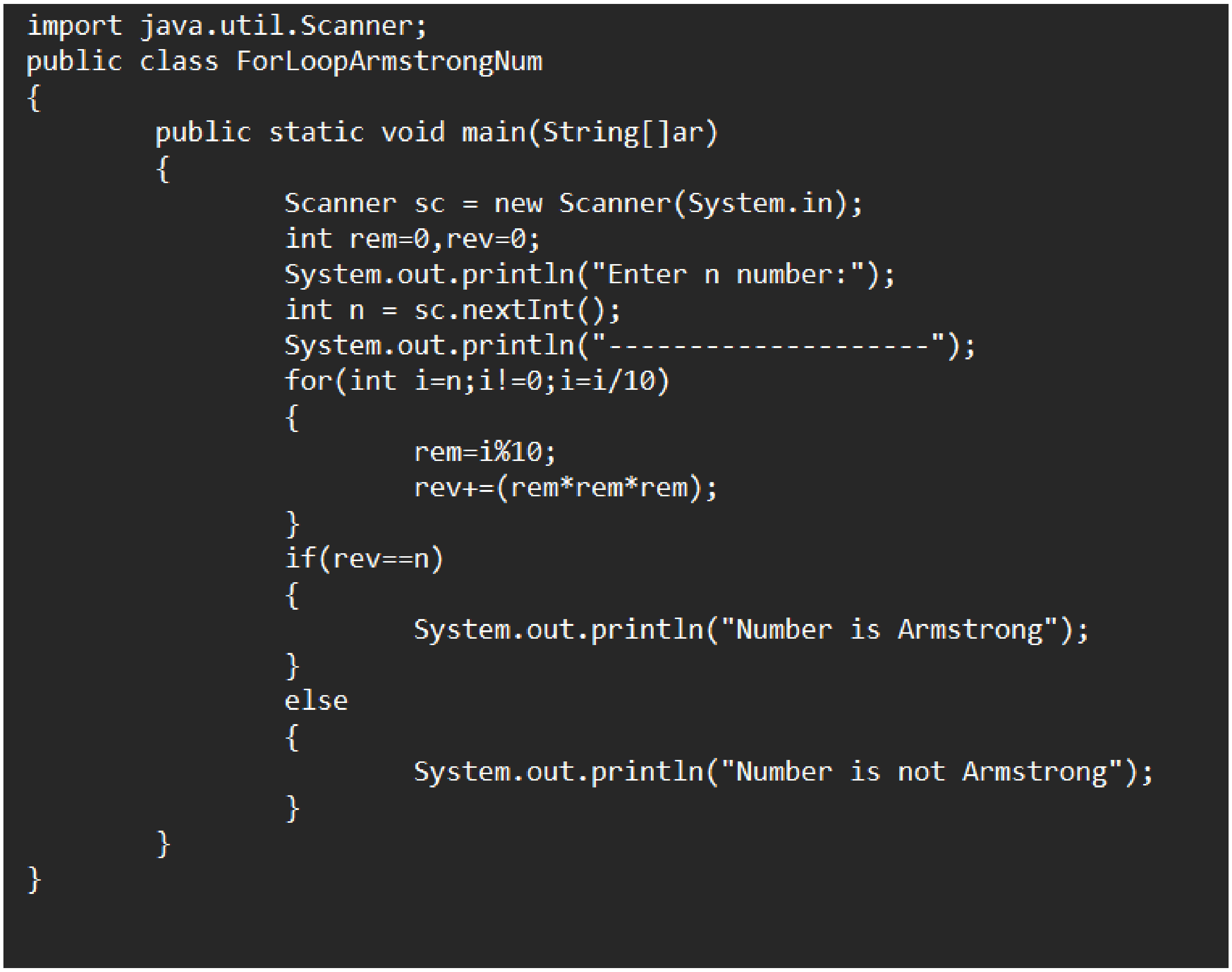
```
import java.util.Scanner;
public class ForLoopPrimeNum
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                int c=0;
                System.out.println("Enter n number:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                for(int i=1;i<=n;i++)
                {
                        if(n%i==0)
                        {
                                c++;
                        }

                }
                if(c==2)
                {
                        System.out.println("Number is Prime");
                }
                else
                {
                        System.out.println("Number is not Prime");
                }
        }
}
```

Output:



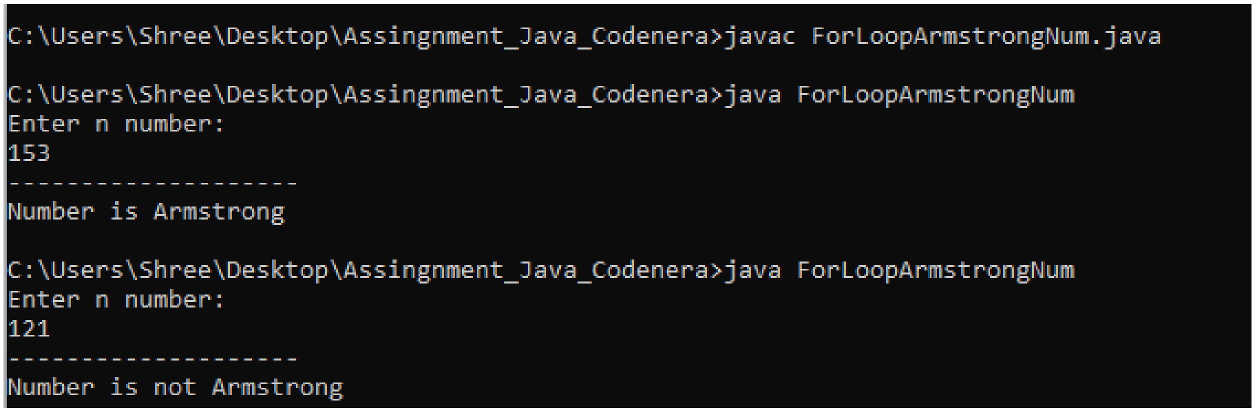
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopPrimeNum
Enter n number:
3
--------------------
Number is Prime

