DATE: / /
PAGE NO: Usha Gold

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;
import java.lang.Math.*;
public class quadratic{
    public static void main(String[] args){
        float a, b, c;
        Scanner s=new Scanner(System.in);
        System.out.println("enter value of a, b and c");
        a=S.nextFloat();
        b=S.nextFloat();
        c=S.nextFloat();
        Float root1, root2;
        Float desc=(float)(Pow(b,2)-4*a*c);
        if(desc>0){
            root1=(float)(-b/(2*a)-Sqrt(desc)/(2*a));
            root2=(float)(-b/(2*a)+Sqrt(desc)/(2*a));
            System.out.println("root1="+root1+"\n root2="+root2);
        }else if(desc<0){
            System.out.println(" no real solution exists");
        }else{
            System.out.println("root="+(-b/(2*a)));
        }}}
```

Output

enter the values of a, b & c

1 2 3

equation has no real solution

write a java program to create a class Employee with members empid, empname, empnohrs, emphasic, emphra (%), empda(%), empit(%), empgross.

include methods to do the following

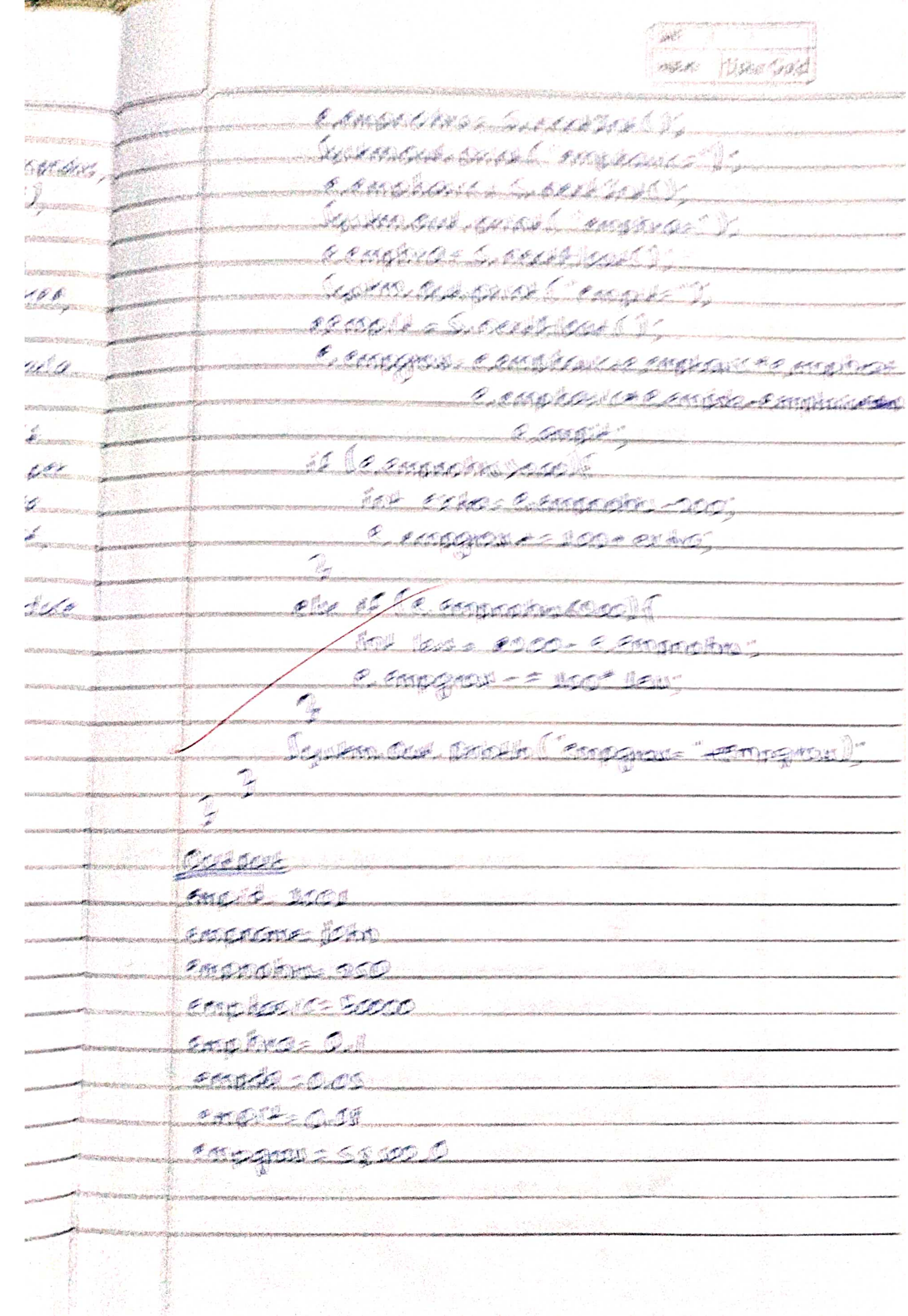
i) accept all values from the user. Note HRA, DA and IT are given in (%)

ii) Calculate the gross salary based on formula

empgross = emphasic + empbasic*emphra + empbasic + empda - empbasic*empit

iii) Consider the overtime amount to be Rs 100 per hour. if empnohrs > 200, for every hour the employee is to be given additional payment. Calculate the additional payment and update the gross. if empnohrs < 200, reduce Rs 100 per hour and update the gross.

