Name of the Course : Java for beginners: Step–by–step hands-on guide to Java

Level : Moderate

Tool Stack : Java Basic Programming,static and logical operator

Problem Statement : To speed up his composition of generating unpredictable rhythms, A.R.Rahman wants

the list of prime numbers available in a range of numbers.Can you help him out?

Write a java program to print all prime numbers in the interval [a,b] (a and b, both inclusive).

Description : Create a class PrimeNumbers.java with the main method.

Note:

Input 1 should be lesser than Input 2. Both the inputs should be positive. Range must always be greater than zero. If any of the condition mentioned above fails, then display "Provide valid input"

Use a minimum of one for loop and one while loop

**Code:**

**import java.io.\*;**

**import java.util.Scanner;**

**public class PrimeNumbers{**

**public static void main (String[] args) throws IOException {**

**Scanner sc =new Scanner(System.in);**

**System.out.println("Enter two number :");**

**int a=sc.nextInt();**

**int b=sc.nextInt();**

**boolean p=primeNumberList(a,b);**

**if(p==false) {**

**System.out.println("Provide valid input");**

**}**

**}**

**public static boolean primeNumberList(int a,int b) {**

**if(a>b || a==b ||a<0 || b<0 || a==0 || b==0)**

**{ return false;}**

**else{**

**for(int i=a;i<=b;i++)**

**{**

**if(isPrime(i))**

**{**

**System.out.print(i+" ");**

**}**

**}**

**return true;**

**}**

**}**

**public static boolean isPrime(int n)**

**{**

**int c=0;**

**int i=1;**

**while(i<=n)**

**{**

**if(n%i==0)**

**{**

**c++;**

**}**

**i++;**

**}**

**if(c==2)**

**{**

**return true;**

**}**

**else**

**return false;**

**}**

**}**

Junit Testing

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.Test;

class PrimeNumbersTest {

@Test

void testIsPrime() {

assertEquals(true, PrimeNumbers.isPrime(5));

assertEquals(false, PrimeNumbers.isPrime(8));

}

@Test

void testPrimeNumberList() {

assertEquals(true, PrimeNumbers.primeNumberList(2,15));

assertEquals(false, PrimeNumbers.primeNumberList(8,5));

}

}

Test Data1

2

15

2 3 5 7 11 13

Test Data2

8

5

Provide valid input

Learning outcome: Participant could able to learn how to use for loop and while loop and also learn how to use logical operator.