

1. Develop a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c=0$. Read in a, b, c and use the quadratic formula. If the discriminant b^2-4ac is negative, display a message stating that there are no real solutions.

```
import java.util.Scanner;

class First{

    public static void main(String args[]){

        int a,b,c;

        double r1,r2,d;

        System.out.println("enter the coefficients of quadratic equation");

        Scanner s1= new Scanner(System.in);

        a=s1.nextInt();

        b=s1.nextInt();

        c=s1.nextInt();

        if(a==0){

            System.out.println("coefficients are invalid");

        }

        else{

            d=b*b-(4*a*c);

            if(d>0){

                System.out.println("it has real and distinct roots");

                r1=(-b+Math.sqrt(d))/(2*a);

                r2=(-b-Math.sqrt(d))/(2*a);

                System.out.println("the roots are "+r1+" and "+r2);

            }

            else if(d==0){

                System.out.println("it has real and equal roots");

                r1=(-b)/(2*a);

                System.out.println("the roots are "+r1+" and "+r1);

            }

            else{
```

```
System.out.println("it has no real roots");
```

```
}
```

```
}
```

```
}
```

```
}
```

Output:

```
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bmsce\Desktop\1BM22CS170>javac First.java

C:\Users\bmsce\Desktop\1BM22CS170>java First
enter the coefficients of quadratic equation
1
2
1
it has real and equal roots
the roots are -1.0 and -1.0

C:\Users\bmsce\Desktop\1BM22CS170>java First
enter the coefficients of quadratic equation
1
1
4
it has no real roots

C:\Users\bmsce\Desktop\1BM22CS170>java First
enter the coefficients of quadratic equation
1
4
1
it has real and distinct roots
the roots are -0.2679491924311228 and -3.732050807568877

C:\Users\bmsce\Desktop\1BM22CS170>_
```

2. Develop a Java program to create a class Student with members usn ,name ,and array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

```
import java.util.*;

class student6{

    int usn;

    int credits[]=new int[5];

    float marks[]=new float[5];

    float sgpa;

    void studdetails(int usn,int credits[],float marks[]){

        this.usn=usn;

        this.credits=credits;

        this.marks=marks;

    }

    void printdetails(){

        System.out.println("the usn of student is"+usn);

        System.out.println("the number of credits of subjects is");

        for(int i=0;i<5;i++){

            System.out.println(credits[i]);

        }

        System.out.println("the marks of student is");

        for(int i=0;i<5;i++){

            System.out.println(marks[i]);

        }

    }

    float studsgpa(){

        int sum=0;

        int creditssum=0;

        for(int i=0;i<5;i++){

            if(marks[i]>=90){
```

```

sum=sum+(credits[i]*10);
}
else if(marks[i]>=80){
sum=sum+(credits[i]*9);
}
else if(marks[i]>=70){
sum=sum+(credits[i]*8);
}
else if(marks[i]>=60){
sum=sum+(credits[i]*7);
}
else if(marks[i]>=50){
sum=sum+(credits[i]*6);
}
else if(marks[i]>=35){
sum=sum+(credits[i]*5);
}
else{
sum=sum+(credits[i]*0);
}
}
for(int i=0;i<5;i++){
creditssum=creditssum+credits[i];
}
sgpa=sum/(float)creditssum;
return sgpa;
}
}

class student5{
public static void main(String xx[]){
int usn;

```

```
int credits[]=new int[5];
float marks[]=new float[5];
student6 s=new student6();
Scanner s1=new Scanner(System.in);
System.out.println("enter the usn of student");
usn=s1.nextInt();
System.out.println("enter the number of credits of subjects");
for(int i=0;i<5;i++){
credits[i]=s1.nextInt();
}
System.out.println("enter the marks of student");
for(int i=0;i<5;i++){
marks[i]=s1.nextFloat();
}
s.studdetails(usn,credits,marks);
s.printdetails();
float sgpa=s.studsgpa();
System.out.println("the sgpa of student is"+sgpa);
}
}
```

