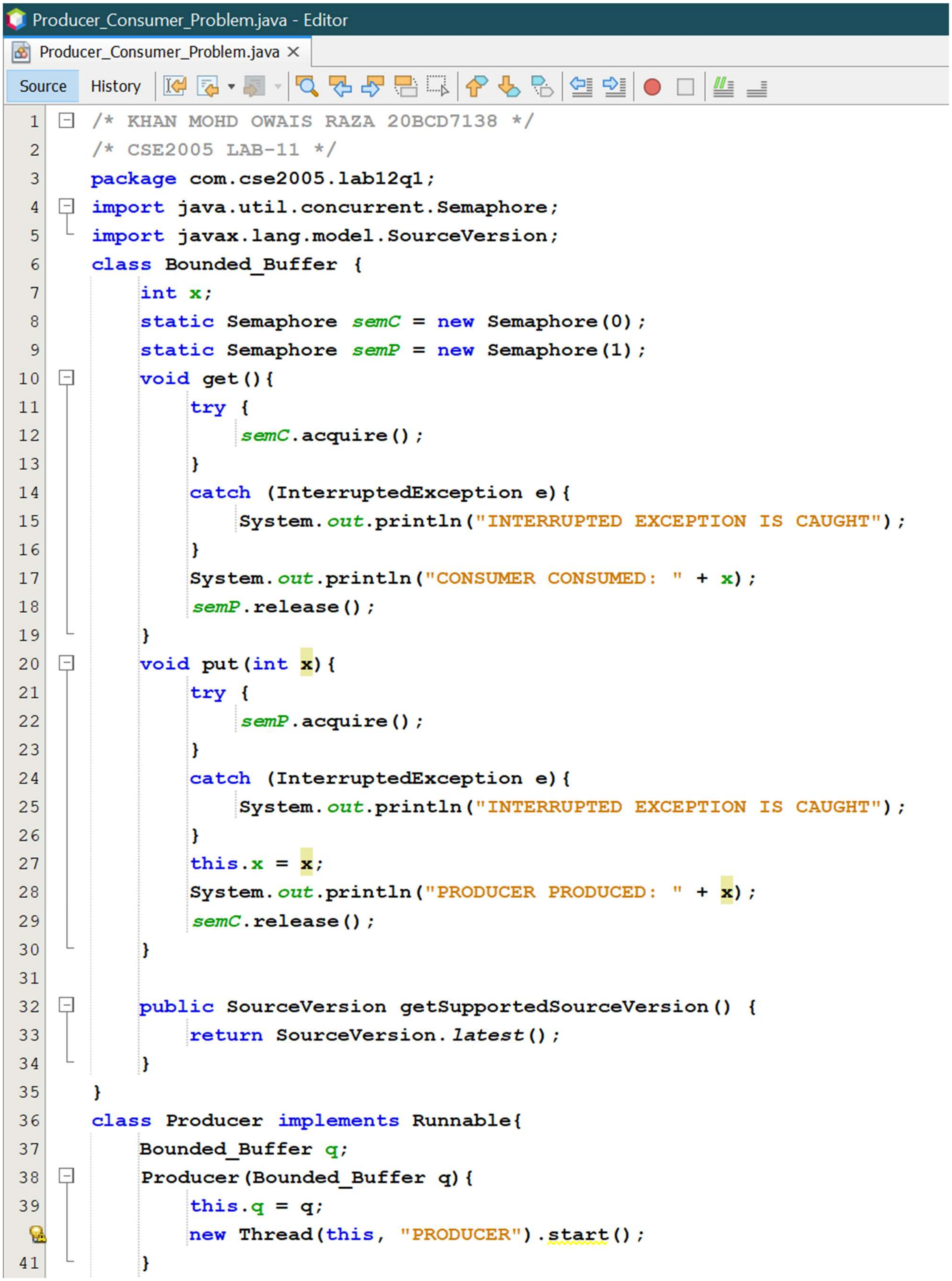
CSE2005 (OOP in Java) Lab – 12 22/12/2021

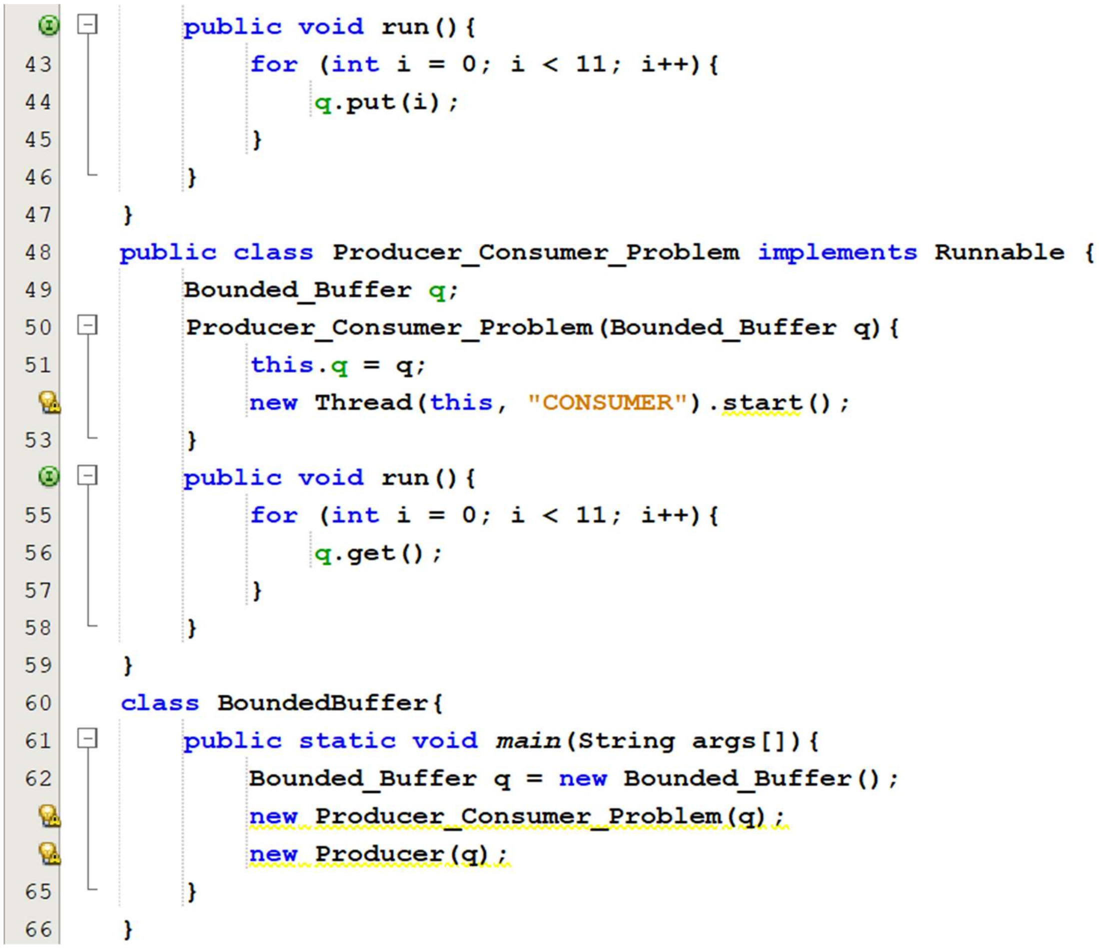
KHAN MOHD. OWAIS RAZA
20BCD7138

Q.1]

