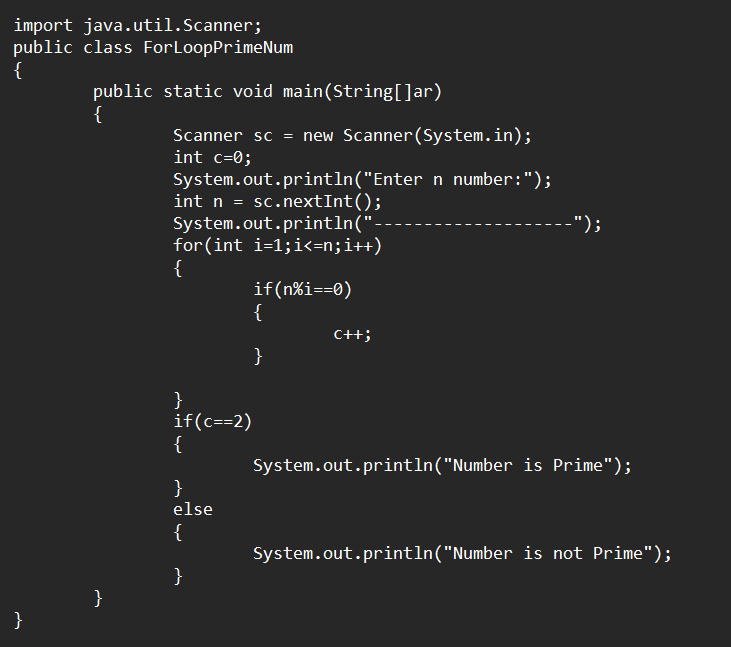
**Assignment No:-5**

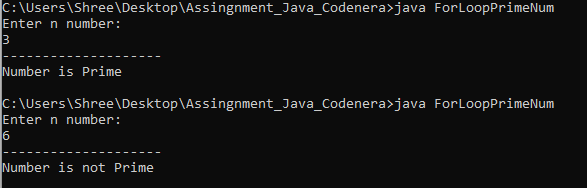
Name:-Suryawanshi Sangramsingh Sambhaji

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

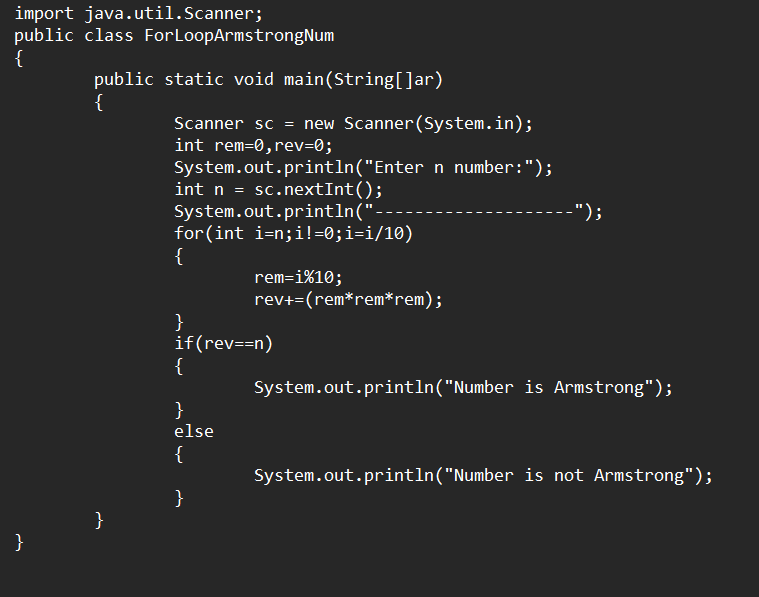
**1. Write a program to check if a number enter from keyboard is prime or not.**