Output:

```
C:\Windows\System32\cmd.e  X + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>javac student5.java

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>java student5
enter the usn of student
1
enter the number of credits of subjects
4
4
3
3
2
enter the marks of student
90
80
80
70
60
the usn of student is1
the number of credits of subjects is
4
4
3
3
2
the marks of student is
90.0
80.0
80.0
70.0
60.0

C:\Windows\System32\cmd.e  X + v

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>java student5
enter the usn of student
1
enter the number of credits of subjects
4
4
3
3
2
enter the marks of student
90
80
80
70
60
the usn of student is1
the number of credits of subjects is
4
4
3
3
2
the marks of student is
90.0
80.0
80.0
70.0
60.0
the sgpa of student is8.8125

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>
```

3. Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString() method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.Scanner;

class Books {
    String name,author;
    int price,numPages;

    Books(String name, String author, int price, int numPages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.numPages = numPages;
    }

    public String toString() {
        return "Book name: " + this.name + "\n" +
            "Author name : " + this.author + "\n" +
            "Price : " + this.price + "\n" +
            "Number of pages : " + this.numPages + "\n";
    }
}

class BookMain {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int n;

        String name, author;
        int price, numPages;

        System.out.println("Enter the number of books:");

        n = sc.nextInt();

        Books b[] = new Books[n];
```

```
System.out.println("Enter Name,author,price and number of pages:");  
for (int i = 0; i < n; i++) {  
    name = sc.next();  
    author = sc.next();  
    price = sc.nextInt();  
    numPages = sc.nextInt();  
    b[i] = new Books(name, author, price, numPages);  
}  
System.out.println("Book details:");  
for (int i = 0; i < n; i++) {  
    System.out.println(b[i].toString());  
}  
}  
}
```

Output:


```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>javac BookMain.java

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>java BookMain
Enter the number of books:
5
Enter Name,author,price and number of pages:
a
abc
250
200
b
bac
250
200
c
abc
250
200
d
bac
250
200
e
abc
250
200
Book details:
Book name: a
Author name : abc

Book name: b
Author name : bac
Price : 250
Number of pages : 200

Book name: c
Author name : abc
Price : 250
Number of pages : 200

Book name: d
Author name : bac
Price : 250
Number of pages : 200

Book name: e
Author name : abc
Price : 250
Number of pages : 200

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>
```

4. Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

```
abstract class shape{
protected int dimension1;
protected int dimension2;
public shape(int dimension1,int dimension2){
this.dimension1=dimension1;
this.dimension2=dimension2;
}
public abstract void printarea();
}
class rectangle extends shape{
public rectangle(int length,int width){
super(length,width);
}
public void printarea(){
double area=dimension1*dimension2;
System.out.println("area of rectangle:"+area);
}
}
class triangle extends shape{
public triangle(int base,int height){
super(base,height);
}
public void printarea(){
double area=0.5*dimension1*dimension2;
System.out.println("area of triangle:"+area);
}
```

```
}  
  
class circle extends shape{  
    public circle(int radius){  
        super(radius,0);  
    }  
  
    public void printarea(){  
        double area=Math.PI*dimension1*dimension1;  
        System.out.println("area of circle:"+area);  
    }  
}  
  
public class rect{  
    public static void main(String xx[]){  
        rectangle r=new rectangle(2,10);  
        triangle t=new triangle(3,7);  
        circle c=new circle(5);  
        r.printarea();  
        t.printarea();  
        c.printarea();  
    }  
}
```

Output:

```
C:\Windows\System32\cmd.e  X  +  v  -  □  X
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>javac rect.java

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>java rect
area of rectangle:20.0
area of triangle:10.5
area of circle:78.53981633974483

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>
```

5. Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- a) Accept deposit from customer and update the balance.
- b) Display the balance.
- c) Compute and deposit interest
- d) Permit withdrawal and update the balance Check for the minimum balance, impose penalty if necessary and update the balance.