```
import java.util.Scanner;
class employee{
    int empid, empnohrs, emp basic;
    String empname;
    float emphra, empda, empit, empgross;
}
public class emp {
    public static void main(String[] args){
        Scanner S = new Scanner(System.in);
        employee e = new employee();
        System.out.print("empid = ");
        e.empid = S.nextInt();
        System.out.print("empname= ");
        e.empname = S.next();
        System.out.print("empnohrs");
```



Q) Develop a java program to create a class student with members usn, name, an array credit and an array marks. Include methods to accept & display details & a method to calculate SGPA of a student.

=>
```
import java.util.Scanner;
class func{
    float sgpa;
    public void calcsgpa(int marks[], int credit[]){
    {
      int sum=0;
      int num=0;
      for(int i=0; i<5; i++){
        sum+=credit[i];
      }
      for(int i=0; i<5; i++){
        int reduce;
        if(marks[i]>90 && marks[i]<=100){
          reduce=10;
        }
        else if(marks[i]>80 && marks[i]<=90){
          reduce=9;
        }
        else if(marks[i]>70 && marks[i]<=80){
          reduce=8;
        }
        else if(marks[i]>60 && mark[i]<=70){
          reduce=7;
        }
        else if(mark[i]>50 && marks[i]<=60){
          reduce=6;
        }
```

```
else if (marks[i]>=90 && marks[i]<=50){
    reduce=90;
}
num+= credit[i]*reduce;
}
sgpa=(float)num/sum;
}
public float getsgpa(){
    return sgpa;
}
}
class student{
    public static void main(String[] args){
        String usn, name;
        int credit[]=new int[5];
        int marks[]=new int[5];
        Scanner S= new Scanner(System.in);
        System.out.println("Enter the usn");
        usn= S.next();
        System.out.println("Enter the name:");
        name= S.next();
        System.out.println("Enter the marks in
                            marks array:");
        for(int i=0; i<5; i++){
            System.out.print("marks["+i+"]=");
            marks[i]= S.nextInt();
        }
        System.out.println("Enter the ~~marks~~ credits in credit
                            array");
        for(int j=0 ; j<5; j++){
            System.out.print("credit["+j+"]=");
            credit[j]=S.nextInt();
        }
```

```
        func f = new func();
        f.calcsgpa(marks, credit);
        System.out.println(f.getsgpa());
    }
}
```

Output

