## PROGRAM

## //Program to create a class to implement super keyword in a method in java.

class saving

{

int interest=5;

void balence()

{

System.out.println("Here saving account balence is available.");

}

}

class current extends saving

{

void display()

{

super.balence();//Calls the super class method.

System.out.println("This is the current account.");

}

}

class super\_account

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

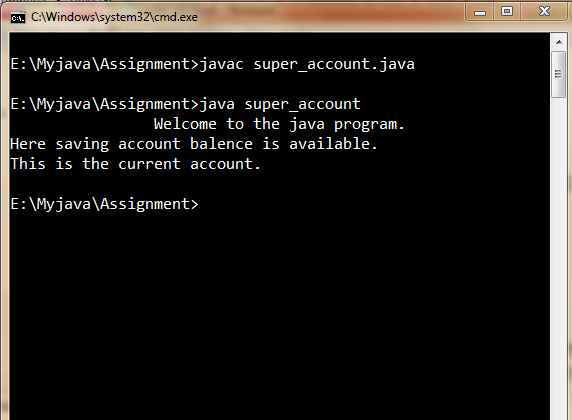
current ob=new current();

ob.display();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a class to implement super class constructor in sub-class bysuper keyword in java.

class current

{

int price;

current(int p)

{

price=p;

System.out.println("The price of current ticket is:- "+price);

}

}

class general extends current

{

general()

{

super(500);//Calls the super class constructor.

System.out.println("The price of general ticket is:- "+(price-100));

}

}

class railway\_super

{

public static void main(String args[])

{

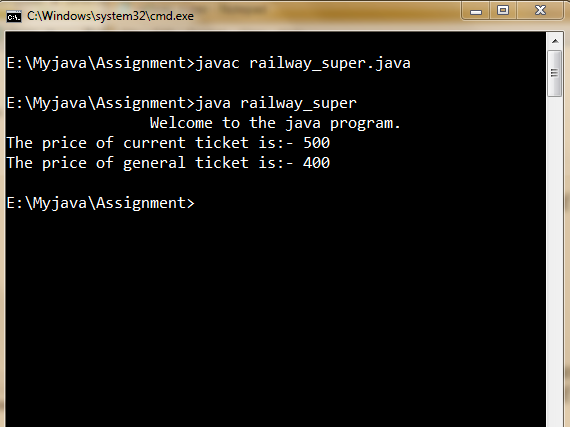
System.out.println("\t\tWelcome to the java program.");

new general();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a package in java.

package pkg;

class mypack

{

public static void main(String args[])

{

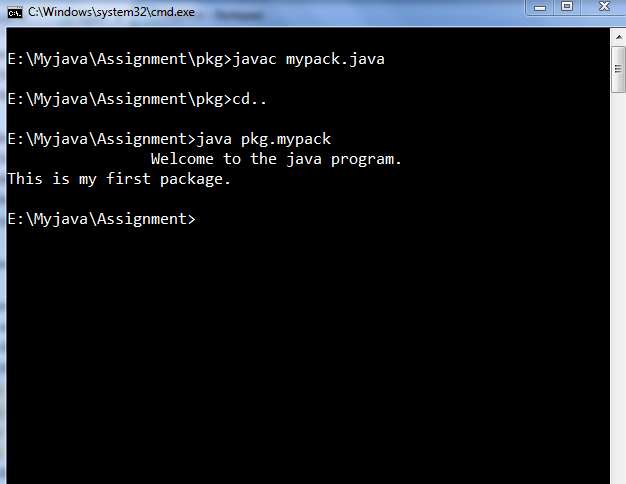
System.out.println("\t\tWelcome to the java program.");

System.out.println("This is my first package.");

}

}

## OUTPUT:-



## PROGRAM

## //Program to create an interface in java.

interface room

{

void rolls();

void panky();

}

class kitchen

{

void cook()

{

System.out.println("Here cook makes food.");

}

}

class rp extends kitchen implements room

{

public void rolls()

{

System.out.println("This is rolls's room.");

}

public void panky()