****

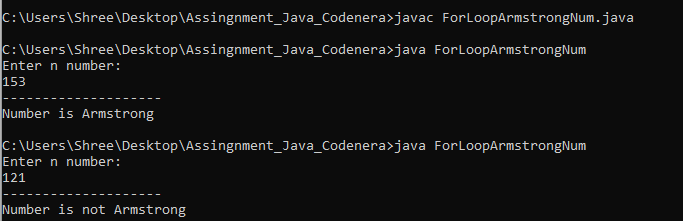
**Output:**



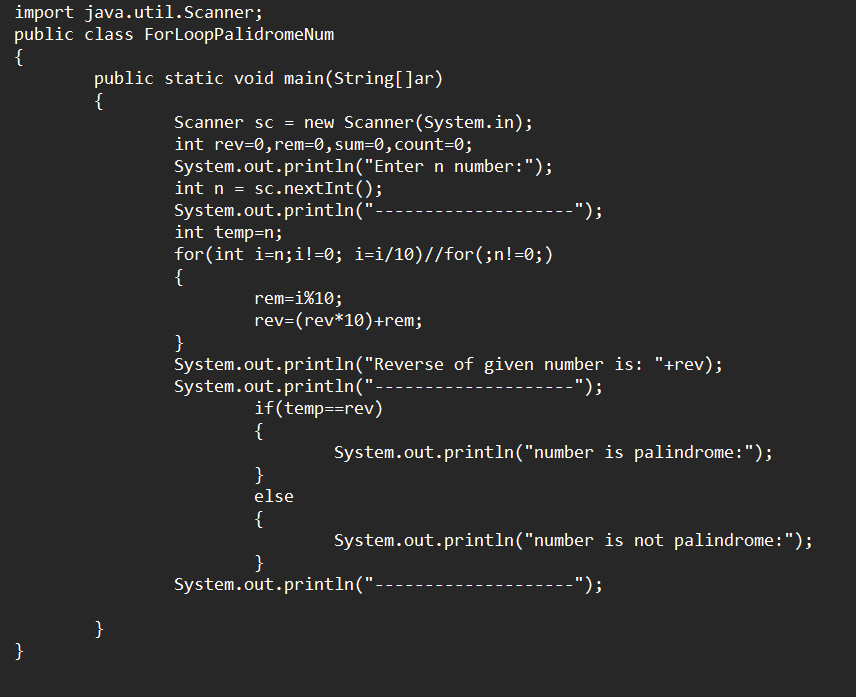
**2. Write a function to check if number is Armstrong of Not.**

****

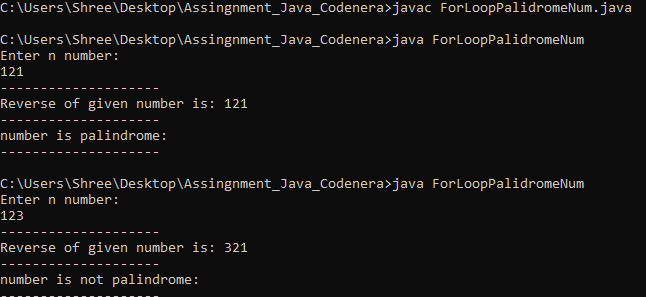
**Output:**



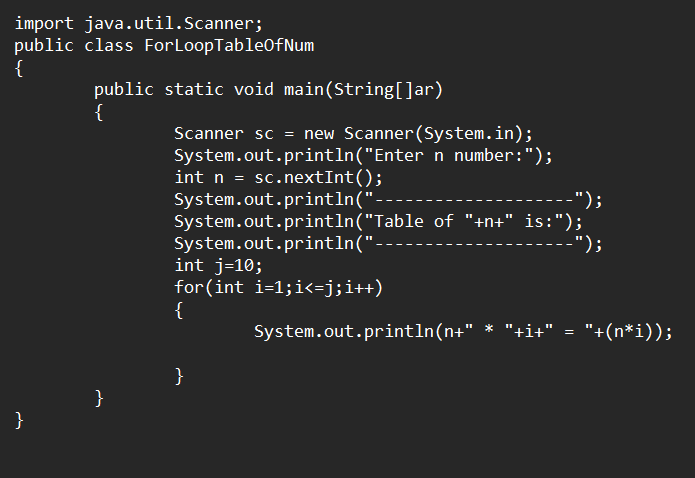
**3.** **Write a java program to Check the number is Palindrome or not.**