Using runnable interface –

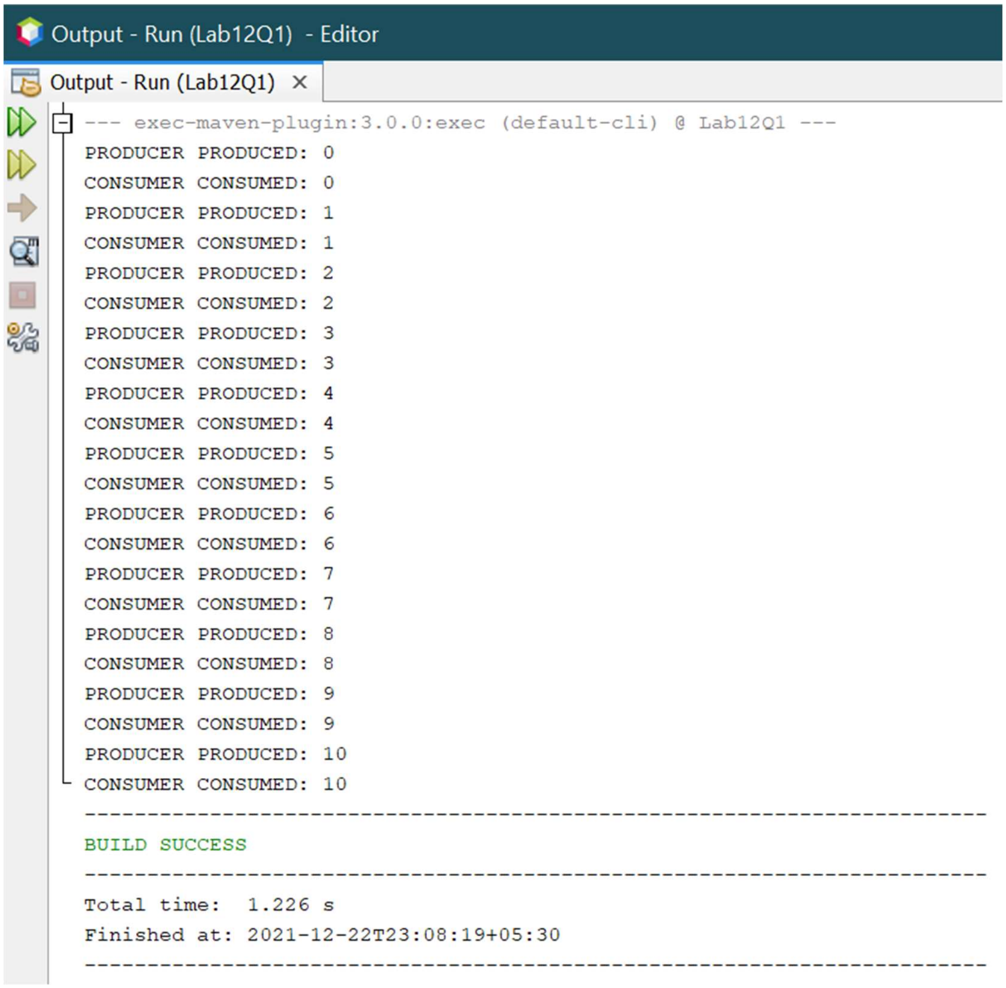

Producer_Consumer_Problem.java - Editor
Producer_Consumer_Problem.java
Source
History

```java
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-11 */
package com.cse2005.lab12q1;
import java.util.concurrent.Semaphore;
import javax.lang.model.SourceVersion;
class Bounded_Buffer {
    int x;
    static Semaphore semC = new Semaphore(0);
    static Semaphore semP = new Semaphore(1);
    void get(){
        try {
            semC.acquire();
        }
        catch (InterruptedException e){
            System.out.println("INTERRUPTED EXCEPTION IS CAUGHT");
        }
        System.out.println("CONSUMER CONSUMED: " + x);
        semP.release();
    }
    void put(int x){
        try {
            semP.acquire();
        }
        catch (InterruptedException e){
            System.out.println("INTERRUPTED EXCEPTION IS CAUGHT");
        }
        this.x = x;
        System.out.println("PRODUCER PRODUCED: " + x);
        semC.release();
    }

    public SourceVersion getSupportedSourceVersion() {
        return SourceVersion.latest();
    }
}
class Producer implements Runnable{
    Bounded_Buffer q;
    Producer(Bounded_Buffer q){
        this.q = q;
        new Thread(this, "PRODUCER").start();
    }
```