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopPrimeNum
Enter n number:
6
--------------------
Number is not Prime
```

2. Write a function to check if number is Armstrong of Not.



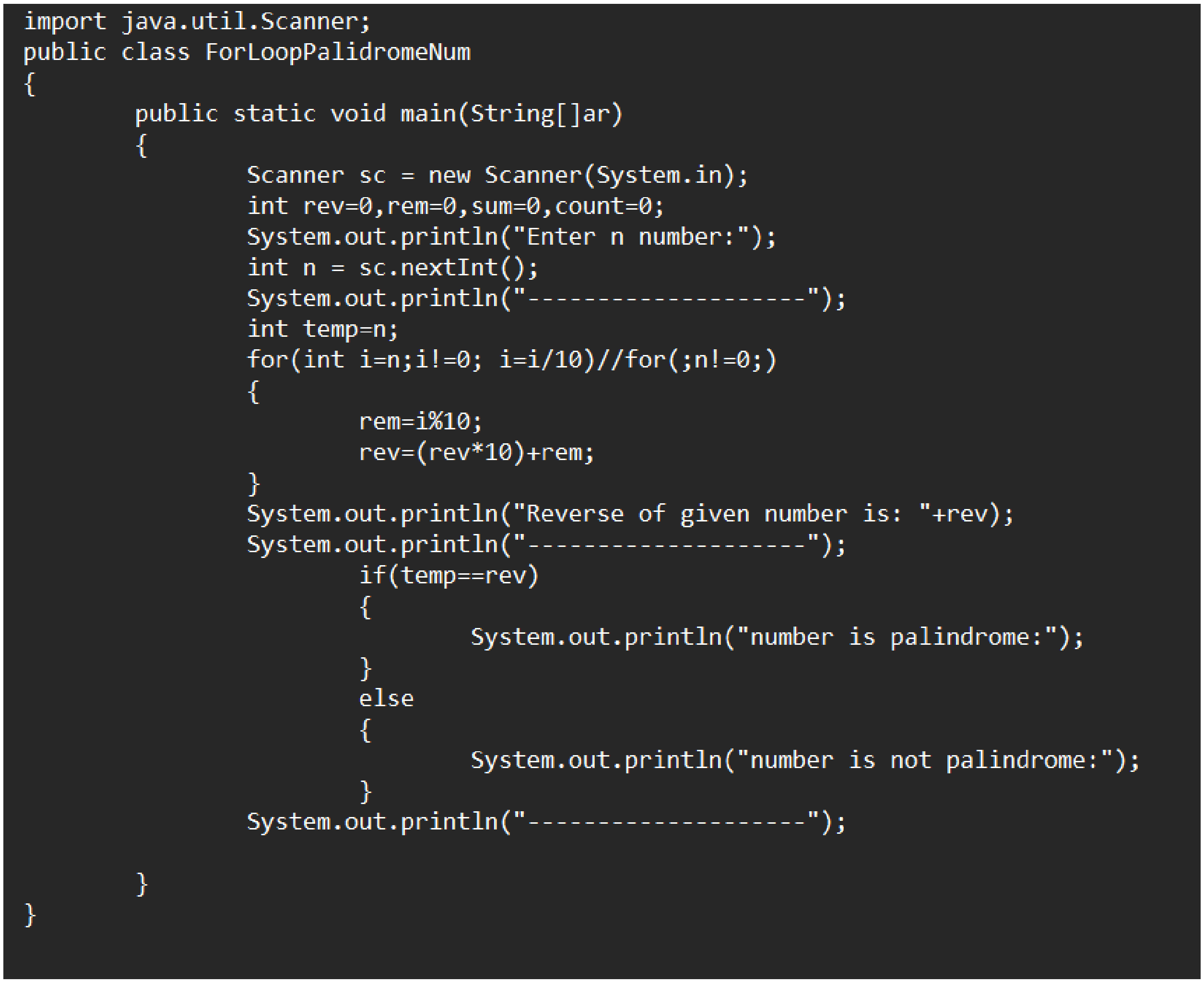
```
import java.util.Scanner;
public class ForLoopArmstrongNum
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                int rem=0,rev=0;
                System.out.println("Enter n number:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                for(int i=n;i!=0;i=i/10)
                {
                        rem=i%10;
                        rev+=(rem*rem*rem);
                }
                if(rev==n)
                {
                        System.out.println("Number is Armstrong");
                }
                else
                {
                        System.out.println("Number is not Armstrong");
                }
        }
}
```

Output:



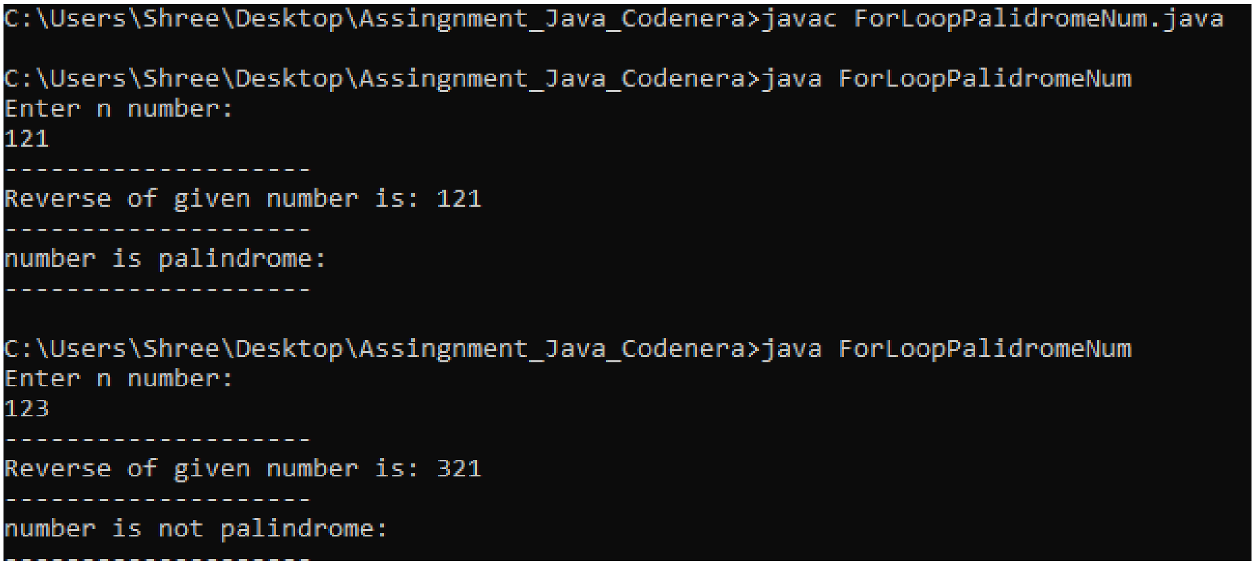
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>javac ForLoopArmstrongNum.java

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopArmstrongNum
Enter n number:
153
--------------------
Number is Armstrong

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopArmstrongNum
Enter n number:
121
--------------------
Number is not Armstrong
```