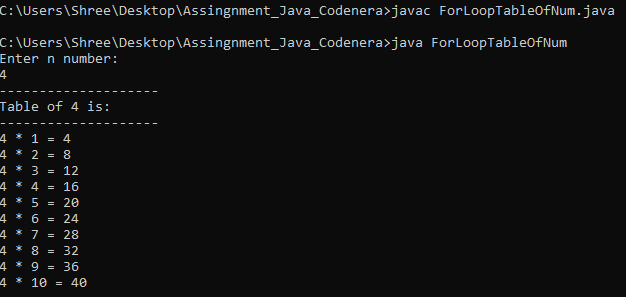
****

**Output:**

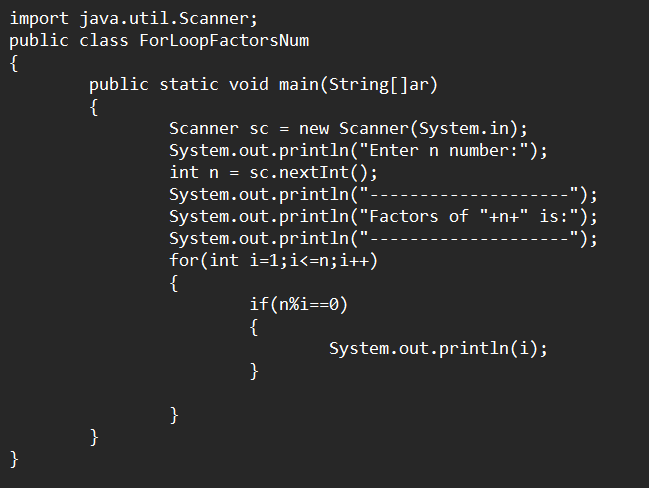


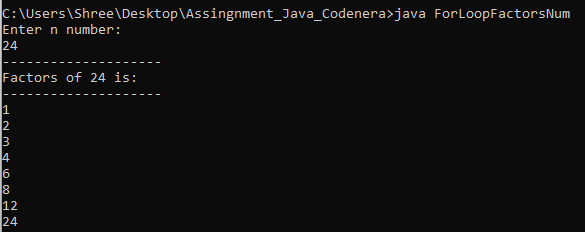
**4. Write a program to enter a number and print he table.**