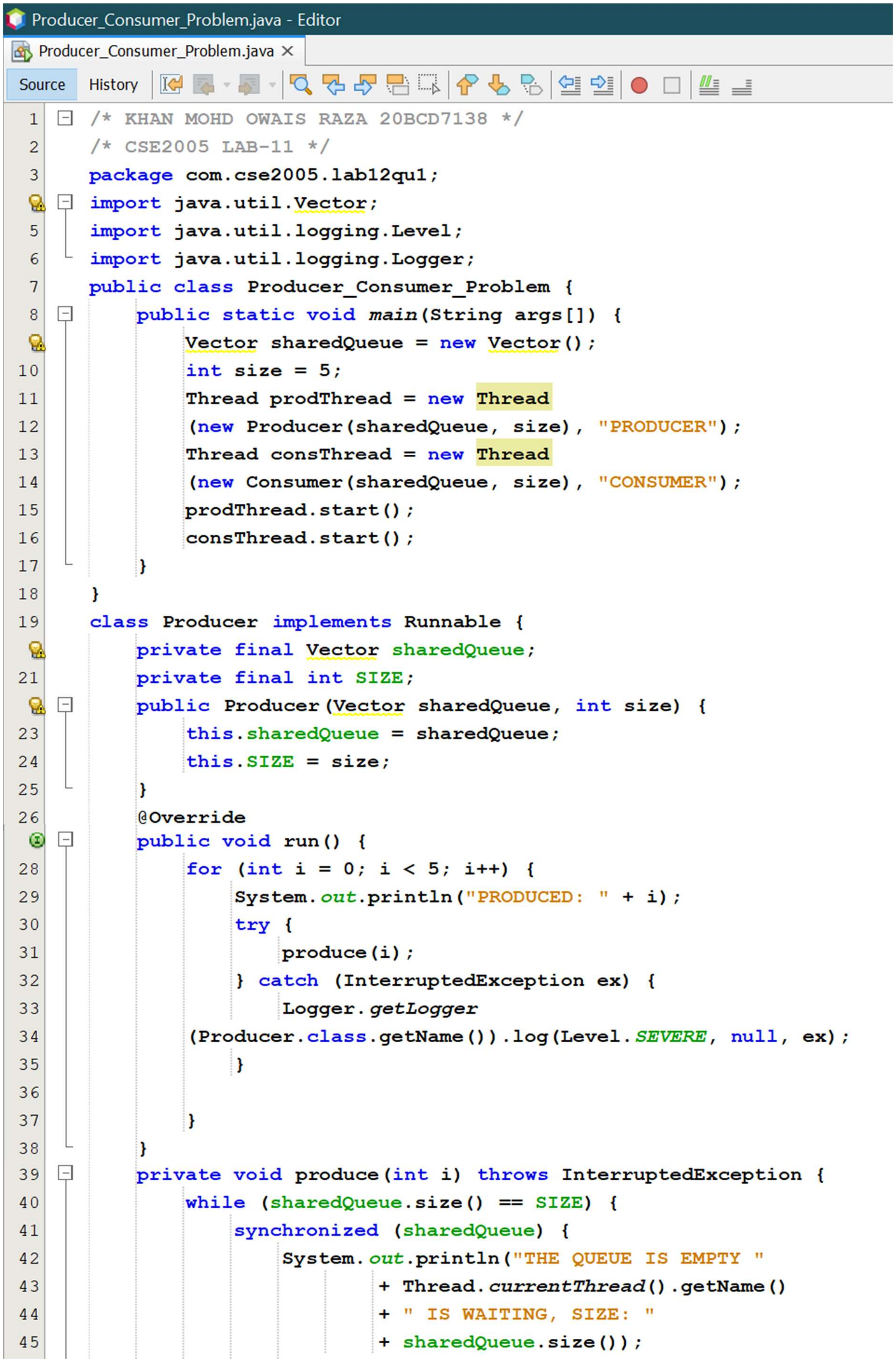
```java
    public void run(){
        for (int i = 0; i < 11; i++){
            q.put(i);
        }
    }
}
public class Producer_Consumer_Problem implements Runnable {
    Bounded_Buffer q;
    Producer_Consumer_Problem(Bounded_Buffer q){
        this.q = q;
        new Thread(this, "CONSUMER").start();
    }
    public void run(){
        for (int i = 0; i < 11; i++){
            q.get();
        }
    }
}
class BoundedBuffer{
    public static void main(String args[]){
        Bounded_Buffer q = new Bounded_Buffer();
        new Producer_Consumer_Problem(q);
        new Producer(q);
    }
}
```

Output console –

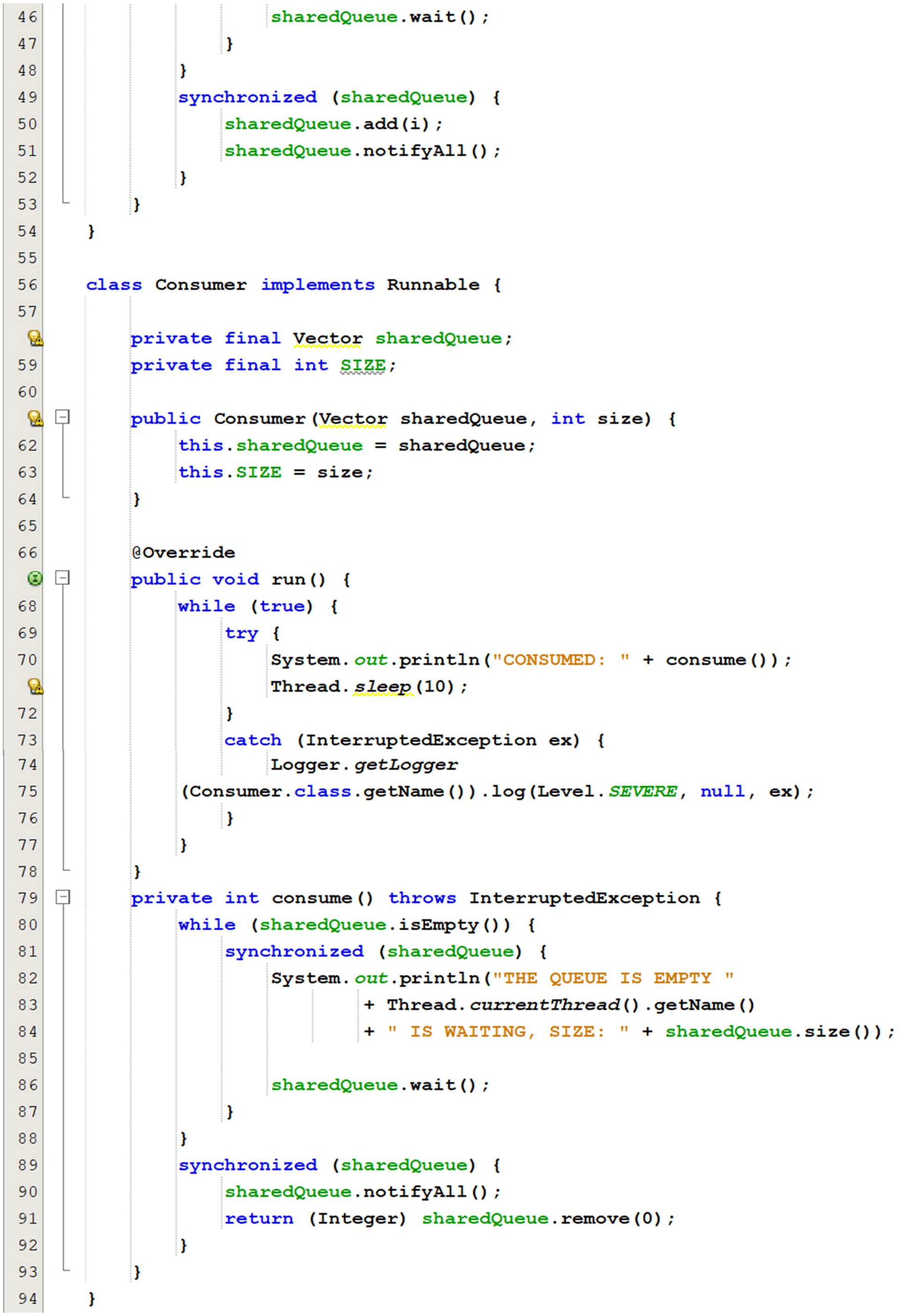


```
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab12Q1 ---
PRODUCER PRODUCED: 0
CONSUMER CONSUMED: 0
PRODUCER PRODUCED: 1
CONSUMER CONSUMED: 1
PRODUCER PRODUCED: 2
CONSUMER CONSUMED: 2
PRODUCER PRODUCED: 3
CONSUMER CONSUMED: 3
PRODUCER PRODUCED: 4
CONSUMER CONSUMED: 4
PRODUCER PRODUCED: 5
CONSUMER CONSUMED: 5
PRODUCER PRODUCED: 6
CONSUMER CONSUMED: 6
PRODUCER PRODUCED: 7
CONSUMER CONSUMED: 7
PRODUCER PRODUCED: 8
CONSUMER CONSUMED: 8
PRODUCER PRODUCED: 9
CONSUMER CONSUMED: 9
PRODUCER PRODUCED: 10
CONSUMER CONSUMED: 10
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.226 s
Finished at: 2021-12-22T23:08:19+05:30
------------------------------------------------------------------------
```