```
Enter the USN 123
Enter the name : kkk
Enter the marks in the marks array:
marks[0]=99
marks[1]=99
marks[2]=99
marks[3]=99
marks[4]=9
Enter the credits in the credit array:
credit[0]=5
credit[1]=5
credit[2]=5
credit[3]=5
credit[4]=5
8.8
```

11.24

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 details of 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, num_pages;
    void setDetails(String name, String author,
                    int price, int num_pages){
        this.price= price;
        this.num_pages=num_pages;
        this.name = name;
        this.author= author;
    }
    public String toString(){
        return ("name:" + name+"\n author"+author+
                "\n.price"+price+"\n num_pages"+
                num_pages); }
}
public class First(){
    public static void main(String [] args){
        book b1[]= new book[4];
        Scanner s= new Scanner(System.in);
        for (int i=0; i<4; i++){
        System.out.println(" name = ");
        String name = s.next();
        String author= s.next();
```

```
        System.out.println("price=");
        int price = S.nextInt();
        System.out.println("num_pages=");
        int num_pages = S.nextInt();
        b1[i] = new book();
        b1[i].setDetails(name, author, price, num_pages);
    }
    System.out.println("Display complete details");
    for(int i=0; i<4; i++){
        System.out.println(b1[i]);
    }
  }
}
```

output

```
name = A
author = a
price = 99
num_pages = 199
name = B
author = b
price = 199
num_pages = 299
name = C
author = c
price = 299
num_pages = 399
name = D
author = d
price = 399
num_pages = 499
```

DATE / /
PAGE NO. Usha Gold

Display complete details

name = A
author = a
price = 99
num_pages = 199
name = B
author = b
price = 199
num_pages = 299
name = C
author = c
price = 299
num_pages = 399
name = D
author = d
price = 399
num_pages = 499

```
import java.util.Scanner;
abstract class Shape{
    int lengthSide1;
    int breadthSide2;
    abstract void printarea();
}
class Rectangle extends Shape{
    void printarea(){
        System.out.println
```

9/1/24

DATE / /
PAGE NO. Usha Gold

```
import java.util.Scanner;
abstract class shape{
        int side1;
        int side2;
        abstract void printarea();
}
class Rectangle{
        public shape(int side1, int side2){
                this.side1 = side1;
                this.side2 = side2;
        }
    public abstract void printarea();
}
class Rectangle extends shape{
    public Rectangle(int length, int breadth){
            super(length, breadth);
    }
    double area;
    public void printarea(){
        area = side1 * side2;
        System.out.println("area = " + area);
    }

class Triangle extends shape{
    public Triangle extends shape{
    public Triangle(int base, int height){
            super(base, height);
    }
    public void printarea(){
        double area = (0.5) * side1 * side2;
        System.out.println("area = " + area);
    }
}
```

```
class circle extends shape{
    public circle (int radius){
        super( radius, 0);
    }
    print a
    public void printarea(){
        double area = (3.14)*(side1)*(side1);
        System.out.println("area = "+area);
    }
}

public class Main{
    public static void main (String[] args){
        Rectangle r = new Rectangle()
        System.out.println("Enter length of rectangle");
        Scanner s = new Scanner(System.in);
        int length = s.nextInt();
        System.out.println("Enter breadth of rectangle");
        int breadth = s.nextInt();
        Rectangle r = new Rectangle(length, breadth);
        System.out.println("Enter base and height of
                            Triangle");
        int base = s.nextInt();
        int height = s.nextInt();
        Triangle t = new Triangle(base, height);
        System.out.println("Enter radius of circle");
        int radius = s.nextInt();
        Circle c = new Circle(radius);
}
```

DATE / /
PAGE NO. Usha Gold

output:

Enter length of rectangle
4
Enter breadth of rectangle
5
Area = 20
Enter base and height of rectangle
3 8
Area: 12.0
Enter radius of circle
Area = 113.09733851

```
import java.util.Scanner;

class wrongage extends Exception {
    public String toString(){
        return "age cannot be less than 0";
    }
}

class cannotbe extends Exception {
    public String toString(){
        return "son cannot be older than father";
    }
}

class father{
    int age;
    father(int age) throws wrongage {
        this.age=age;
        if (age<0){
            throw new wrongage();
        }
    }
}

class son extends father{
    int fatherage, sonage;
    son(int fatherage, int sonage)
        throws wrongage, cannotbe{
        super(sonage);
        this.father= father;
        this.son = son;
```

```
if (son > father){
        throw new cannotbe();
    }
    else{
        System.out.println("father age:"+
                father age+"\n son age:"+son age);
    }
}
}
class Main{
    public static void main(String[] args){
        int father, son;
        Scanner S=new Scanner(System.in);
        try{
            System.out.println("Enter father age");
            father = S.nextInt();
            son
            father f= new father(father);
            System.out.println("Enter son age:");
            son = S.nextInt();
            son S= new son(son);
        }
        catch (wrongage | cannbe e){
            System.out.println("Exception:"+
                    e);
        }
    }
}
```

Enter father age:
45
Enter son age:-2
Exception: Age cannot be less than 0.

```
Package CIE
public class student{
    public String usn;
    public String name;
    public int sem;

    public student(String usn, String name, int sem){
        this.usn=usn;
        this.name=name;
        this.sem= sem;
    }
}
```

```
package CIE
public class Internals{
    public int [] marks= new int [5];
    public Internals(int [] marks){
        if( marks.length==5){
            this.marks= marks;
        }
        else{
            System.out.println("please provide marks
                for all five courses");
        }
    }
}
```

```
package see
import CIE.student;
public class external extends student{
    public int [] marks = new int [5];
    public external (String usn, String name, int sem,
                     int marks[]){
        super(usn, name, sem);
        if (marks.length == 5){
            this.marks = marks;
        }
        else{
            System.out.println("please provide marks
                               for all courses");
        }
    }
}
CIE.student;
CIE.Internals;
SEE.Externals;
class final marks{
    public static void main (String [] args){
        student [] students = new student [2];

        int[] student1internalmarks = {80, 75, 85, 90, 70};
        int[] student1SEEmarks = {70, 65, 75, 80, 60};
        students[0] = new external ("1BM22CS001", "Alice",
                                    5, student1SEEMarks);
        Internals student1Internals = new Internals(
                                    student1Internalmarks);
```

DATE / /
PAGE NO: Usha Gold

```
int[] student2 Internal marks = {85, 80, 90, 85, 75};
int[] student2 SEE Marks = {75, 70, 80, 85, 65};
student[1] = new external ("1BM22CS002", "John", 5,
                    student SEE marks);
Internals student2 Internals = new Inkrnals(student2 Internal
                                            marks);
for (int j=0; j<5; j++){
    finalmarks[j] = student1 Internalmarks[j] +
                    student
}

for (int i=0; i< students.length; i++){
    int[] final marks = new int[5]
    System.out.println(" student :"+students[i].name);
    System.out.println(" usn:" + students[i].usn);
    System.out.println("Semester:"+students[i].sem);

    for(int j=0; j<5; j++){
        finalmarks[j] = studentInternals.marks[j]+
        students[i].marks[j];
        System.out.println("finalmarks :"+(j+1)+":"+
                           finalmarks[j]);
    }
    System.out.println();
}
}
```

```
class Generic<A, B, C, D, E> {
    A a;
    B b;
    C c;
    D d;
    E e;

    Generic(A a1, B b1, C c1, D d1, E e1){
        a = a1;
        b = b1;
        c = c1;
        d = d1;
        e = e1;
    }
void showtype(){
    System.out.println(a.getClass().getNam
    System.out.println(b.getClass().getNa
    System.out.println(c.getClass().getNam
    System.out.println(d.getClass().getName
    System.out.println(e.getClass().getNam
}
A returntype(){
    return a;}
B returntype1(){
    return b;
}
C returntype2(){
    returne c;
}
D returntype3(){
    return d;
}
```

```
T returntype4(){
    return e;
}
}

class Generic demo{
    public static void main(String[] args){
        Generic<Integer, Float, String, Double, Boolean>
                g1;
        g1 = new Generic<Integer, Float, String,
                Double, Boolean>
            (1, (float) 1.233, "BMS", 2.1133, true);
        g1.showtype();
        int x = g1.returntype();
        System.out.println(x);
        Float x1 = g1.returntype();
        System.out.println(x1);
        String x2 = g1.returntype();
        System.out.println(x2);
        Double x3 = g1.returntype3();
        System.out.println(x3);
        Boolean x4 = g1.returntype4();
        System.out.println(x4);
    }
}
```

