



Continuous Assessment Question-8

Name-Ankush Kumar

Registration No-12204834

Roll No- B55

Section-D2212

Group-2

Question set-ODD

Sandbox Link: -

<https://codehs.com/sandbox/id/reg-no-12204834-RbnSLI>

Q-Creates a thread that continuously generates random numbers between 1 and 100. Another thread should continuously check if the generated number is a prime number and print it to the console if it is. Use synchronization to make sure that the two threads do not interfere with each other.

Code

import java.util.Random;

public class MyProgram {

 private static int randomNumber;

 private static boolean isPrime;

 public static void main(String[] args) {

 Random rand = new Random();

 Thread numberGeneratorThread = new Thread(new
Runnable() {

 @Override

 public void run() {

```
while (true) {  
    synchronized (PrimeNumberGenerator.class) {  
        randomNumber = rand.nextInt(100) + 1;  
        PrimeNumberGenerator.class.notify();  
    }  
    try {  
        Thread.sleep(1000);  
    } catch (InterruptedException e) {  
        e.printStackTrace();  
    }  
}  
}  
});
```

```
Thread primeCheckerThread = new Thread(new  
Runnable() {  
  
    @Override  
  
    public void run() {  
  
        while (true) {  
  
            synchronized (PrimeNumberGenerator.class) {
```

```

_____ try {
_____ PrimeNumberGenerator.class.wait();
_____ } catch (InterruptedException e) {
_____ e.printStackTrace();
_____ }
_____ isPrime = checkPrime(randomNumber);
_____ if (isPrime) {
_____ System.out.println(randomNumber + " is a
prime number.");
_____ }
_____ }
_____ }
_____ }
_____ }
_____ });

_____ numberGeneratorThread.start();
_____ primeCheckerThread.start();
_____ }

```

```

_____ private static boolean checkPrime(int n) {

```

```
____ if (n <= 1) {  
____   return false;  
____ }  
____ for (int i = 2; i <= Math.sqrt(n); i++) {  
____   if (n % i == 0) {  
____     return false;  
____   }  
____ }  
____ return true;  
____ }  
  
}
```

Output

New +

PrimeNumberGenerator.java

MyProgram.java

Reg No-12204834

Share Save

Output Docs More

Run Stop

```
1 import java.util.Random;
2
3 public class MyProgram {
4
5     private static int randomNumber;
6     private static boolean isPrime;
7
8     public static void main(String[] args) {
9         Random rand = new Random();
10         Thread numberGeneratorThread = new Thread(new Runnable() {
11             @Override
12             public void run() {
13                 while (true) {
14                     synchronized (PrimeNumberGenerator.class) {
15                         randomNumber = rand.nextInt(100) + 1;
16                         PrimeNumberGenerator.class.notify();
17                     }
18                     try {
19                         Thread.sleep(1000);
20                     } catch (InterruptedException e) {
21                         e.printStackTrace();
22                     }
23                 }
24             }
25         });
26
27         Thread primeCheckerThread = new Thread(new Runnable() {
28             @Override
29             public void run() {
30                 while (true) {
31                     synchronized (PrimeNumberGenerator.class) {
32                         try {
33                             PrimeNumberGenerator.class.wait();
34                         } catch (InterruptedException e) {
35                             e.printStackTrace();
36                         }
37                         isPrime = checkPrime(randomNumber);
38                     }
39                 }
40             }
41         });
42         primeCheckerThread.start();
43         numberGeneratorThread.start();
44     }
45 }
```

```
37 is a prime number.
29 is a prime number.
73 is a prime number.
59 is a prime number.
17 is a prime number.
83 is a prime number.
73 is a prime number.
89 is a prime number.
73 is a prime number.
5 is a prime number.
83 is a prime number.
13 is a prime number.
83 is a prime number.
47 is a prime number.
67 is a prime number.
[]
```