Another technique using runnable interface –

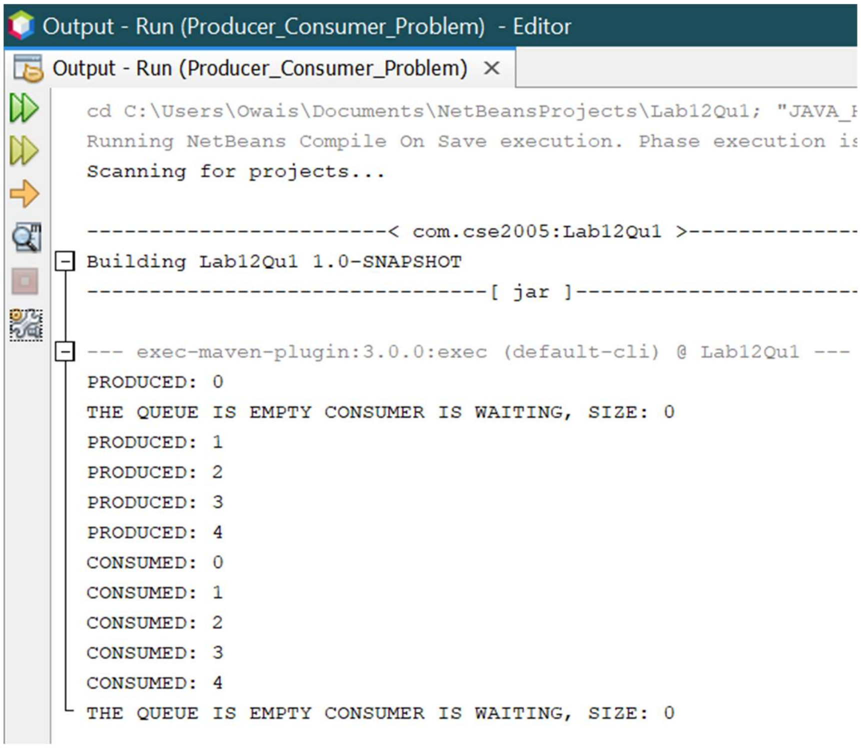

Producer_Consumer_Problem.java - Editor
Producer_Consumer_Problem.java ✕
Source
History

```
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-11 */
package com.cse2005.lab12qu1;
import java.util.Vector;
import java.util.logging.Level;
import java.util.logging.Logger;
public class Producer_Consumer_Problem {
    public static void main(String args[]) {
        Vector sharedQueue = new Vector();
        int size = 5;
        Thread prodThread = new Thread
        (new Producer(sharedQueue, size), "PRODUCER");
        Thread consThread = new Thread
        (new Consumer(sharedQueue, size), "CONSUMER");
        prodThread.start();
        consThread.start();
    }
}
class Producer implements Runnable {
    private final Vector sharedQueue;
    private final int SIZE;
    public Producer(Vector sharedQueue, int size) {
        this.sharedQueue = sharedQueue;
        this.SIZE = size;
    }
    @Override
    public void run() {
        for (int i = 0; i < 5; i++) {
            System.out.println("PRODUCED: " + i);
            try {
                produce(i);
            } catch (InterruptedException ex) {
                Logger.getLogger
        (Producer.class.getName()).log(Level.SEVERE, null, ex);
            }

        }
    }
    private void produce(int i) throws InterruptedException {
        while (sharedQueue.size() == SIZE) {
            synchronized (sharedQueue) {
                System.out.println("THE QUEUE IS EMPTY "
                        + Thread.currentThread().getName()
                        + " IS WAITING, SIZE: "
                        + sharedQueue.size());
```