Output:

java.lang.Integer
java.lang.Float
java.lang.String
java.lang.Double
java.lang.Boolean

1

1 2322

BMS

2 1 1332

true

9) write a program to show user defined exception. Create class employee use inheritance. Throw exception when manager salary is less than employee

```
class Myexception extends Exception {
    public String tostring() {
        return "Mangr salary cannot be less
                than worker";
    }
}
class Manager {
    int Manager salary;
    public Manager(int managersalary) {
        this.managersalar = manager salary;
    }
}
class worker extends Manager {
    int workersalary;
    public worker(int workersalary, int Mangersalay
            throws myexception {
        super (managersalary);
        if (managersalay < workersalary) {
            throw new myexception ();
        }
    }
}
public class Employee {
    public static void main (String [] args) {
    try {
        worker w = new worker (90000, 30000);
    }
    catch (Myexception e) {
        System.out.println(e);
    }
}
}
```

Q) Write a program which creates two threads, One displaying "BMS College of Engineering" Once every ten seconds and another displaying "CSE" once every two seconds

=> Class DisplayThread extends Thread {
    private String message;
    private int interval;
    public DisplayThread(String message, int interval) {
        this.message = message;
        this.interval = interval;
    }
    public void run() {
        while (true) {
            try {
                System.out.println(message);
                thread.sleep(interval);
            }
            catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}

Class Display message {
    public static void main(String[] args) {
        Thread bmsThread = new DisplayThread
            ("BMS College", 10000);
        Thread cseThread = new DisplayThread
            ("CSE", 2000);
    }
}

```
        bms Thread. start ();
        Cse Thread. start ();
    }
}
```

OUTPUT

BMS collage
CSE
CSE
CSE
CSE
CSE
BMS collage
CSE
CSE
CSE
CSE
CSE
BMS collage

8) Develop a java program to create a class Bank that maintains two kind of accounts for its customers, one called saving account & the other current account. the savings account provides compound interest and withdrawal facility but no cheque book facility. The current account provides cheque book facility no interest. Current account holders should also maintain 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 & 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 & update the balance.

b) display the balance

c) compute & deposite interest

d) permit withdrawal & update the balance.