** Output:**

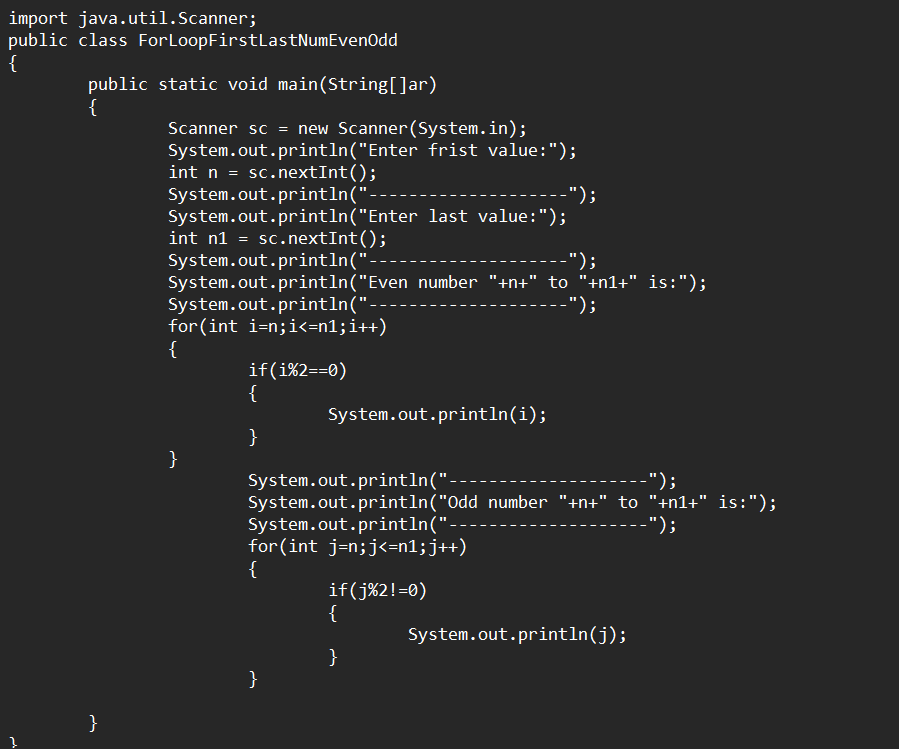


**5. Write a program to print the factors of the given number.**

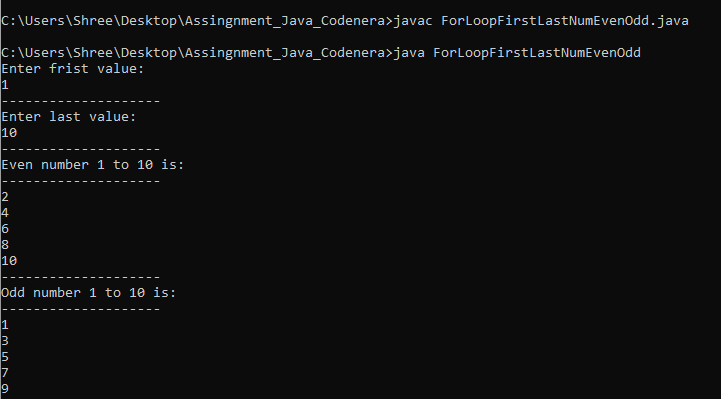
**Output:**



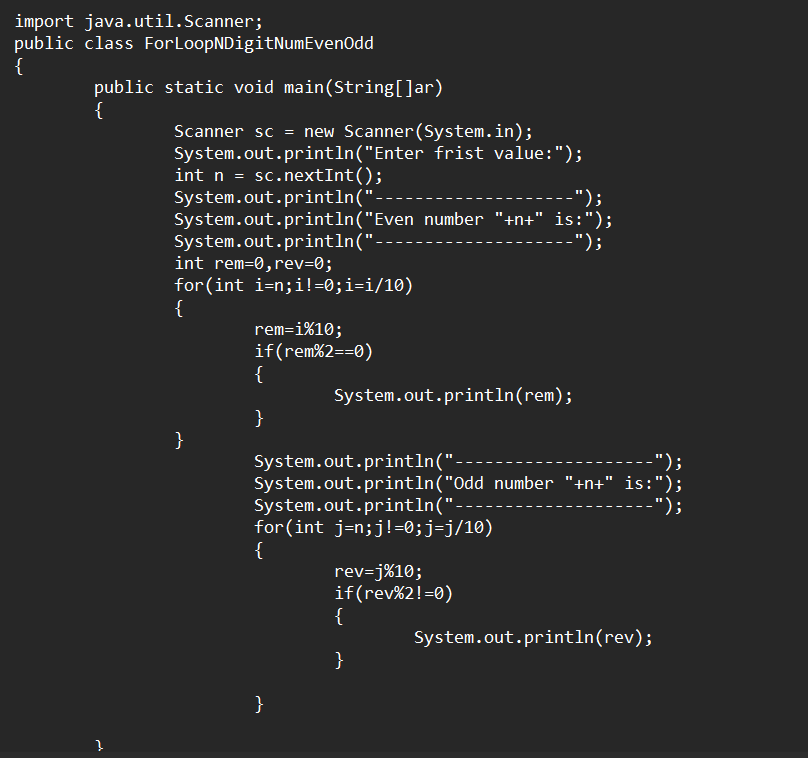
**6. Write a program print even, and odd number from given series from last to start (means you need to take start value and last value, and then print even and odd numbers separately from end to first).**

****

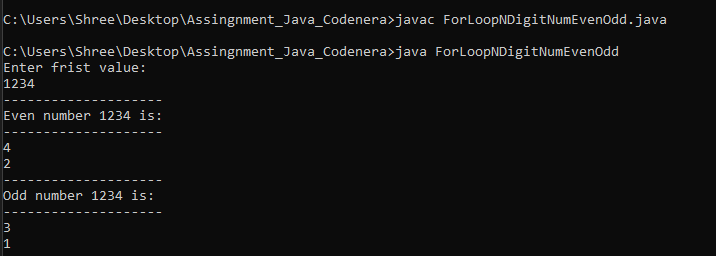
**Output:**

****

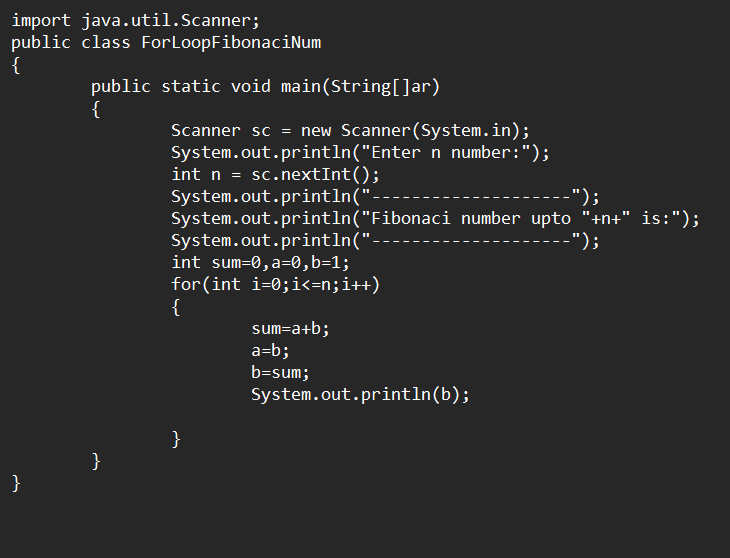
**7. Write a program to take an n digit number, and print even and odd numbers from that number separately by breaking that number.**

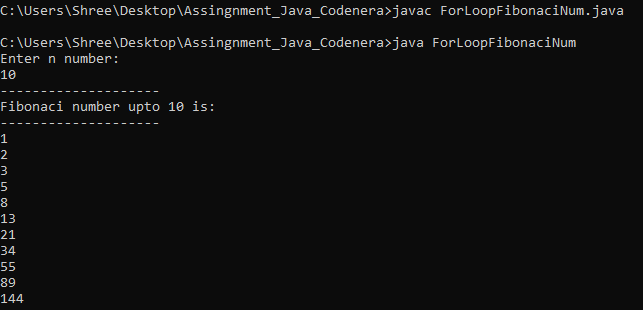
****

**Output:**



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

** Output:**



**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