3. Write a java program to Check the number is Palindrome or not.



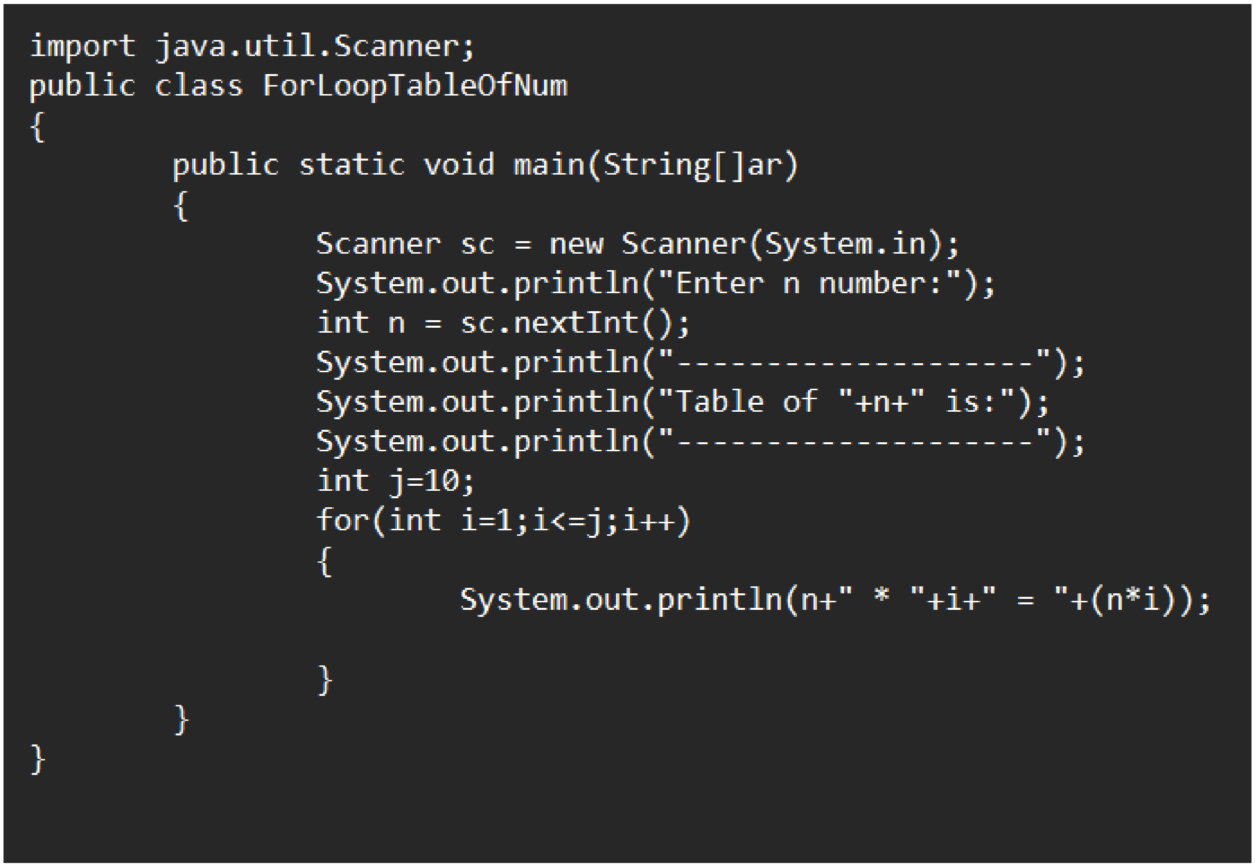
```
import java.util.Scanner;
public class ForLoopPalidromeNum
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                int rev=0,rem=0,sum=0,count=0;
                System.out.println("Enter n number:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                int temp=n;
                for(int i=n;i!=0; i=i/10)//for(;n!=0;)
                {
                        rem=i%10;
                        rev=(rev*10)+rem;
                }
                System.out.println("Reverse of given number is: "+rev);
                System.out.println("--------------------");
                        if(temp==rev)
                        {
                                System.out.println("number is palindrome:");
                        }
                        else
                        {
                                System.out.println("number is not palindrome:");
                        }
                System.out.println("--------------------");

        }
}
```

Output:



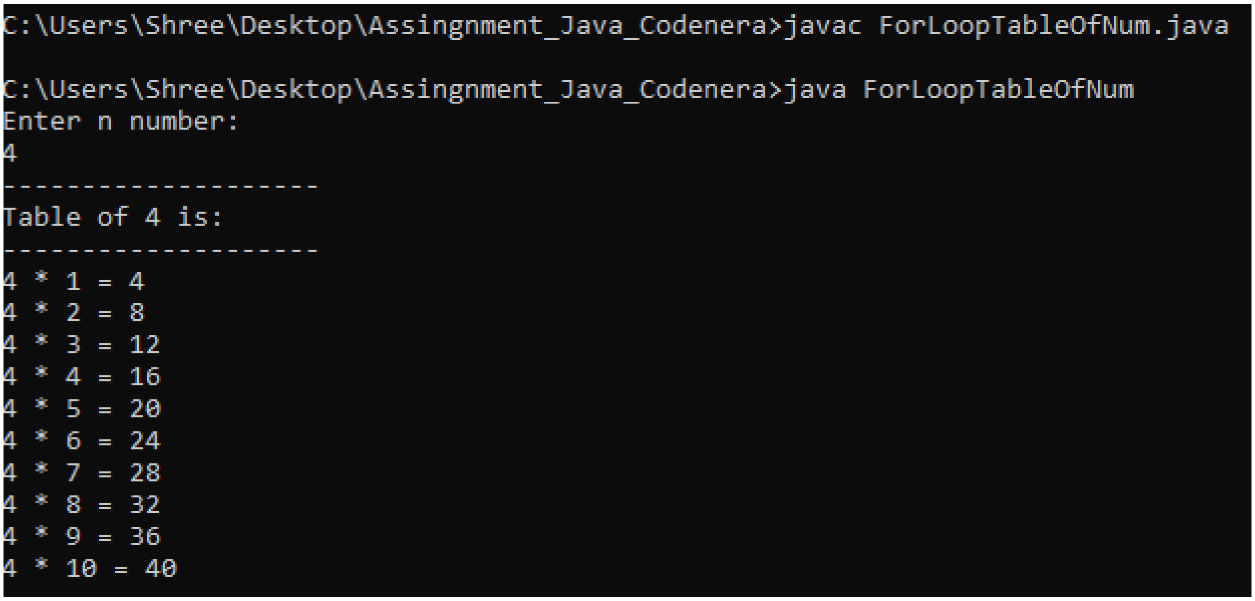
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>javac ForLoopPalidromeNum.java

