|  |  |  |
| --- | --- | --- |
| **LAB221 Assignment** | **Type:** | **Short Assignment** |
| **Code:** | **J2.S.P0005** |
| **LOC:** | **35** |
| **Slot(s):** | **1** |

**Title**

Synchronized the list of integer numbers

**Background Context**

A computer system can take and do more than a task at a time; the problem is how to control the shared data, memory, recourses, CPU…among these tasks. Let see a simple sample of Synchronized the list of integer number, how we do the synchronization the tasks.

In this assignment, assume that we have 2 threads (Adding and Deleting). Every a second, Adding try to append a random whole number (between 1 and 100) to the end of the list, until number of items in the list up to 5, Adding will inform the Deleting to delete 3 last items of the list, and will inform back Adding, so on…

**Program Specifications**

Create an application using Thread synchronization to implement the application. The application should consist of the following classes:

1. **Store.java**

This class contains information of a whole number, including that number and a boolean available said that the list is ready to be deleted or not.

**2. Consumer.java**

This is a subclass of Thread class that consumes and prints the deleted numbers that are saving in the Store currently.

**3. Producer.java**

This is a subclass of Thread class that produces and prints the inserted numbers that are saving in the Store currently.

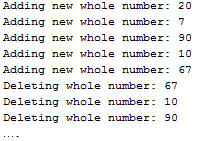
**4. Main.java**

Create 2 threads Consumer and Producer; initiate the execution of the both threads.

***Function details:***

1. Create two threads (Producer and consumer) initiate the execution of both the threads.

***Expectation of User interface:***



**Guidelines**

To be able to get a random number between 1 and 100 you can use code

int min = 1;

int max = 100;

Random r = new Random();

int i1 = r.nextInt(max - min + 1) + min;