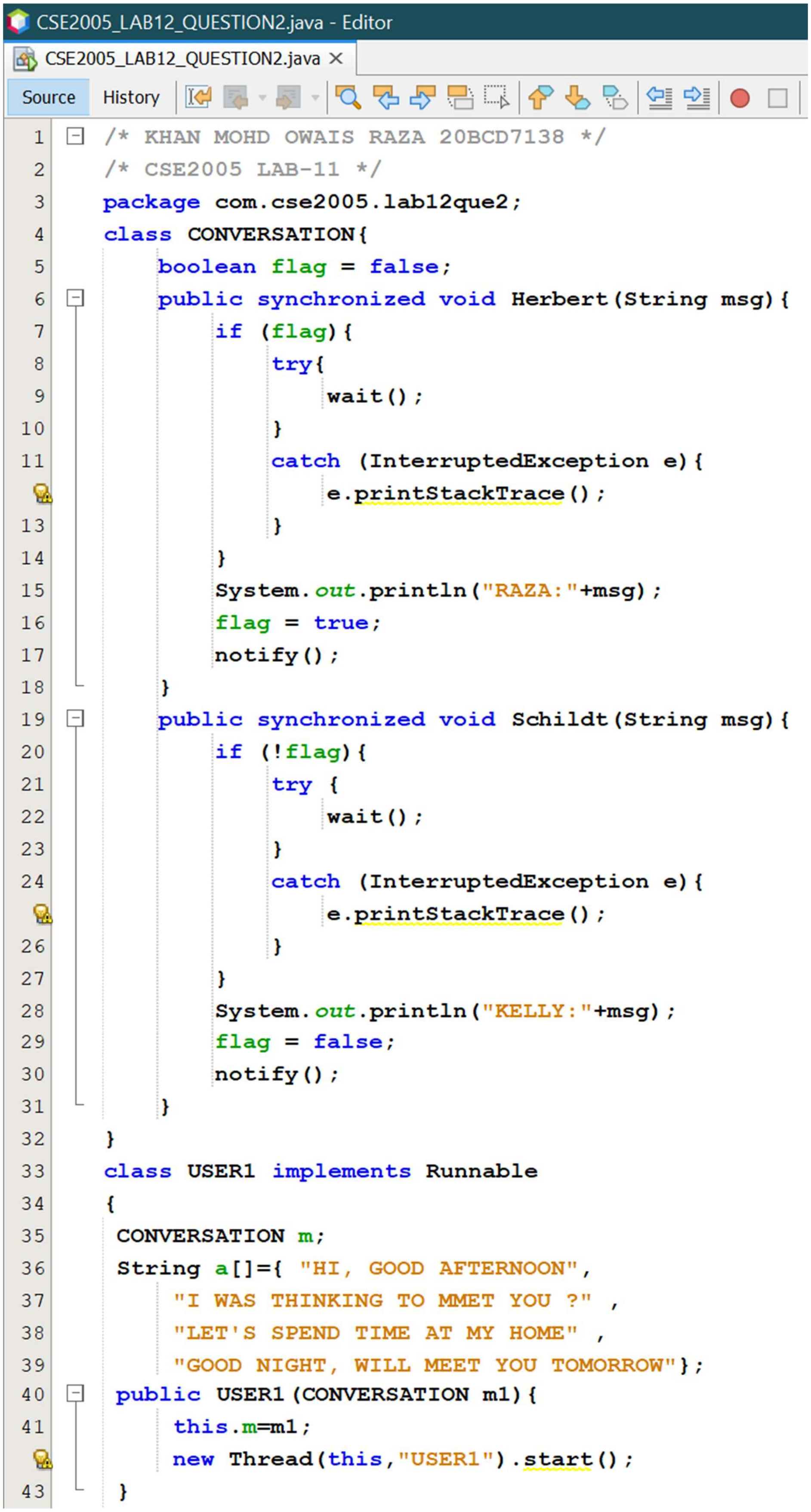
```java
                sharedQueue.wait();
            }
        }
        synchronized (sharedQueue) {
            sharedQueue.add(i);
            sharedQueue.notifyAll();
        }
    }
}

class Consumer implements Runnable {

    private final Vector sharedQueue;
    private final int SIZE;

    public Consumer(Vector sharedQueue, int size) {
        this.sharedQueue = sharedQueue;
        this.SIZE = size;
    }

    @Override
    public void run() {
        while (true) {
            try {
                System.out.println("CONSUMED: " + consume());
                Thread.sleep(10);
            }
            catch (InterruptedException ex) {
                Logger.getLogger
        (Consumer.class.getName()).log(Level.SEVERE, null, ex);
            }
        }
    }
    private int consume() throws InterruptedException {
        while (sharedQueue.isEmpty()) {
            synchronized (sharedQueue) {
                System.out.println("THE QUEUE IS EMPTY "
                        + Thread.currentThread().getName()
                        + " IS WAITING, SIZE: " + sharedQueue.size());

                sharedQueue.wait();
            }
        }
        synchronized (sharedQueue) {
            sharedQueue.notifyAll();
            return (Integer) sharedQueue.remove(0);
        }
    }
}
```

Output console –

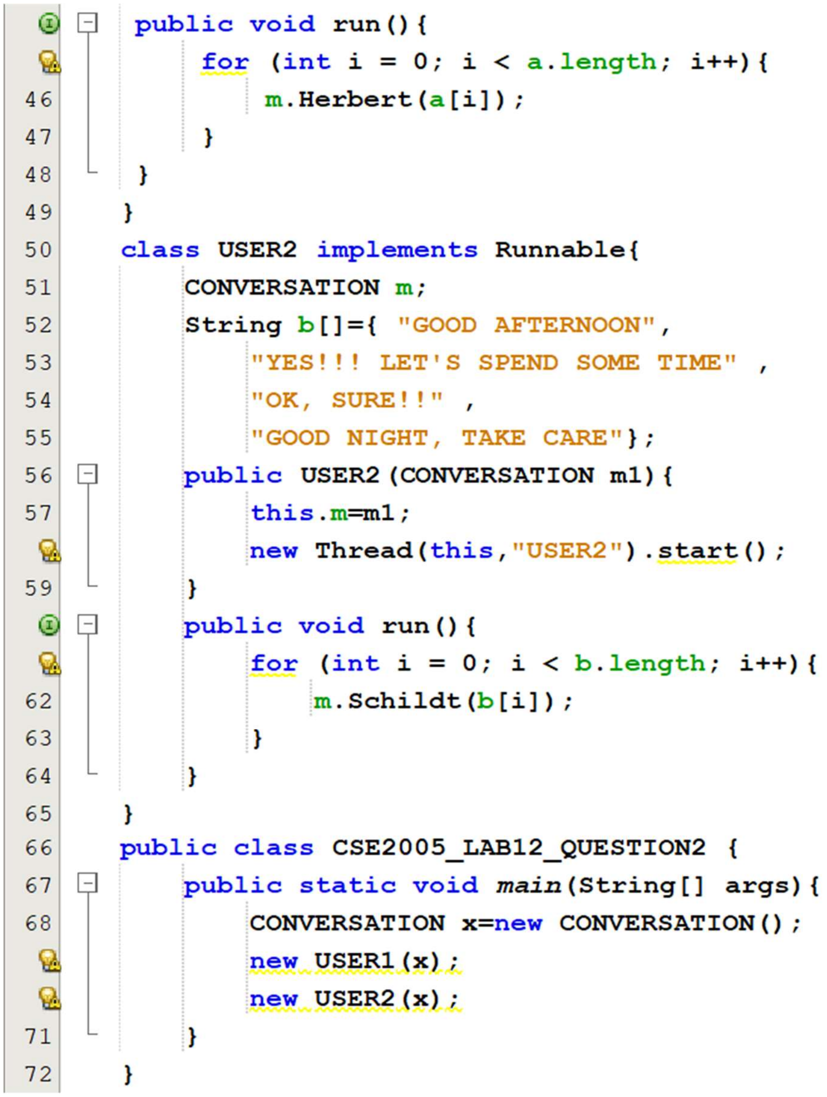

Output - Run (Producer_Consumer_Problem) - Editor
Output - Run (Producer_Consumer_Problem) ✕

```
cd C:\Users\Owais\Documents\NetBeansProjects\Lab12Qu1; "JAVA_
Running NetBeans Compile On Save execution. Phase execution is
Scanning for projects...

------------------------< com.cse2005:Lab12Qu1 >--------------
Building Lab12Qu1 1.0-SNAPSHOT
--------------------------------[ jar ]-----------------------

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab12Qu1 ---
PRODUCED: 0
THE QUEUE IS EMPTY CONSUMER IS WAITING, SIZE: 0
PRODUCED: 1
PRODUCED: 2
PRODUCED: 3
PRODUCED: 4
CONSUMED: 0
CONSUMED: 1
CONSUMED: 2
CONSUMED: 3
CONSUMED: 4
THE QUEUE IS EMPTY CONSUMER IS WAITING, SIZE: 0
```

