

```
/*  
 * To change this license header, choose License Headers in Project Properties.  
 * To change this template file, choose Tools | Templates  
 * and open the template in the editor.  
 */
```

```
package producerconsumer;
```

```
/**  
 *  
 * @author DELL  
 */
```

```
public class Consumer implements Runnable {
```

```
    private MyQueue myQueue;
```

```
    public Consumer(MyQueue myQueue) {
```

```
        this.myQueue = myQueue;
```

```
    }
```

```
    public void run(){
```

```
        while(true){
```

```
            //consumer get items
```

```
            myQueue.get();
```

```
            try{
```

```
                Thread.sleep(2000);
```

```
            }catch(InterruptedException e){
```

```
e.printStackTrace();
```

```
}
```

```
}
```

```
}
```

```
}
```

```
/*
```

```
* To change this license header, choose License Headers in Project Properties.
```

```
* To change this template file, choose Tools | Templates
```

```
* and open the template in the editor.
```

```
*/
```

```
package producerconsumer;
```

```
import java.util.concurrent.Semaphore;
```

```
/**
```

```
*
```

```
* @author DELL
```

```
*/
```

```
public class MyQueue {
```

```
    //an item
```

```
    private int item;
```

```
    //semaphoreConsumerninitialized with 0 permits
```

```

//to ensure put() method executes first

private Semaphore semaphoreConsumer = new Semaphore(0);

private Semaphore semaphoreProducer = new Semaphore(1);


//to get an item from buffer

public void get(){

    try{

        //before a consumer can consume an item,

        //it must acquire permit from semaphoreConsumer

        semaphoreConsumer.acquire();

    }catch(InterruptedException e){

        System.out.println("InterruptedException caught");

    }


//consumer consuming an item

// System.out.println("Consumer consumed item:" +item);

    String[] studentName = new String[30];

    studentName[0] = "Nolwazi Dlamini";

    studentName[1] = "Sipho Shiba";

    studentName[2] = "Muzi Dlamini";

    studentName[3] = "Lwazi Gama";

    studentName[4] = "Sabo Mnisi";

    studentName[5] = "SBura Zwane";

    studentName[6] = "Mandla Matse";

    studentName[7] = "Thoko Zulu";

```

```
studentName[8] = "Kwazi Dlamini";
```

```
studentName[9] = "Sive Dube";
```

```
int[] marks = new int[10];
```

```
marks[0] =60;
```

```
marks[1] =70;
```

```
marks[2] =90;
```

```
marks[3] =80;
```

```
marks[4] =69;
```

```
marks[5] =67;
```

```
marks[6] =78;
```

```
marks[7] =64;
```

```
marks[8] =74;
```

```
marks[9] =60;
```

```
int[] ID = new int[10];
```

```
ID[0] =12345678;
```

```
ID[1] =12435677;
```

```
ID[2] =24325567;
```

```
ID[3] =80768990;
```

```
ID[4] =65675349;
```

```
ID[5] =63546547;
```

```
ID[6] =75567448;
```

```
ID[7] =60098974;
```

```
ID[8] =74345664;
```

```
ID[9] =60445644;
```

```
String[] course = new String[30];
```

```
course[0]= "CSC111";course[1]= "CSC112";course[2]= "CSC115";course[3]= "CSC113";course[4]=  
"CSC121";
```

```
course[5]= "CSC134";course[6]= "CSC116";course[7]= "CSC145";course[8]= "CSC151";course[9]=  
"CSC173";
```

```
String[] Programme = new String[30];
```

```
Programme[0]="IT";
```

```
Programme[1]="Education";
```

```
Programme[2]="Comp-Sci";
```

```
Programme[3]="Hums";
```

```
Programme[4]="Agri";
```

```
Programme[5]="Health";
```

```
Programme[6]="Nursing";
```

```
Programme[7]="Physics";
```

```
Programme[8]="Spors";
```

```
Programme[9]="IT-Education";
```

```
for (int i=0; i<10; i++)
```

```
{
```

```
    System.out.println("Consumer consumed item:"+studentName[i] + ID [i] + course [i] +  
marks[i] + Programme [i]);
```

```
}
```

```

        //after consumer consumes an item,

        //it releases semaphoreProducer to notify producer
        semaphoreProducer.release();
    }

    //to put an item in buffer
    public void put(int item){
        try{
            //before producer an produce an item,

            //it must acquire a permit from semaphoreProducer
            semaphoreProducer.acquire();
        }catch(InterruptedException e){
            System.out.println("InterruptedException caught");

        }

        //producer producing an item

        // this.item=item;

        //System.out.println("Producer produced item:" +item);
    }