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopPalidromeNum
Enter n number:
121
--------------------
Reverse of given number is: 121
--------------------
number is palindrome:
--------------------

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopPalidromeNum
Enter n number:
123
--------------------
Reverse of given number is: 321
--------------------
number is not palindrome:
--------------------
```

4. Write a program to enter a number and print he table.



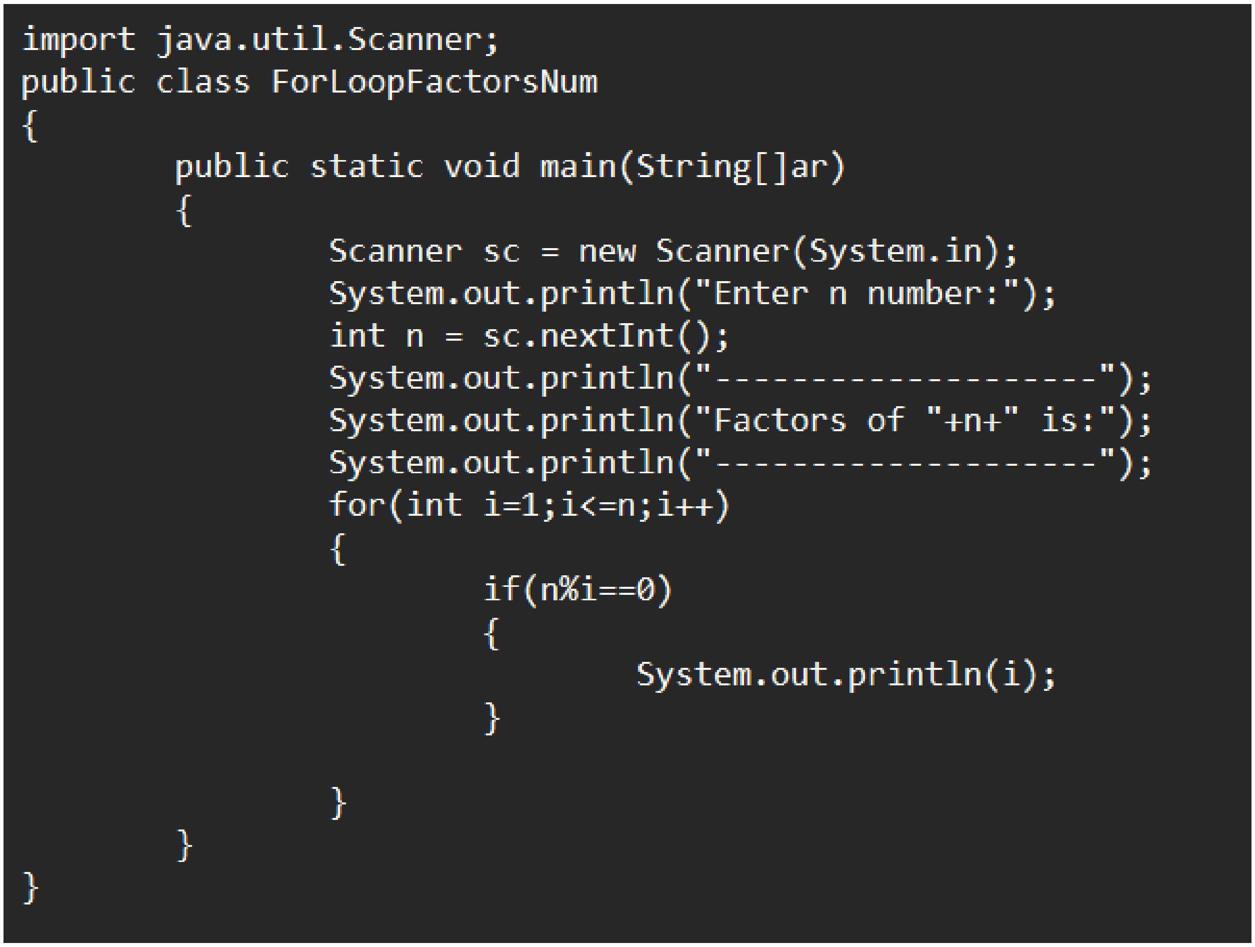
```java
import java.util.Scanner;
public class ForLoopTableOfNum
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter n number:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                System.out.println("Table of "+n+" is:");
                System.out.println("--------------------");
                int j=10;
                for(int i=1;i<=j;i++)
                {
                        System.out.println(n+" * "+i+" = "+(n*i));

                }
        }
}
```

Output:



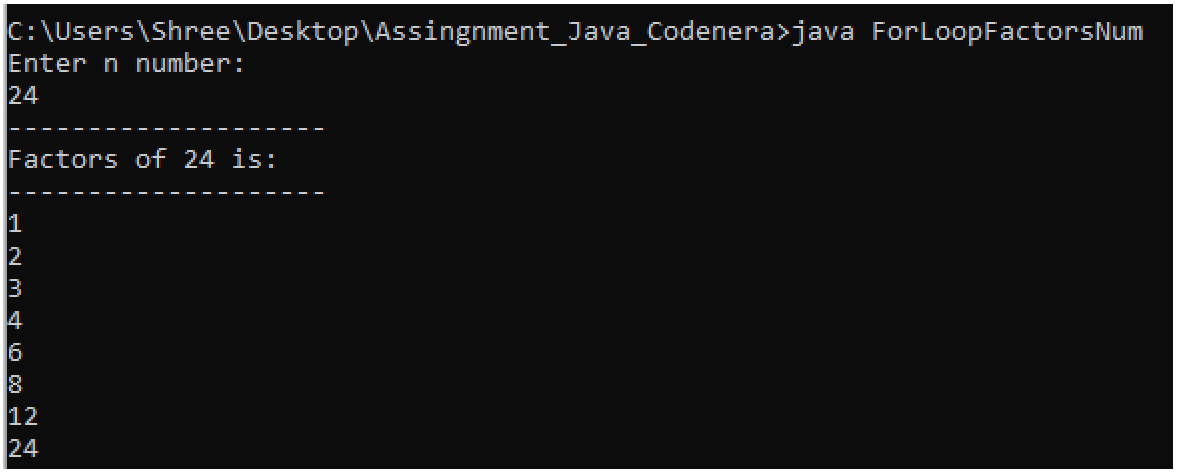
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>javac ForLoopTableOfNum.java

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopTableOfNum
Enter n number:
4
--------------------
Table of 4 is:
--------------------
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
4 * 10 = 40
```