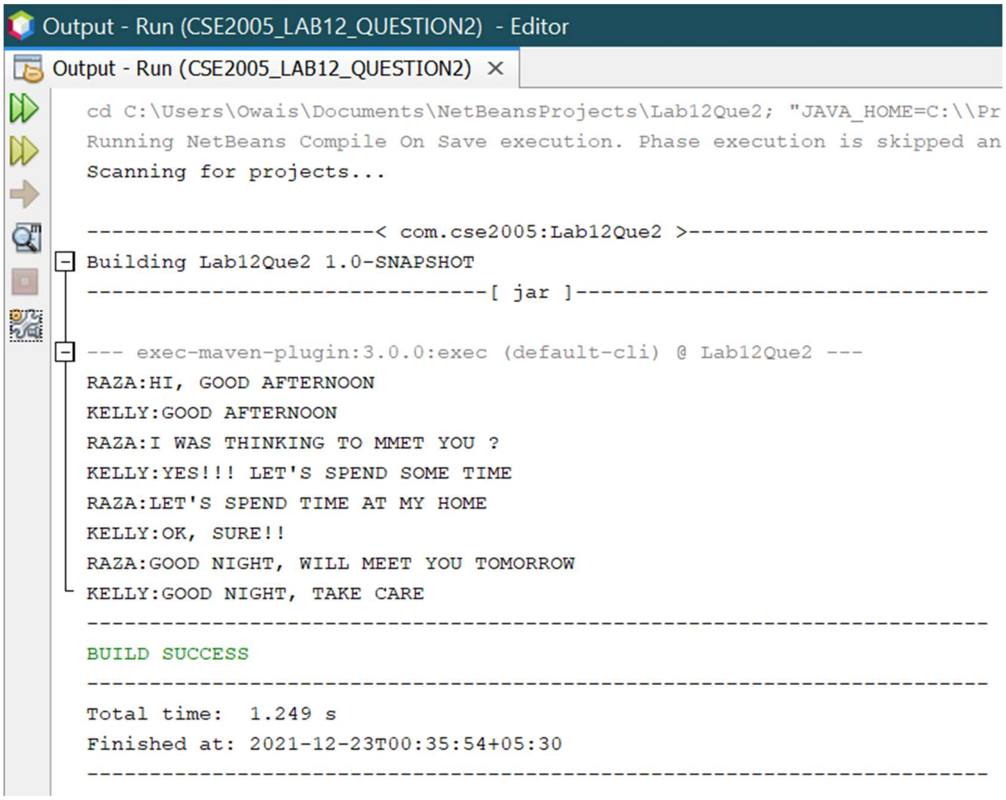
Q.2]


CSE2005_LAB12_QUESTION2.java - Editor
CSE2005_LAB12_QUESTION2.java ×
Source
History

```java
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-11 */
package com.cse2005.lab12que2;
class CONVERSATION{
    boolean flag = false;
    public synchronized void Herbert(String msg){
        if (flag){
            try{
                wait();
            }
            catch (InterruptedException e){
                e.printStackTrace();
            }
        }
        System.out.println("RAZA:"+msg);
        flag = true;
        notify();
    }
    public synchronized void Schildt(String msg){
        if (!flag){
            try {
                wait();
            }
            catch (InterruptedException e){
                e.printStackTrace();
            }
        }
        System.out.println("KELLY:"+msg);
        flag = false;
        notify();
    }
}
class USER1 implements Runnable
{
 CONVERSATION m;
 String a[]={ "HI, GOOD AFTERNOON",
     "I WAS THINKING TO MMET YOU ?" ,
     "LET'S SPEND TIME AT MY HOME" ,
     "GOOD NIGHT, WILL MEET YOU TOMORROW"};
 public USER1(CONVERSATION m1){
     this.m=m1;
     new Thread(this,"USER1").start();
 }
```

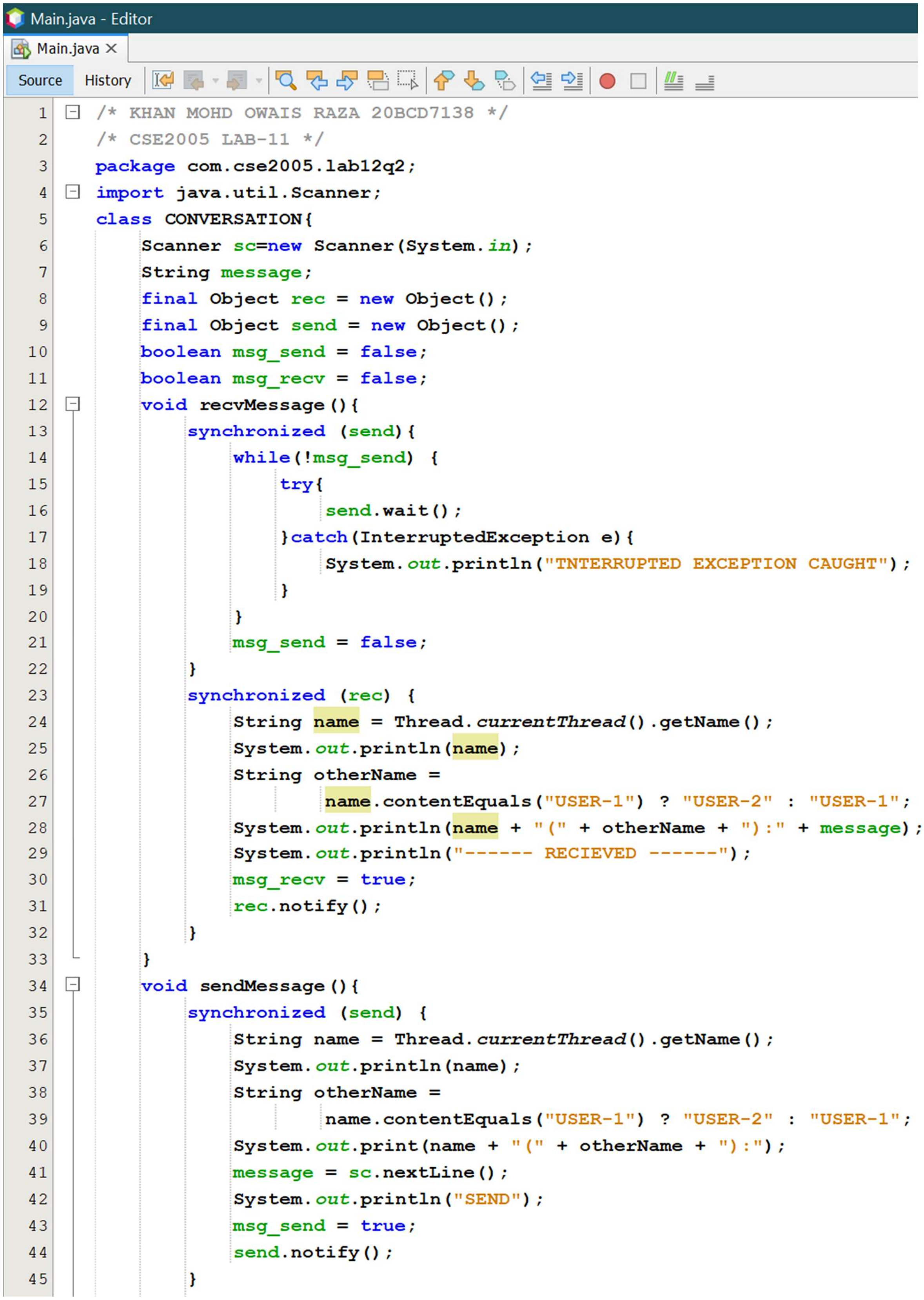


```java
    public void run(){
        for (int i = 0; i < a.length; i++){
            m.Herbert(a[i]);
        }
    }
}
class USER2 implements Runnable{
    CONVERSATION m;
    String b[]={ "GOOD AFTERNOON",
        "YES!!! LET'S SPEND SOME TIME" ,
        "OK, SURE!!" ,
        "GOOD NIGHT, TAKE CARE"};
    public USER2(CONVERSATION m1){
        this.m=m1;
        new Thread(this,"USER2").start();
    }
    public void run(){
        for (int i = 0; i < b.length; i++){
            m.Schildt(b[i]);
        }
    }
}
public class CSE2005_LAB12_QUESTION2 {
    public static void main(String[] args){
        CONVERSATION x=new CONVERSATION();
        new USER1(x);
        new USER2(x);
    }
}
```