```
import java.util.Scanner;
```

```
class account
```

```
{
```

```
    String name;
```

```
    int accno;
```

```
    String type;
```

```
    double balance;
```

```
    account(String name,int accno,String type,double balance)
```

```
    {
```

```
        this.name=name;
```

```
        this.accno=accno;
```

```
        this.type=type;
```

```
        this.balance=balance;
```

```
    }
```

```
    void deposit(double amount)
```

```
    {
```

```
        balance+=amount;
```

```

    }

    void withdraw(double amount)
    {
        if((balance-amount)>=0)
        {
            balance-=amount;
        }
        else
        {
            System.out.println("insufficient balance,cant withdraw");
        }
    }

    void display()
    {
        System.out.println("name:"+name+"accno:"+accno+"type:"+type+"balance:"+balance);
    }
}

class savAcct extends account
{

    private static double rate=5;

    savAcct(String name,int accno,double balance)
    {
        super(name,accno,"savings",balance);
    }

    void interest()
    {

```

```

        balance+=balance*(rate)/100;

        System.out.println("balance:"+balance);
    }

}

class curAcct extends account
{

    private double minBal=500;
    private double serviceCharges=50;

    curAcct(String name,int accno,double balance)
    {
        super(name,accno,"current",balance);
    }

    void checkmin()
    {

        if(balance<minBal)
        {
            System.out.println("balance is less than min balance,service charges
            imposed: "+serviceCharges);
            balance-=serviceCharges;
            System.out.println("balance is:"+balance);
        }
    }

}

```

```

}
class accountMain
{
    public static void main(String a[])
    {
        Scanner s=new Scanner(System.in);
        System.out.println("enter the name :");
        String name=s.next();
        System.out.println("enter the type(current/savings):");
        String type=s.next();
        System.out.println("enter the account number:");
        int accno=s.nextInt();
        System.out.println("enter the intial balance:");
        double balance=s.nextDouble();
        int ch;
        double amount1,amount2;
        account acc=new account(name,accno,type,balance);
        savAcct sa=new savAcct(name,accno,balance);
        curAcct ca=new curAcct(name,accno,balance);
        while(true)
        {
            if(acc.type.equals("savings"))
            {
                System.out.println("\nMenu\n1.deposit 2.withdraw 3.compute
                interest 4.display");
                System.out.println("enter the choice:");
                ch=s.nextInt();
                switch(ch)
                {
                    case 1:System.out.println("enter the amount:");

```



```

        amount1=s.nextInt();
        sa.deposit(amount1);
        break;
    case 2:System.out.println("enter the amount:");
        amount2=s.nextInt();
        sa.withdraw(amount2);
        break;
    case 3:sa.interest();
        break;
    case 4:sa.display();
        break;
    case 5:System.exit(0);
    default:System.out.println("invalid input");
        break;
    }
}
else
{
    System.out.println("\nMenu\n1.deposit 2.withdraw 3.display");
    System.out.println("enter the choice:");
    ch=s.nextInt();
    switch(ch)
    {
        case 1:System.out.println("enter the amount:");
            amount1=s.nextInt();
            ca.deposit(amount1);
            break;
        case 2:System.out.println("enter the amount:");
            amount2=s.nextInt();
            ca.withdraw(amount2);
            ca.checkmin();

```

```
break;
```

```
case 3:ca.display();
```

```
break;
```

```
case 4:System.exit(0);
```

```
default:System.out.println("invalid input");
```

```
break;
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

Output:

```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\lBM22CS170>javac accountMain.java

C:\Users\saisr\OneDrive\Desktop\lBM22CS170>java accountMain
enter the name :
abcd
enter the type(current/savings):
savings
enter the account number:
170
enter the intial balance:
1000

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
2
enter the amount:
100

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
3
balance:945.0

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
4

enter the type(current/savings):
savings
enter the account number:
170
enter the intial balance:
1000

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
2
enter the amount:
100

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
3
balance:945.0

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
4
name:abcdaccno:170type:savingsbalance:945.0

Menu
1.deposit 2.withdraw 3.compute interest 4.display
enter the choice:
5

C:\Users\saisr\OneDrive\Desktop\lBM22CS170>
```