5. Write a program to print the factors of the given number.



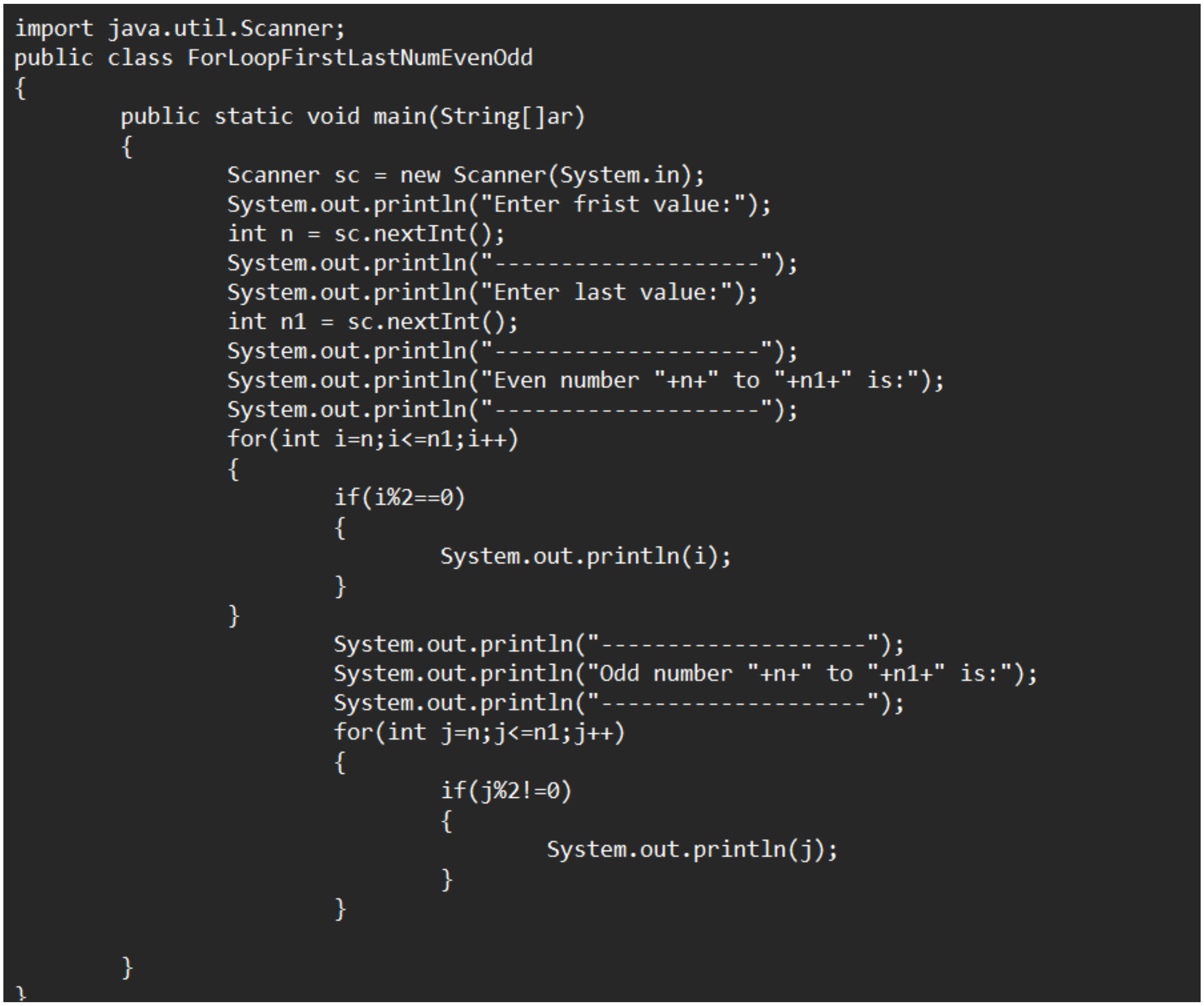
```java
import java.util.Scanner;
public class ForLoopFactorsNum
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter n number:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                System.out.println("Factors of "+n+" is:");
                System.out.println("--------------------");
                for(int i=1;i<=n;i++)
                {
                        if(n%i==0)
                        {
                                System.out.println(i);
                        }

                }
        }
}
```

Output:



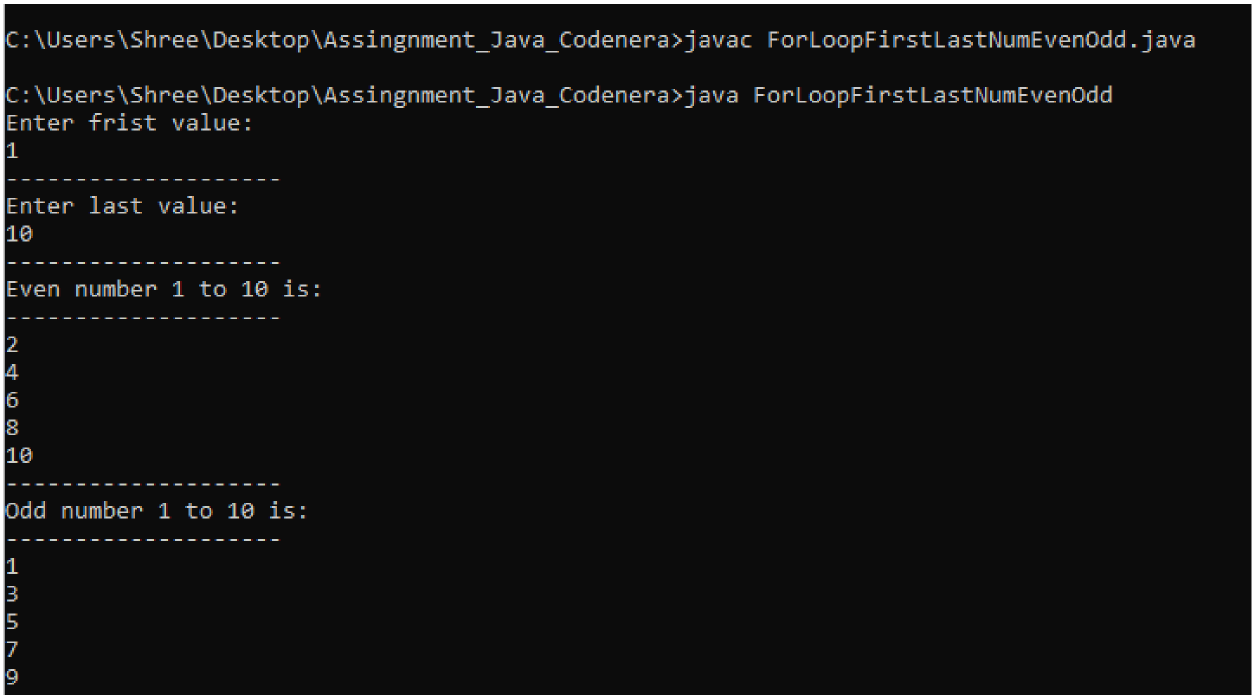
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopFactorsNum
Enter n number:
24
--------------------
Factors of 24 is:
--------------------
1
2
3
4
6
8
12
24
```

6. Write a program print even, and odd number from given series from last to start (means you need to take start value and last value, and then print even and odd numbers separately from end to first).



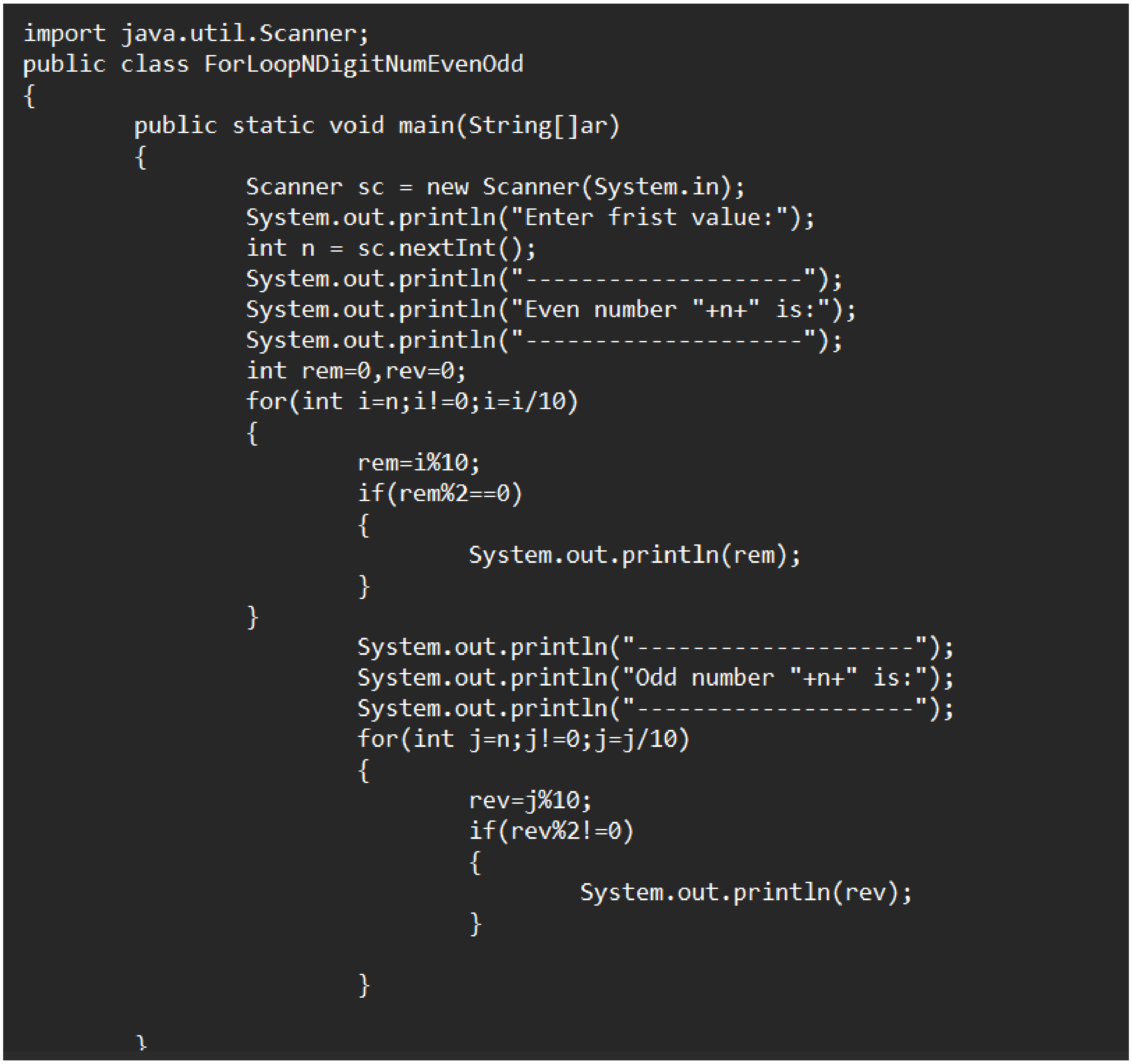
```
import java.util.Scanner;
public class ForLoopFirstLastNumEvenOdd
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter frist value:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                System.out.println("Enter last value:");
                int n1 = sc.nextInt();
                System.out.println("--------------------");
                System.out.println("Even number "+n+" to "+n1+" is:");
                System.out.println("--------------------");
                for(int i=n;i<=n1;i++)
                {
                        if(i%2==0)
                        {
                                System.out.println(i);
                        }
                }
                        System.out.println("--------------------");
                        System.out.println("Odd number "+n+" to "+n1+" is:");
                        System.out.println("--------------------");
                        for(int j=n;j<=n1;j++)
                        {
                                if(j%2!=0)
                                {
                                        System.out.println(j);
                                }
                        }