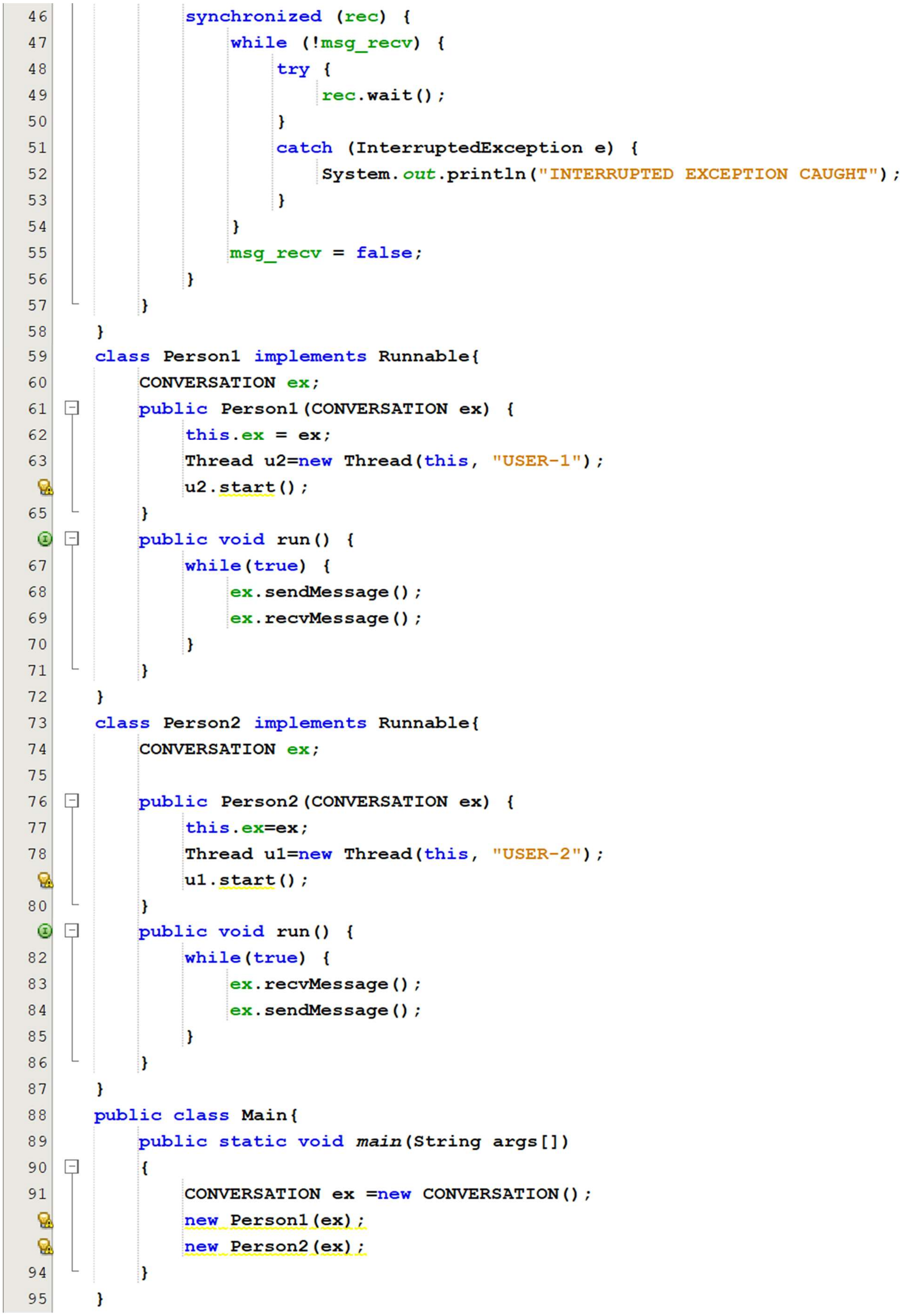

Output - Run (CSE2005_LAB12_QUESTION2) - Editor
Output - Run (CSE2005_LAB12_QUESTION2)

```
cd C:\Users\Owais\Documents\NetBeansProjects\Lab12Que2; "JAVA_HOME=C:\\Pr
Running NetBeans Compile On Save execution. Phase execution is skipped an
Scanning for projects...

-----------------------< com.cse2005:Lab12Que2 >------------------------
Building Lab12Que2 1.0-SNAPSHOT
--------------------------------[ jar ]---------------------------------

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab12Que2 ---
RAZA:HI, GOOD AFTERNOON
KELLY:GOOD AFTERNOON
RAZA:I WAS THINKING TO MMET YOU ?
KELLY:YES!!! LET'S SPEND SOME TIME
RAZA:LET'S SPEND TIME AT MY HOME
KELLY:OK, SURE!!
RAZA:GOOD NIGHT, WILL MEET YOU TOMORROW
KELLY:GOOD NIGHT, TAKE CARE
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.249 s
Finished at: 2021-12-23T00:35:54+05:30
------------------------------------------------------------------------
```

Another technique (using scanner function) –

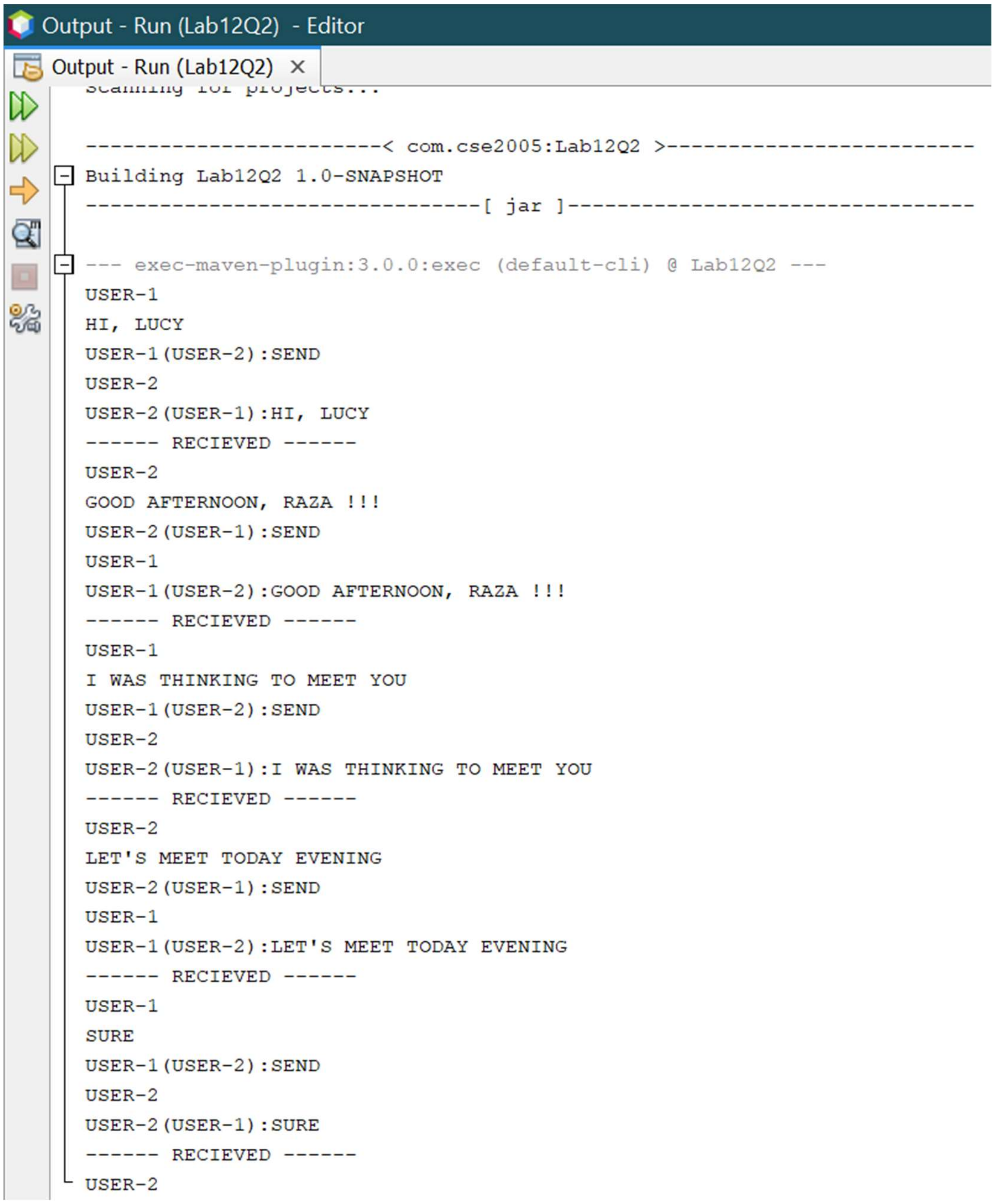


```java
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-11 */
package com.cse2005.lab12q2;
import java.util.Scanner;
class CONVERSATION{
    Scanner sc=new Scanner(System.in);
    String message;
    final Object rec = new Object();
    final Object send = new Object();
    boolean msg_send = false;
    boolean msg_recv = false;
    void recvMessage(){
        synchronized (send){
            while(!msg_send)  {
                try{
                    send.wait();
                }catch(InterruptedException e){
                    System.out.println("TNTERRUPTED EXCEPTION CAUGHT");
                }
            }
            msg_send = false;
        }
        synchronized (rec)  {
            String name = Thread.currentThread().getName();
            System.out.println(name);
            String otherName =
                    name.contentEquals("USER-1")  ?  "USER-2"  :  "USER-1";
            System.out.println(name + "(" + otherName + "):" + message);
            System.out.println("------ RECIEVED ------");
            msg_recv = true;
            rec.notify();
        }
    }
    void sendMessage(){
        synchronized (send)  {
            String name = Thread.currentThread().getName();
            System.out.println(name);
            String otherName =
                    name.contentEquals("USER-1")  ?  "USER-2"  :  "USER-1";
            System.out.print(name + "(" + otherName + "):");
            message = sc.nextLine();
            System.out.println("SEND");
            msg_send = true;
            send.notify();
        }
```



```
        synchronized (rec) {
            while (!msg_recv) {
                try {
                    rec.wait();
                }
                catch (InterruptedException e) {
                    System.out.println("INTERRUPTED EXCEPTION CAUGHT");
                }
            }
            msg_recv = false;
        }
    }
}
class Person1 implements Runnable{
    CONVERSATION ex;
    public Person1(CONVERSATION ex) {
        this.ex = ex;
        Thread u2=new Thread(this, "USER-1");
        u2.start();
    }
    public void run() {
        while(true) {
            ex.sendMessage();
            ex.recvMessage();
        }
    }
}
class Person2 implements Runnable{
    CONVERSATION ex;

    public Person2(CONVERSATION ex) {
        this.ex=ex;
        Thread u1=new Thread(this, "USER-2");
        u1.start();
    }
    public void run() {
        while(true) {
            ex.recvMessage();
            ex.sendMessage();
        }
    }
}
public class Main{
    public static void main(String args[])
    {
        CONVERSATION ex =new CONVERSATION();
        new Person1(ex);
        new Person2(ex);
    }
}
```

Output console –



```
Scanning for projects...

------------------------< com.cse2005:Lab12Q2 >------------------------
Building Lab12Q2 1.0-SNAPSHOT
--------------------------------[ jar ]---------------------------------

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab12Q2 ---
USER-1
HI, LUCY
USER-1(USER-2):SEND
USER-2
USER-2(USER-1):HI, LUCY
------ RECIEVED ------
USER-2
GOOD AFTERNOON, RAZA !!!
USER-2(USER-1):SEND
USER-1
USER-1(USER-2):GOOD AFTERNOON, RAZA !!!
------ RECIEVED ------
USER-1
I WAS THINKING TO MEET YOU
USER-1(USER-2):SEND
USER-2
USER-2(USER-1):I WAS THINKING TO MEET YOU
------ RECIEVED ------
USER-2
LET'S MEET TODAY EVENING
USER-2(USER-1):SEND
USER-1
USER-1(USER-2):LET'S MEET TODAY EVENING
------ RECIEVED ------
USER-1
SURE
USER-1(USER-2):SEND
USER-2
USER-2(USER-1):SURE
------ RECIEVED ------
USER-2
```