```

```

String[] studentName = new String[30];

studentName[0] = "Nolwazi Dlamini";

studentName[1] = "Sipho Shiba";

studentName[2] = "Muzi Dlamini";

studentName[3] = "Lwazi Gama";

studentName[4] = "Sabo Mnisi";

studentName[5] = "SBura Zwane";

```

```
studentName[6] = "Mandla Matse";
```

```
studentName[7] = "Thoko Zulu";
```

```
studentName[8] = "Kwazi Dlamini";
```

```
studentName[9] = "Sive Dube";
```

```
int[] marks = new int[10];
```

```
marks[0] =60;
```

```
marks[1] =70;
```

```
marks[2] =90;
```

```
marks[3] =80;
```

```
marks[4] =69;
```

```
marks[5] =67;
```

```
marks[6] =78;
```

```
marks[7] =64;
```

```
marks[8] =74;
```

```
marks[9] =60;
```

```
int[] ID = new int[10];
```

```
ID[0] =12345678;
```

```
ID[1] =12435677;
```

```
ID[2] =24325567;
```

```
ID[3] =80768990;
```

```
ID[4] =65675349;
```

ID[5] =63546547;

ID[6] =75567448;

ID[7] =60098974;

ID[8] =74345664;

ID[9] =60445644;

String[] course = new String[30];

course[0]= "CSC111";course[1]= "CSC112";course[2]= "CSC115";course[3]= "CSC113";course[4]=
"CSC121";

course[5]= "CSC134";course[6]= "CSC116";course[7]= "CSC145";course[8]= "CSC151";course[9]=
"CSC173";

String[] Programme = new String[30];

Programme[0]="IT";

Programme[1]="Education";

Programme[2]="Comp-Sci";

Programme[3]="Hums";

Programme[4]="Agri";

Programme[5]="Health";

Programme[6]="Nursing";

Programme[7]="Physics";

Programme[8]="Spors";

Programme[9]="IT-Education";

for (int i=0; i<10; i++)


```
        {  
            System.out.println("Producer produced item:"+studentName[i] + ID [i] + course [i] +  
marks[i] + Programme [i]);  
        }
```

```
        //After producer produces the item,  
        //it releases semaphoreConsumer to notify consumer  
        semaphoreConsumer.release();
```

```
    }
```

```
}
```

```
package producerconsumer;
```

```
import java.util.Random;
```

```
/**
```

```
 *
```

```
 * @author DELL
```

```

*/

public class Producer implements Runnable{

    private MyQueue myQueue;


    public Producer(MyQueue myQueue){

        this.myQueue = myQueue;

    }

    public void run(){

        while(true){

            Random random = new Random();

            int data = random.nextInt(10);


            //producer put items

            myQueue.put(data);

            try{

                Thread.sleep(2000);

            }catch(InterruptedException e ){

                e.printStackTrace();

            }

        }

    }

}

/*

```

* To change this license header, choose License Headers in Project Properties.

* To change this template file, choose Tools | Templates

* and open the template in the editor.

*/

```
package producerconsumer;
```

```
/**
```

```
*
```

```
* @author DELL
```

```
*/
```

```
public class ProducerConsumer {
```

```
/**
```

```
* @param args the command line arguments
```

```
*/
```

```
public static void main(String[] args) {
```

```
    // creating buffer queue
```

```
    MyQueue myQueue = new MyQueue();
```

```
    Producer producer = new Producer(myQueue);
```

```
    Consumer consumer = new Consumer(myQueue);
```

```
    Thread producerThread = new Thread(producer);
```

```
    //starting producer thread
```

```
    producerThread.start();
```

```
Thread consumerThread = new Thread(consumer);
```

```
//starting consumer thread
```

```
consumerThread.start();
```

```
}
```

```
}
```