check for minimum balance, impose penalty & update the balance.

```
import java.util.scanner;
class Bank{
    double accno;
    String name;
    String type;
    double balance;
    public Bank(double accno, String name,
            String type, double balance){
        this.accno= accno;
        this.name= name;
        this.type= type;
        this.balance= balance;
    }
public void display(){
    System.out.println("current balance
            is: "+ balance);
    }
public void deposit(double deposit){
    this.balance+= deposit;
    System.out.println("updated balance
            is "+ balance);
    }
public void withdraw(double amount){
    balance-= amount;
if( balance<amount){
    System.out.println("insufficient balance");
    }
else{
    balance-= amount;
    System.out.println("After withdrawing,
        the balance is: "+ balance);
    }
}
```

```
class saving extends Book {
    double rate;
    int time;
    public saving (double accno, String
    name, double balance, int time,
        double rate){
        super(accno, name, "saving" balance);
        this.time = time;
        this.rate = rate;
    }
    public void calculateInterest(){
        balance += (balance * time * rate)/100;
    }
}
class Current extends Bank{
    double minBalance;
        public current (double accno, String
        name, double balance, double minbal
    {   super(accno, name, "Current", balance
        this.minBalance = min balance;
    }
    public void ApplyServiceCharge(){
        if (balance < min balance){
            System.out.println("Servile
            Charge of 5% is applied:");
            balance -= balance * 0.05;
        }
    }
}
```

```
class Main{
    public static void main(String [] args){
        double accno;
        String name;
        double balance;
        Scanner s= new Scanner(system.in)
        System.out.println("Enter accno:");
        accno = s.nextDouble();
        System.out.println("Enter name");
        name = s.next();
        System.out.println("Enter Balance");
        balance= s.nextDouble();
        System.out.println("time of loan");
        ~~balance~~ time = s.nextInt();
        System.out.println("Enter balance rate");
        double rate = s.nextDouble();
        System.out.println(" min balance");
        double min_balance= s.nextDouble();
        System.out.println(" 1, current account
                2. saving account :");
        int ch=s.nextInt();
        if(ch==1){
            Saving s1= new Saving(accno, name, balance,
                    time, rate);
            System.out.println("Depositing 1000rs");
            s1.deposit(1000);
            System.out.println("withdrawing 600rs");
            s1.withdraw(500);
            s1.calculate interest();
            System.out.println("Displaying balance");
            s1.display();
        }
```

```
else if (ch==2){
    Current c= new current(accno, name,
                balance, min_balance);
    System.out.println("Depositing 1000rs");
    c.deposit(1000);
    System.out.println("withdrawing 500rs");
    c.withdraw(500);
    c.applyServiceCharge();
    System.out.println("Displaying balance");
    c.displa();
}
S.close();
}
}
```

Output

```
Enter accno:
123343
Enter name
Malinga
Enter balance
5000
Enter time of loan:
2
Enter balance rate:
5
Enter min balance
1000
Do you want current account (1) or saving account(2)
1
Depositing 1000rs
updated balance is 6000.0
withdrawing 500rs
```

After withdrawing balance is: 5500.0
Displaying balance
current balance is 6050.0

DATE: / /
PAGE NO: Usha Gold

```
import java.awt.*
import java.awt.event.*;
public class Division main extends Frame
    implements action listener {
      TextField num1, num2;
      Button dresult; label out Result;
      String out = " ";
      double result+num;
      int flag=0;
    public Divisionmain(){
      setlayout (New Flow layout());
      dResult = new Button("Result");
      label number1= new label("Number1:";
                                label1);
    label number2= new label("Number2:", label2);
    num1 = newTextfield (5);
    num2 = new TextField (5);
    outResult = new label ("Result", label Right);
    add(number1);
    add (num1);
    add(number2);
     add(num2);
     add (dresult);
     add (out Result);
    num1.addAction listener( this);
    num2.addAction listener(this);
    dResult.add Action listener (this);
    add window listener (new window Adapter()
  { public Void window( Losing c:window
                  Event wc) {system exit (0);}
```

```
public void actionPerformed(ActionEvent ae)
{
  double n1, n2;
  try{
    if(ae.getSource()==dresult){
    n1=doubleparseDouble(num1.getText());
    n2=DoubleparseDouble(num2.getText());
    out+=n1+" "+n2;
    out+=String.valueOf(result,Num);
    repaint()
  }}
catch(ArithmaticException er){
    flag=1
    out="Divide By 0 exception!"+e;
    repaint();
  }
  catch(numberFormatException e1){
    flag=1;
    out="Number format Exception)"+e1;
    repraint();
  }
}
public void print(Graphics g){
  if(flag==0)
  g.DrawString(out, outResult.getX()+
      out.Result.getwidth(), out.Result.
        getrigHeight()-8);
  else{
      g.DrawString(out, 160, 200);
        flag=0;
    }
}
```

DATE: / /
PAGE NO: Usha Gold

```
public static void main (String args[])
    Division main dm = new Division main ();
    dm.setSize(new Dimension (800,900));
    dm.setTitle ("Division of Integers");
```