        }
}
```

Output:



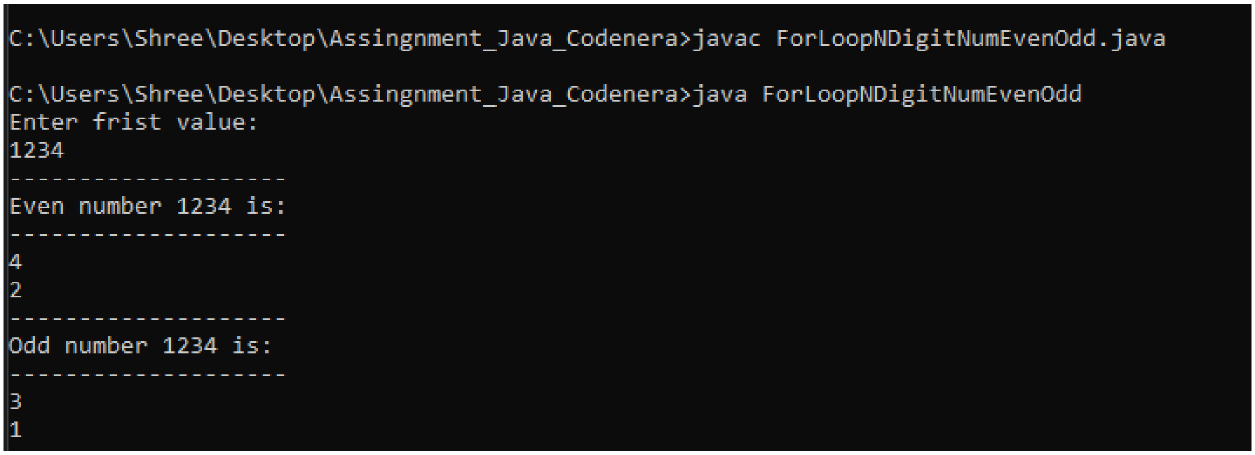
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>javac ForLoopFirstLastNumEvenOdd.java

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopFirstLastNumEvenOdd
Enter frist value:
1
--------------------
Enter last value:
10
--------------------
Even number 1 to 10 is:
--------------------
2
4
6
8
10
--------------------
Odd number 1 to 10 is:
--------------------
1
3
5
7
9
```

7. Write a program to take an n digit number, and print even and odd numbers from that number separately by breaking that number.



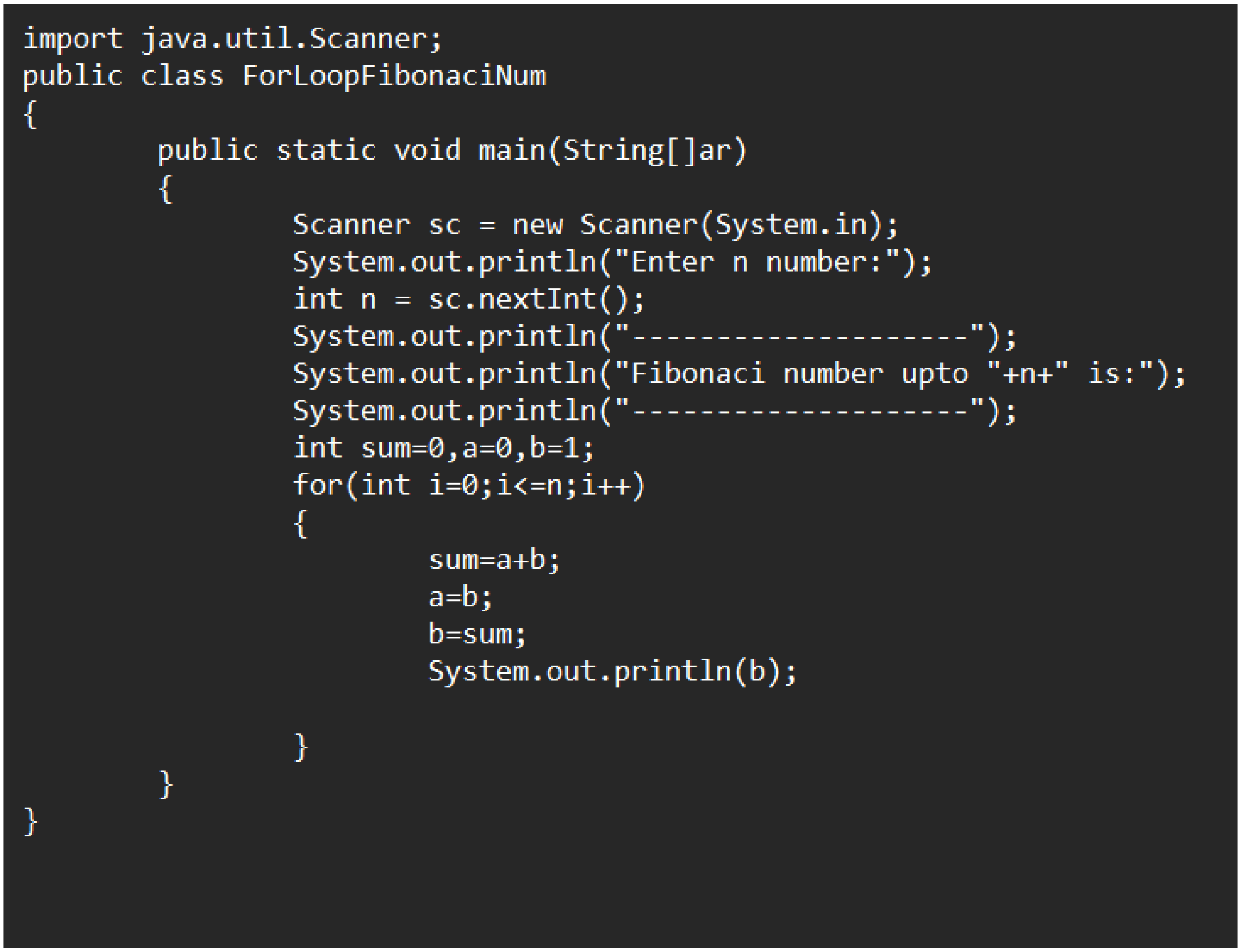
```
import java.util.Scanner;
public class ForLoopNDigitNumEvenOdd
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter frist value:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                System.out.println("Even number "+n+" is:");
                System.out.println("--------------------");
                int rem=0,rev=0;
                for(int i=n;i!=0;i=i/10)
                {
                        rem=i%10;
                        if(rem%2==0)
                        {
                                System.out.println(rem);
                        }
                }
                        System.out.println("--------------------");
                        System.out.println("Odd number "+n+" is:");
                        System.out.println("--------------------");
                        for(int j=n;j!=0;j=j/10)
                        {
                                rev=j%10;
                                if(rev%2!=0)
                                {
                                        System.out.println(rev);
                                }

                        }

        }
```

Output:



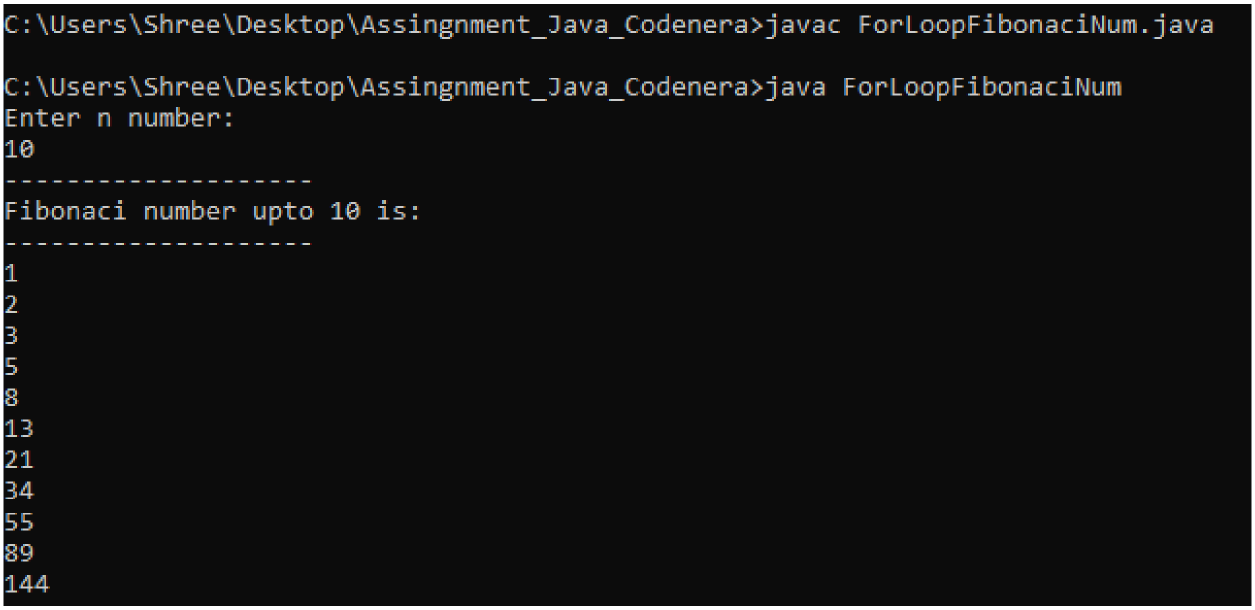
```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>javac ForLoopNDigitNumEvenOdd.java

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopNDigitNumEvenOdd
Enter frist value:
1234
--------------------
Even number 1234 is:
--------------------
4
2
--------------------
Odd number 1234 is:
--------------------
3
1
```

8. Write a java program to print the Fibonacci series.



```
import java.util.Scanner;
public class ForLoopFibonaciNum
{
        public static void main(String[]ar)
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter n number:");
                int n = sc.nextInt();
                System.out.println("--------------------");
                System.out.println("Fibonaci number upto "+n+" is:");
                System.out.println("--------------------");
                int sum=0,a=0,b=1;
                for(int i=0;i<=n;i++)
                {
                        sum=a+b;
                        a=b;
                        b=sum;
                        System.out.println(b);

                }
        }
}
```

Output:



```
C:\Users\Shree\Desktop\Assingnment_Java_Codenera>javac ForLoopFibonaciNum.java

C:\Users\Shree\Desktop\Assingnment_Java_Codenera>java ForLoopFibonaciNum
Enter n number:
10
--------------------
Fibonaci number upto 10 is:
--------------------
1
2
3
5
8
13
21
34
55
89
144
```

Theory:

Q1. Difference between JDK, JRE and JVM.

Answer: JVM: - is a virtual machine. That does not physically exists.it can also run those program which is written in other language and compile in byte code. JVM loads and execute the .class file. JVM is platform dependent.

JDK: - JRE + development tools.

JDK is an implementation of any one of the below given Java

1) Standard Edition 2) Enterprise Edition Java Platform 3) Micro Edition Java Platform

The JDK contains a private Java Virtual Machine (JVM) and a few other resources such as

An interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator

(Javadoc) to complete the development of a Java Application.

JRE: - JVM**+** set of libraries + other files. It is physically exists. It is used to provide the runtime environment. It is the implementation of JVM.

It contains a set of libraries + other files that JVM uses at runtime.

Q2.What is Class Loader Subsystem. Explain.

Answer: -Class loader subsystem: .class file is pass to class loader subsystem

I) loading: it loads the class.

1. Bootstrap Class Loader: This is the first class loader which is the super class of

Extension class loader. It loads the rt.jar file

2. Extension Class Loader: This is the child class loader of Bootstrap and parent

Class loader of System class loader. It loads the jar files in lib folder

3. Application Class Loader: This is the child class loader of Extension

Class loader. It loads the class files from environment variable.

II) Linking: It divided into three parts.

1. Verify: it check the byte code. Weather byte code is correct or not .if the byte code is not correct then it gives runtime error.

2. Prepare: it used to assign the default values to static variable and methods.

3. Resolve: All the symbolic representation get its default value.

III) Initialization: it is used to gets actual value of static variable and here in this step all static method and block will be executed.

Q3.What are the memory areas in java, explain one by one.

Answer: - 1) Method Area: All the class level data stored in method area.

2) Heap Area: All the object are stored in Heap area.

3) Stack Area: All the thread information is stored in this area.

4) Program Counter Register: all the threads are separated and give instruction of the thread based on thread scheduler. If that thread execution is completed then next instruction will be executed.

5) Native Method Stack: It contains all the native methods used in the application

Q4. What is execution engine? Explain

Answer: - execution engine is execute the code.

1. Interpreter: Read bytecode stream then execute the interpret the byte code line by line if it found the repeated code it passes to JIT compiler.

2. Just-In-Time (JIT) compiler: It is used to improve the performance, it compile the code at the time of compilation and it convert to the native code.

3. Garbage Collector: it used to remove unused and unwanted object to clean the memory System.gc() method is used.

Q5. What are the default values and size of all datatypes?

Answer: -

1. Boolean:

Default Value: false

Size: 1 bit

2. Byte:

Default Value: 0

Size: 1 byte

3. Short:

Default Value: 0

Size: 2 bytes

4. Int:

Default Value: 0

Size: 4 bytes

5. Long:

Default Value: 0L

Size: 8 bytes

6. Float:

Default Value: 0.0f

Size: 4 bytes

7. Double:

Default Value: 0.0d

Size: 8 bytes

8. Char:

Default Value: \u0000

Size: 2 bytes