{

System.out.println("This is panky's room.");

}

}

class myroom\_interface3

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

rp ob=new rp();

ob.rolls();

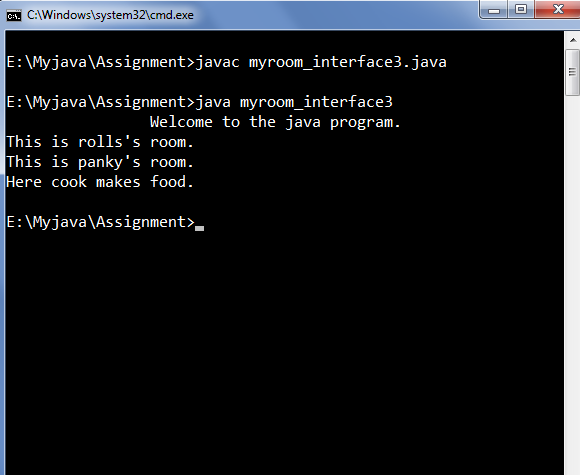
ob.panky();

ob.cook();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create an interface which implements another interface in java.

interface college1

{

static final int num\_st\_iilm=5000;

void iilm();

}

interface college2 extends college1

{

int num\_st\_niet=2000;//By default varible is final.

void niet();

}

class detail implements college2

{

public void iilm()

{

System.out.println("This is my college.");

System.out.println("The totle students are:- "+num\_st\_iilm);

}

public void niet()

{

System.out.println("This is my friend college.");

System.out.println("The totle students are:- "+num\_st\_niet);

}

}

class uptu

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

detail ob=new detail();

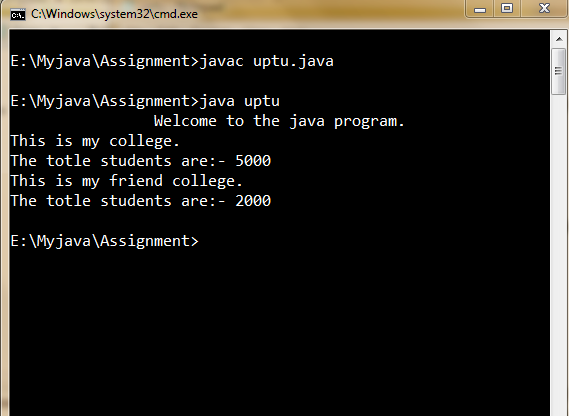
ob.iilm();

ob.niet();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a class which implements interface in javainterface library.

interface library

{

void cs();

void me();

static int st\_fine=2;

static int tech\_fine=1;

}

class teacher implements library

{

teacher()//constructor.

{

System.out.println("\tTeacher interface:-");

System.out.println("Each day fine on each book is:"+tech\_fine+"Rs.");

}

public void cs()

{

System.out.println("Computer science books for teacher.");

}

public void me()

{

System.out.println("Mechanical engineering books for teacher.");

}

}

class student implements library

{

student()//constructor.

{

System.out.println("\tStudnet interface:-");

System.out.println("Each day fine on each book is:"+st\_fine+"Rs.");

}

public void cs()

{

System.out.println("Computer science books for student.");

}

public void me()

{

System.out.println("Mechanical engineering books for student.");

}

}

class student\_teach\_myinterface5

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

teacher ob=new teacher();

ob.cs();

ob.me();

student ob1=new student();

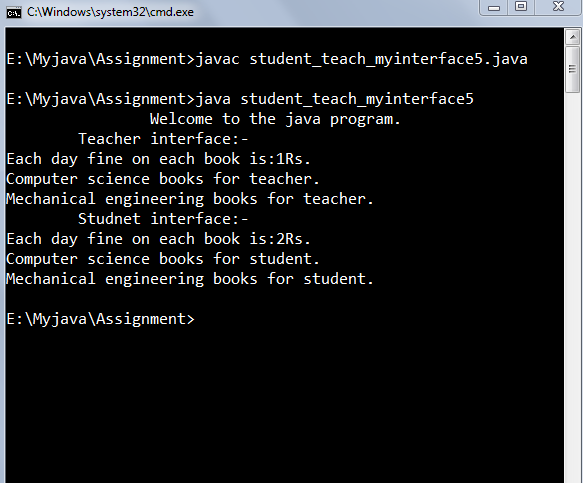
ob1.cs();

ob1.me();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a class which implements more than one interface in java.

interface method1

{

void line();

}

interface method extends method1

{

final int radius=10;

void circle();

void rect();

void square();

}

interface demo

{

void hello();

}

class crs implements method,demo

{

crs()

{

System.out.println("All the interface's methods content are:-");

}

public void circle()//method must declare as public.

{

System.out.println("radius "+radius);

System.out.println("this is circle.");

}

public void rect()

{

System.out.println("this is rectangle.");

}

public void square()

{

System.out.println("this is square.");

}

public void line()

{

System.out.println("this is line.");

}

public void hello()

{

System.out.println("Hello.");

}

}

class multiinterface\_coordinate

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

crs ob=new crs();

ob.circle();

ob.rect();

ob.square();

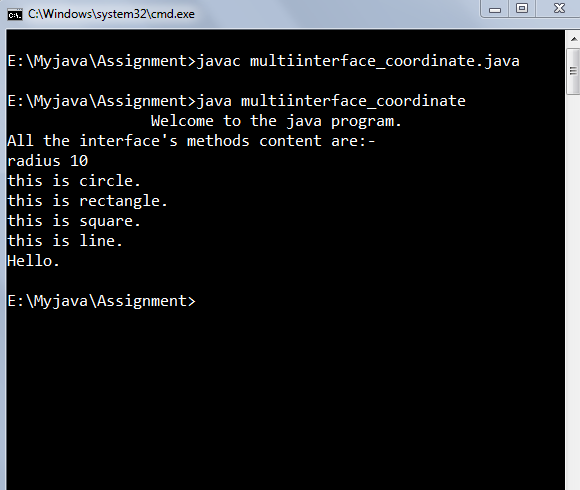
ob.line();

ob.hello();

}

}

## OUTPUT:-



## PROGRAM

## Program to create an abstract class in java.

abstract class nokia

{

abstract void mobile();//Abstract method can not have body.

}

class samsung extends nokia

{

void mobile()

{

System.out.println("This is the nokia model.");

}

}

class my\_abstract

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

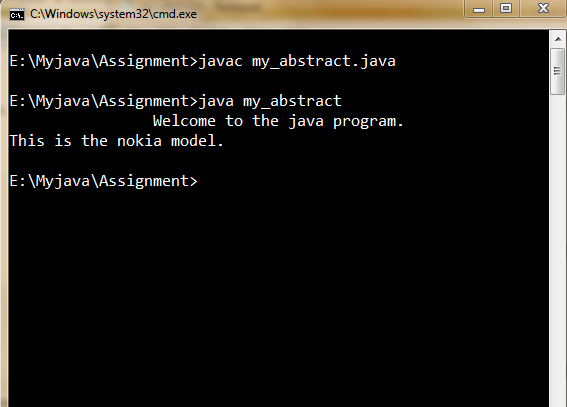
samsung ob=new samsung();

ob.mobile();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a class which implements abstract class in java.

abstract class table

{

abstract void books();//abstract mehtod can not have body.

}

abstract class material extends table

{

abstract void assignment();

void books()//without abstract method having body.

{

System.out.println("Book is on the table.");

}

}

class money extends material//we extend material class because it is abstract.

{

void assignment()

{

System.out.println("This is page of assignment.");

}

void display()

{

System.out.println("by this class'method we show all item.");

}

}

class study

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

money ob=new money();

ob.books();

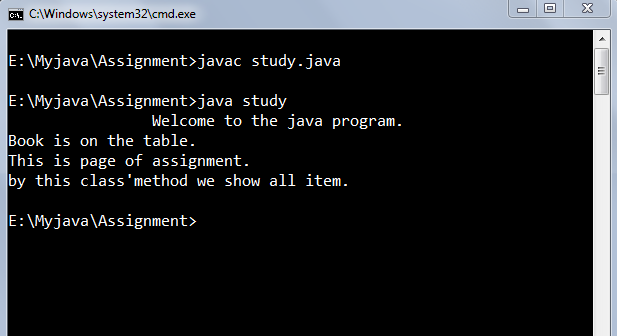
ob.assignment();

ob.display();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a simple class to handle exception in java.

class excpn

{

void mathexp()

{

int i=1,j=0;

try

{

int k=i/j;

System.out.println("This will never be printed.");

}

catch(ArithmeticException e)

{

System.out.println("Error occured i.e.:"+e);

}

}

}

class myexception

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to java program.");

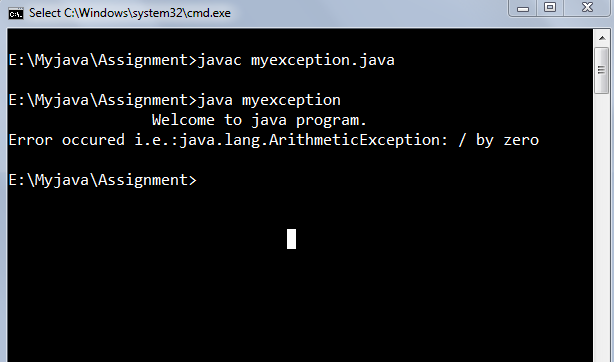
excpn ob=new excpn();

ob.mathexp();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a class to handle multiple-exception in java.

import java.io.\*;

class multiexce

{

int j;

multiexce(int a)

{

j=a;

}

void exce()

{

int i=1,a[]=new int[2];

try

{

int k=i/j;

a[3]=j;

System.out.println("This will never be printed.");

}

catch(ArithmeticException e)

{

System.out.println("Error occured i.e.:"+e);

}

catch(ArrayIndexOutOfBoundsException e)

{

System.out.println("Error occured i.e.:"+e);

}

}

}

class multiexception

{

public static void main(String args[])throws IOException

{

System.out.println("\t\tWelcome to java program.");

BufferedReader br=new BufferedReader(new InputStreamReader (System.in));

System.out.println("Enter a number:");

int a=Integer.parseInt(br.readLine());

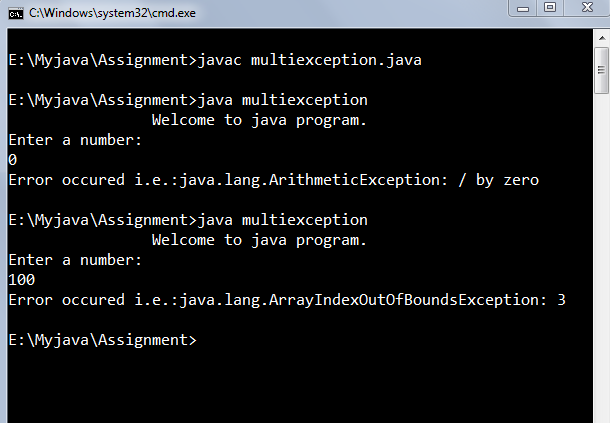
multiexce ob=new multiexce(a);

ob.exce();

}

}

## OUTPUT:-



## PROGRAM

## Program to create a class to handle exception by the help of finally keyword in

## java.

class finallyexception

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.\n");

int l=Integer.parseInt(args[0]);

int a=1,b;

int c[]=new int[2];

try

{

b=a/l;

c[3]=l;

}

finally

{

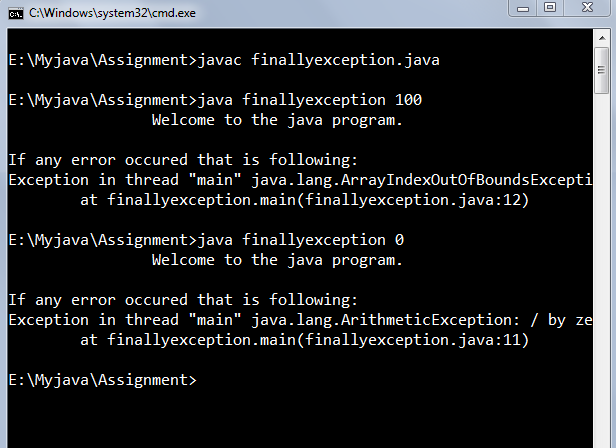
System.out.println("If any error occured that is following:");

}

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a thread in java.

import java.io.\*;

class threadclass extends Thread

{

threadclass(String s)

{

super(s);//Name for current thread.

System.out.println(this);//Name for current thread.

start();//Start the thread.

}

public void run()//This is the entry point for thread.

{

try

{

for(int i=1;i<=2;i++)

{

System.out.println(getName()+":"+i);

Thread.sleep(1000);//Pauses the current thread for one second.

}

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

System.out.println("Child threadclass exiting.");

}

}

class threadinterface implements Runnable

{

Thread t;

String name;//Name of the thread.

threadinterface(String s)

{

name=s;

t=new Thread(this,s);//This is for Reference to the child thread.

System.out.println(this);//Name of the current thread.

t.start();//Start the thread.

}

public void run()//This is the entry point for thread.

{

try

{

for(int i=1;i<=2;i++)

{

System.out.println(t.getName()+":"+i);

Thread.sleep(1000);//Pauses the current thread for one second.

}

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

System.out.println("Child threadinterface exiting.");

}

}

public class mythread

{

public static void main(String args[])throws IOException

{

choice c=new choice();

c.mychoice();

char ch;

m ob=new m();

do

{

BufferedReader br=new BufferedReader(new InputStreamReader

(System.in));

System.out.println("Enter ur choice:");

ch=(char)br.read();

switch(ch)

{

case 'c':

System.out.println("Execution by thread class.");

threadclass ob1=new threadclass("Child threadclass");

try

{

ob1.join();//Waiting for child thread to finish.

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

ob.thread();

break;

case 'i':

System.out.println("Executiion by thread interface.");

threadinterface ob2=new threadinterface("Child threadinterface");

try

{

ob2.t.join();//Waiting for child thread to finish.

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

ob.thread();

break;

case 'e':

br.close();

default:

System.out.println("Reenter ur choice:");

ch=(char)br.read();

}

}while(true);

}

}

class m

{

void thread()

{

try

{

for(int i=1;i<=2;i++)

{

System.out.println("Main Thread"+":"+i);

Thread.sleep(500);//Pauses the current thread for half second.

}

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

System.out.println("Main thread exiting.");

}

}

class choice

{

void mychoice()

{

System.out.println("\t\tWelcome to the java program.");

System.out.println("\t\tChoice menu is here.");

System.out.println("'c'->By Thread class");

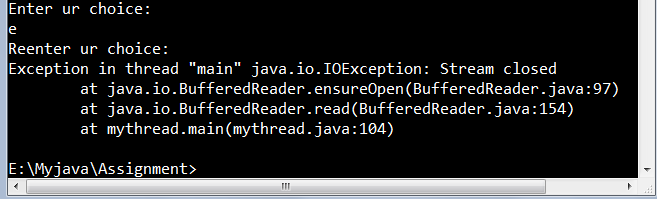
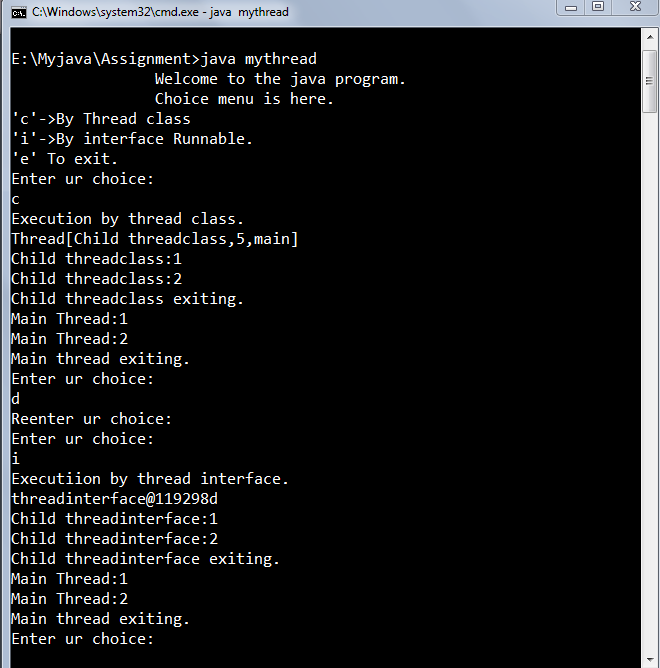
System.out.println("'i'->By interface Runnable.");

System.out.println("'e' To exit.");

}

}

## OUTPUT:-



## PROGRAM

## //Program to create multiple thread in java.

class threadclass extends Thread

{

threadclass(String s)

{

super(s);//Name for current thread.

System.out.println(this);//Name for current thread.

start();//Start the thread.

}

public void run()//This is the entry point for thread.

{

try

{

for(int i=1;i<=2;i++)

{

System.out.println(getName()+":"+i);

Thread.sleep(500);//Pauses the current thread for one second.

}

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

System.out.println(getName()+"exiting.");

}

}

public class multithread

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

threadclass ob1=new threadclass("FIRSRT THREAD");

threadclass ob2=new threadclass("SECOND THREAD");

threadclass ob3=new threadclass("THIRD THREAD");

try

{

ob1.join();//Waiting for thread to finish.

Ob2.join();//Waiting for thread to finish.

Ob3.join();//Waiting for thread to finish.

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

}

try

{

for(int i=1;i<=2;i++)

{

System.out.println("Main Thread"+":"+i);

Thread.sleep(500);//Pauses the current thread for half second.

}

}

catch(InterruptedException e)

{

System.out.println("Error occured that is:"+e);

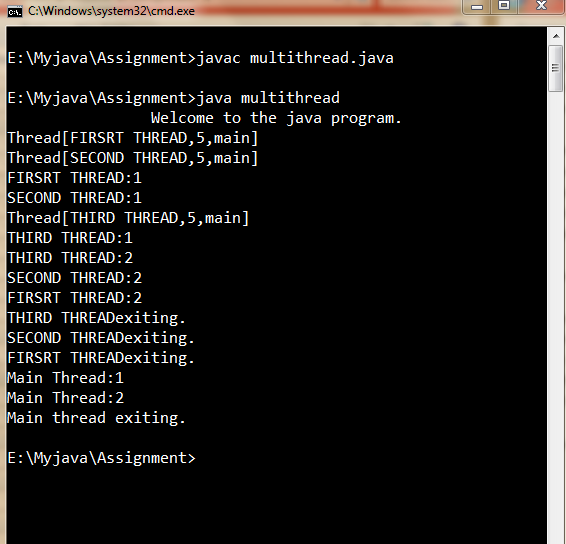
}

System.out.println("Main thread exiting.");

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a synchronozed thread in java.

class threadclass extends Thread

{

synch t;

threadclass(String s,synch th)

{

super(s);//Name for current thread.

t=th;

System.out.println(this);//Name for current thread.

start();//Start the thread.

}

public void run()//This is the entry point for thread.

{

t.display(getName());

}

}

class synch

{

synchronized void display(String s)

{

try

{

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

{

System.out.println(s+":"+i);

Thread.sleep(1000);//Pauses the current thread

for one second.

}

}

catch(InterruptedException e)

{}

}

}

public class mysynchronized

{

public static void main(String args[])

{

synch ob=new synch();

System.out.println("\t\tWelcome to the java program.");

threadclass ob1=new threadclass("FIRST THREAD",ob);

threadclass ob2=new threadclass("SECOND THREAD",ob);

try

{

ob1.join();

ob2.join();

}

catch(InterruptedException e)

{}

System.out.println("Main thread exiting.");

}

}

## OUTPUT:-