6. Create a package CIE which has two classes - Student and Internals. The class Student has members like usn, name, sem. The class Internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

```
package cie;

import java.util.Scanner;

public class student{

    String usn;

    String name;

    int semester;

    public void studdetails(){

        Scanner s1=new Scanner(System.in);

        System.out.println("enter the usn of student");

        usn=s1.next();

        System.out.println("enter the name of student");

        name=s1.next();

        System.out.println("enter the semester of student");

        semester=s1.nextInt();

    }

    public void printdetails(){

        System.out.println("the usn of student is"+usn);

        System.out.println("the name of student is"+name);

        System.out.println("the semester of student is"+semester);

    }

}
```

```
package cie;

import java.util.Scanner;
```

```

public class internals extends student{
public float studmarks[]=new float[5];
public void studmarks(){
Scanner s1=new Scanner(System.in);
System.out.println("enter the internal marks of student");
for(int i=0;i<5;i++){
studmarks[i]=s1.nextFloat();
}
}
public void printmarks(){
System.out.println("the marks of student is");
for(int i=0;i<5;i++){
System.out.println(studmarks[i]);
}
}
}

```

```

package see;
import java.util.Scanner;
import cie.student;
public class external extends student{
public float seemarks[]=new float[5];
public void seemarks(){
Scanner s2=new Scanner(System.in);
System.out.println("enter the see marks of student");
for(int i=0;i<5;i++){
seemarks[i]=s2.nextFloat();
}
}
public void printseemarks(){
System.out.println("the see marks of student is");
}
}

```

```

for(int i=0;i<5;i++){
System.out.println(seemarks[i]);
}
}
}

```

```

import java.util.Scanner;
import cie.student;
import cie.internals;
import see.external;

class main{
public static void main(String xx[]){
int n;
Scanner s3=new Scanner(System.in);
System.out.println("enter the number of students");
n=s3.nextInt();
internals internalmarks[]=new internals[n];
external externalmarks[]=new external[n];
for(int i=0;i<n;i++){
internalmarks[i]=new internals();
externalmarks[i]=new external();
internalmarks[i].studdetails();
internalmarks[i].studmarks();
externalmarks[i].seemarks();
}
float finalmarks[][]=new float[n][5];
for(int i=0;i<n;i++){
for(int j=0;j<5;j++){
finalmarks[i][j]=internalmarks[i].studmarks[j]+((externalmarks[i].seemarks[j])/2);
}
}
}

```

```
for(int i=0;i<n;i++){  
    internalmarks[i].printdetails();  
    internalmarks[i].printmarks();  
    externalmarks[i].printseemarks();  
}  
for(int i=0;i<n;i++){  
    System.out.println("the final marks of "+i+" student is");  
    for(int j=0;j<5;j++){  
        System.out.println(finalmarks[i][j]);  
    }  
}  
}  
}
```

Output:

```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\javal>javac main.java

C:\Users\saisr\OneDrive\Desktop\javal>java main
enter the number of students
2
enter the usn of student
1
enter the name of student
a
enter the semester of student
3
enter the internal marks of student
40
40
40
40
40
enter the see marks of student
80
80
80
80
80
enter the usn of student
2
enter the name of student
b
enter the semester of student
3

enter the usn of student
2
enter the name of student
b
enter the semester of student
3
enter the internal marks of student
45
45
45
45
45
enter the see marks of student
80
80
80
80
80
the usn of student is1
the name of student isa
the semester of student is3
the marks of student is
40.0
40.0
40.0
40.0
40.0
the see marks of student is
80.0
80.0
80.0
80.0
```



```
C:\Windows\System32\cmd.e x + v
the see marks of student is
80.0
80.0
80.0
80.0
80.0
the usn of student is2
the name of student isb
the semester of student is3
the marks of student is
45.0
45.0
45.0
45.0
45.0
the see marks of student is
80.0
80.0
80.0
80.0
80.0
the final marks of 0 student is
80.0
80.0
80.0
80.0
80.0
the final marks of 1 student is
85.0
85.0
85.0
85.0

C:\Windows\System32\cmd.e x + v
80.0
80.0
80.0
the usn of student is2
the name of student isb
the semester of student is3
the marks of student is
45.0
45.0
45.0
45.0
45.0
the see marks of student is
80.0
80.0
80.0
80.0
80.0
the final marks of 0 student is
80.0
80.0
80.0
80.0
80.0
the final marks of 1 student is
85.0
85.0
85.0
85.0

C:\Users\saisr\OneDrive\Desktop\javal>
```

7. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge() when the input age=father's age.

```
class WrongAge extends Exception{
    public WrongAge(String Message){
        super(Message);
    }
}

class Father{
    private int age;
    public Father(int age) throws WrongAge{
        if(age<0){
            throw new WrongAge("age cannot be negative");
        }
        this.age=age;
    }
    public int getAge(){
        return age;
    }
}

class Son extends Father{
    private int sonage;
    public Son(int fatherage,int sonage) throws WrongAge{
        super(fatherage);
        if(sonage>=fatherage){
            throw new WrongAge("son's age cannot be greater than or equal to father's age");
        }
        this.sonage=sonage;
    }
}
```

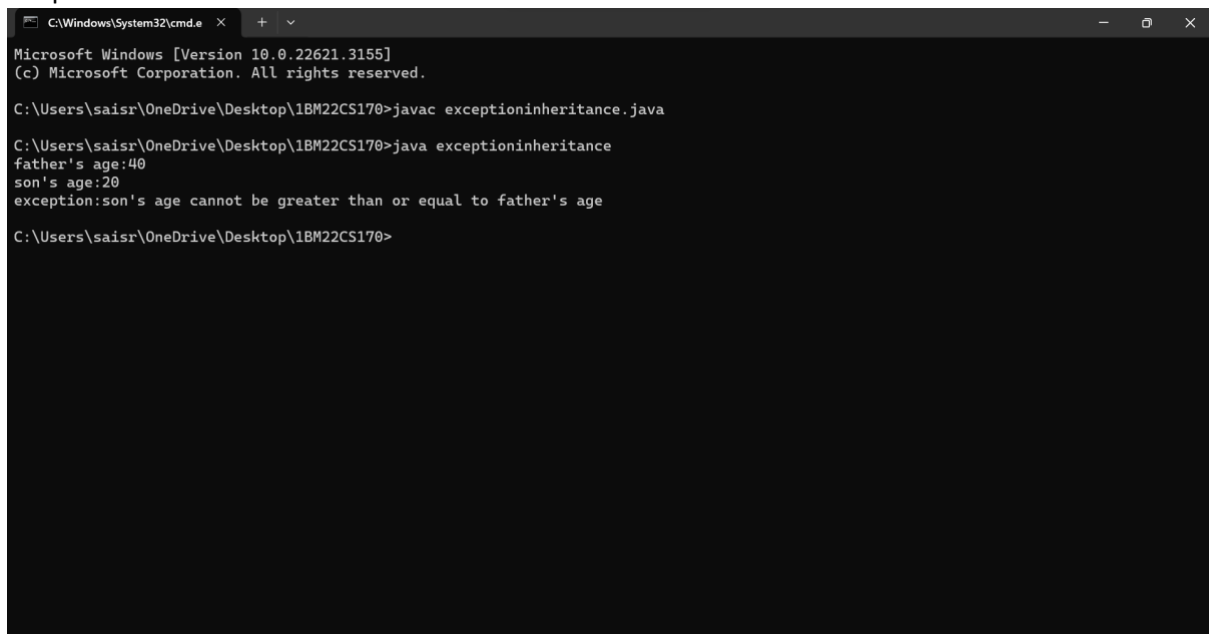
```

public int getsonage(){
return sonage;
}
}

public class exceptioninheritance{
public static void main(String xx[]){
try{
son son1=new son(40,20);
System.out.println("father's age:"+son1.getage());
System.out.println("son's age:"+son1.getsonage());
son invalidson=new son(30,35);
System.out.println("this line will not be reached");
}
catch(wrongage e){
System.out.println("exception:"+e.getMessage());
}
}
}
}

```

Output:



```

C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>javac exceptioninheritance.java
C:\Users\saisr\OneDrive\Desktop\1BM22CS170>java exceptioninheritance
father's age:40
son's age:20
exception:son's age cannot be greater than or equal to father's age
C:\Users\saisr\OneDrive\Desktop\1BM22CS170>

```

8.write a program which creates two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

```
class DisplayMessageThread extends Thread {  
    private final String message;  
    private final long interval; // in milliseconds  
  
    DisplayMessageThread(String message, long interval) {  
        this.message = message;  
        this.interval = interval;  
    }  
  
    public void run() {  
        try {  
            while (true) {  
                System.out.println(message);  
                Thread.sleep(interval);  
            }  
        } catch (InterruptedException e) {  
            System.out.println(Thread.currentThread().getName() + " interrupted.");  
        }  
    }  
}  
  
public class TwoThreadDemo {  
    public static void main(String[] args) {  
        DisplayMessageThread thread1 = new DisplayMessageThread("BMS College of Engineering",  
10000); // 10 seconds  
        DisplayMessageThread thread2 = new DisplayMessageThread("CSE", 2000); // 2 seconds  
  
        thread1.setName("Thread 1");  
        thread2.setName("Thread 2");  
    }  
}
```

```

thread1.start();

thread2.start();

try {

    // Let the threads run for a while

    Thread.sleep(30000); // Let the program run for 30 seconds
} catch (InterruptedException e) {

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

}

// Interrupt both threads to stop them

thread1.interrupt();

thread2.interrupt();

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

}

}

```

Output:

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>javac TwoThreadDemo.java

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>java TwoThreadDemo
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
Main thread exiting.
Thread 1 interrupted.
Thread 2 interrupted.

C:\Users\saisr\OneDrive\Desktop\1BM22CS170>

```

9.write a program thar creates a user interface to perform integer divisions. the user enters two numbers in the text fields, Num1 and Num2. the division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were zero, the program would throw an Arithmetic Exception display the exception in a message dialog box.

```
import java.awt.*;
```

```
import java.awt.event.*;
```

```
public class DivisionMain1 extends Frame implements ActionListener
```

```
{
```

```
    TextField num1,num2;
```

```
    Button dResult;
```

```
    Label outResult;
```

```
    String out="";
```

```
    double resultNum;
```

```
    int flag=0;
```

```
    public DivisionMain1()
```

```
    {
```

```
        setLayout(new FlowLayout());
```

```
        dResult = new Button("RESULT");
```

```
        Label number1 = new Label("Number 1:",Label.RIGHT);
```

```
        Label number2 = new Label("Number 2:",Label.RIGHT);
```

```
        num1=new TextField(5);
```

```
        num2=new TextField(5);
```

```
        outResult = new Label("Result:",Label.RIGHT);
```

```
        add(number1);
```

```
        add(num1);
```

```
        add(number2);
```

```

        add(num2);
        add(dResult);
        add(outResult);

        num1.addActionListener(this);
        num2.addActionListener(this);
        dResult.addActionListener(this);
        addWindowListener(new WindowAdapter()
        {
            public void windowClosing(WindowEvent we)
            {
                System.exit(0);
            }
        });
    }

    public void actionPerformed(ActionEvent ae)
    {
        int n1,n2;
        try
        {
            if (ae.getSource() == dResult)
            {
                n1=Integer.parseInt(num1.getText());
                n2=Integer.parseInt(num2.getText());

                /*if(n2==0)
                    throw new ArithmeticException();*/

                out=n1+" "+n2;
                resultNum=n1/n2;
                out+=String.valueOf(resultNum);
            }
        }
        catch (Exception e)
        {
            out="Error!";
        }
    }
}

```

```

        repaint();

    }

}

catch(NumberFormatException e1)
{
    flag=1;
    out="Number Format Exception! "+e1;
    repaint();
}

catch(ArithmeticException e2)
{
    flag=1;
    out="Divide by 0 Exception! "+e2;
    repaint();
}

}

public void paint(Graphics g)
{
    if(flag==0)

        g.drawString(out,outResult.getX()+outResult.getWidth(),outResult.getY()+outResult.getHeight()-8);

    else
        g.drawString(out,100,200);
    flag=0;
}

public static void main(String[] args)
{

```



```
        DivisionMain1 dm=new DivisionMain1();  
        dm.setSize(new Dimension(800,400));  
        dm.setTitle("DivionOfIntegers");  
        dm.setVisible(true);  
    }  
  
}
```

Output:

DivionOfIntegers

Number 1: Number 2: Result: 4 22.0

DivionOfIntegers

Number 1: Number 2: Result:

Divide by 0 Exception! java.lang.ArithmeticException: / by zero

DivionOfIntegers

Number 1: 4

Number 2: a

RESULT

Result:

Number Format Exception! java.lang.NumberFormatException: For input string: "a"