Name: Khushi Ramaiya

Class: SEIT2

Roll no.:17

Batch: I4

Experiment no.14

AIM: Thread Management and Synchronization in Java

Program for thread in java:

Input:

import java.io.\*;

// Creating thread by creating the

// objects of that class

class ThreadJoining extends Thread

{

@Override

public void run()

{

for (int i = 0; i < 2; i++)

{

try

{

Thread.sleep(500);

System.out.println("Current Thread: "

+ Thread.currentThread().getName());

}

catch(Exception ex)

{

System.out.println("Exception has" +

" been caught" + ex);

}

System.out.println(i);

}

}

}

class Main

{

public static void main (String[] args)

{

// creating two threads

ThreadJoining t1 = new ThreadJoining();

ThreadJoining t2 = new ThreadJoining();

ThreadJoining t3 = new ThreadJoining();

// thread t1 starts

t1.start();

// starts second thread after when

// first thread t1 has died.

try

{

System.out.println("Current Thread: "

+ Thread.currentThread().getName());

t1.join();

}

catch(Exception ex)

{

System.out.println("Exception has " +

"been caught" + ex);

}

// t2 starts

t2.start();

// starts t3 after when thread t2 has died.

try

{

System.out.println("Current Thread: "

+ Thread.currentThread().getName());

t2.join();

}

catch(Exception ex)

{

System.out.println("Exception has been" +

" caught" + ex);

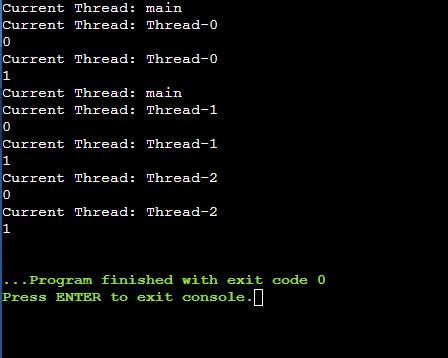
}

t3.start();

}

}

OUTPUT:



Program for synchronization in java:

Input:

import java.io.\*;

class Test {

synchronized void test\_function(int n)

{

// synchronized method

for (int i = 1; i <= 3; i++) {

System.out.println(n + i);

try {

Thread.sleep(500);

}

catch (Exception e) {

System.out.println(e);

}

}

}

}

// Driver Class

public class Main {

// Main function

public static void main(String args[])

{

// only one object

final Test obj = new Test();

Thread a = new Thread() {

public void run() { obj.test\_function(15); }

};

Thread b = new Thread() {

public void run() { obj.test\_function(30); }

};

a.start();

b.start();

}

}

Output:

