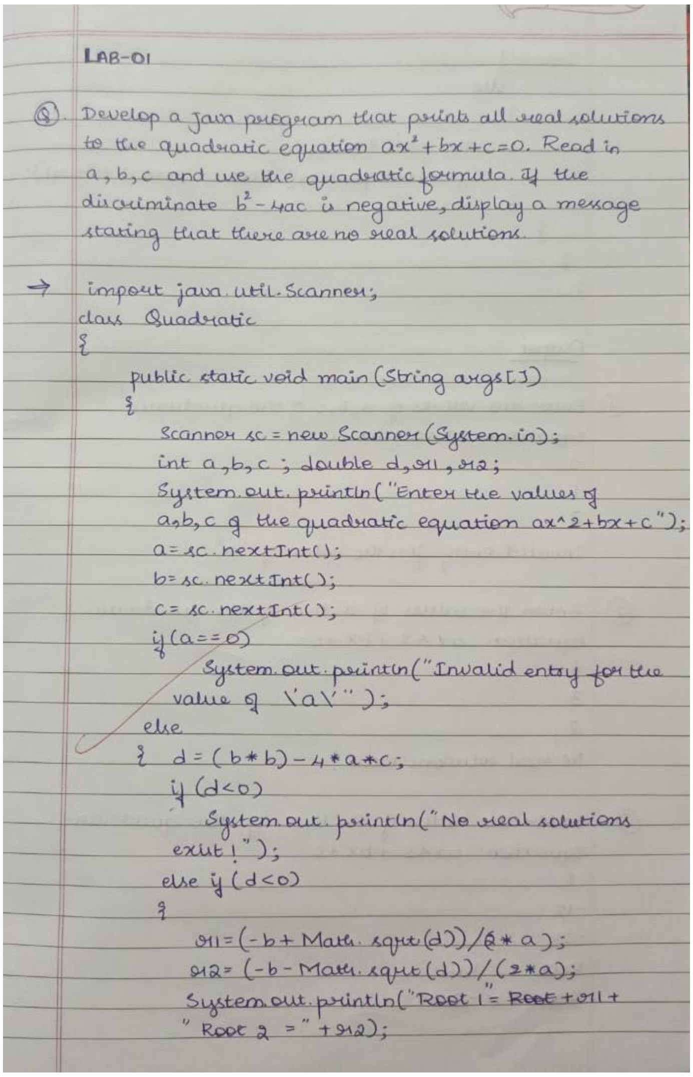
LAB PROGRAMS

1. Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.

```
import java.util.Scanner;
class Quadratic
{
      public static void main(String args[])
      {
            Scanner sc=new Scanner(System.in);
            int a,b,c;double d,r1,r2;
            System.out.println("Enter the values of a,b,c of the quadratic 
equation ax^2+bx+c");
            a=sc.nextInt();
            b=sc.nextInt();
            c=sc.nextInt();
            if(a==0)
                  System.out.println("Invalid entry for the value of 
\'a\'");
            else
            {
                  d=(b*b)-4*a*c;

                  if(d<0)
                  {
                        System.out.println("No real solutions exist!");
                  }
                  else if(d>0)
                  {
                        r1=(-b+Math.sqrt(d))/(2*a);r2=(-b-
Math.sqrt(d))/(2*a);
                        System.out.println("Root 1 = "+r1+"\nRoot 2 = "+r2);
                  }
                  else
                  {
                        r1=-b/(2*a);
                        System.out.println("Root 1 = Root 2 = "+r1);
                  }
            }
      }
}
```

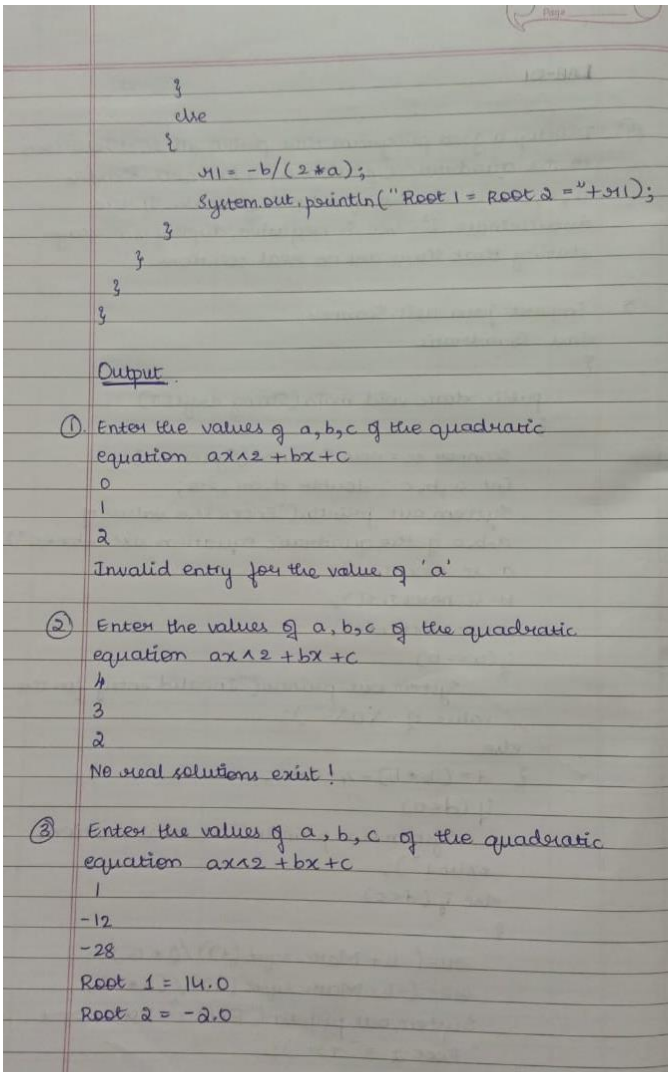


LAB-01

Q. 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 discriminate b^2-4ac is negative, display a message stating that there are no real solutions.

→

```
import java.util.Scanner;
class Quadratic
{
    public static void main(String args[])
    {
      Scanner sc = new Scanner(System.in);
      int a,b,c ; double d,r1,r2;
      System.out.println("Enter the values of
      a,b,c of the quadratic equation ax^2+bx+c");
      a=sc.nextInt();
      b=sc.nextInt();
      c=sc.nextInt();
      if(a==0)
          System.out.println("Invalid entry for the
       value of \'a\'");
     else
     { d=(b*b)-4*a*c;
       if(d<0)
          System.out.println("No real solutions
       exist!");
      else if(d<0)
      {
         r1=(-b+Math.sqrt(d))/(2*a);
         r2=(-b-Math.sqrt(d))/(2*a);
        System.out.println("Root 1 =" ~~Root~~ +r1 +
        " Root 2 = " +r2);
```



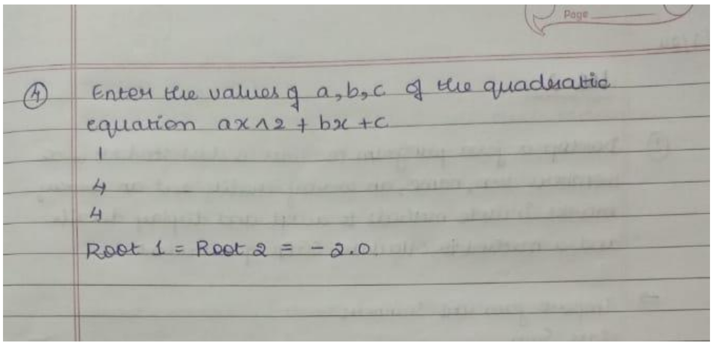
```
      }
      else
      {
         r1 = -b/(2*a);
         System.out.println("Root 1 = Root 2 ="+r1);
      }
    }
  }
}
```

Output

1.
```
Enter the values of a,b,c of the quadratic
equation ax^2 +bx+c
0
1
2
Invalid entry for the value of 'a'
```

2.
```
Enter the values of a,b,c of the quadratic
equation ax^2 +bx +c
4
3
2
No real solutions exist!
```

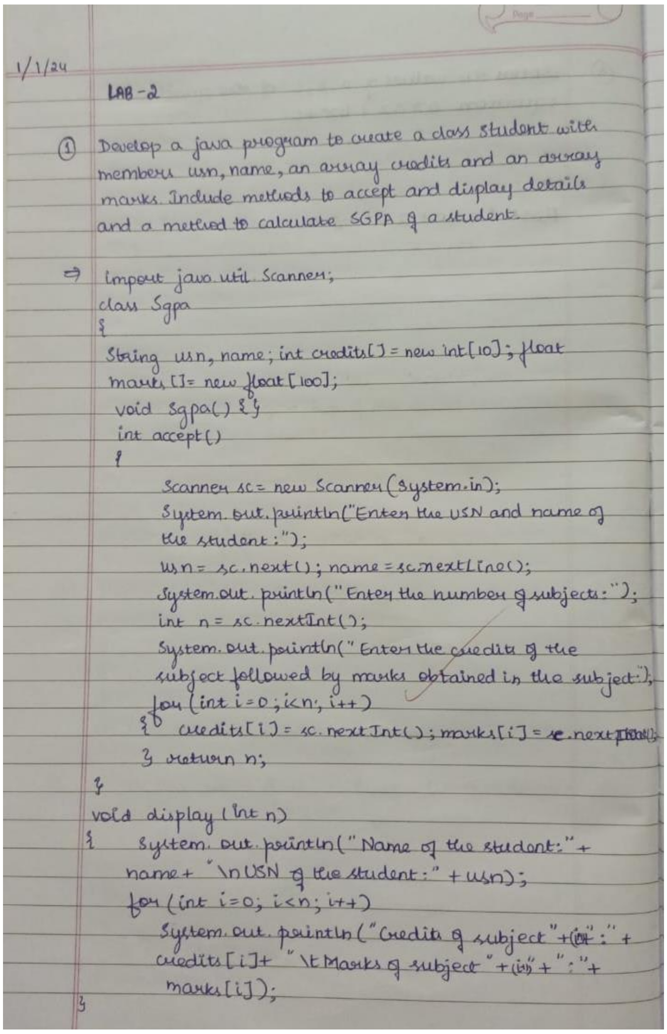
3.
```
Enter the values of a,b,c of the quadratic
equation ax^2 +bx+c
1
-12
-28
Root 1 = 14.0
Root 2 = -2.0
```



4.
```
Enter the values of a,b,c of the quadratic
equation ax^2 + bx +c
1
4
4
Root 1 = Root 2 = -2.0
```

2. Develop a Java program to create a class Student with members usn, name, an 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.Scanner;
class Sgpa
{
String usn,name;int credits[]=new int[100];float marks[]=new float[100];
void Sgpa(){}

int accept(){
Scanner sc=new Scanner(System.in);
System.out.println("Enter the name and usn of the student:");
name=sc.nextLine();
usn=sc.next();
System.out.println("Enter the number of the subjects:");
int n=sc.nextInt();
for(int i=0;i<n;i++){
System.out.println("Enter the credits of the subject followed by the marks
obtained in the subject:");
credits[i]=sc.nextInt();
marks[i]=sc.nextFloat();}
return n;
}

void display(int n){
System.out.println("Name of the student:"+name+"\nUSN of the
student:"+usn);
for(int i=0;i<n;i++)
System.out.println("Credits of subject"+(i+1)+":"+credits[i]+"\tMarks of
subject"+(i+1)+":"+marks[i]);
}

double sgpa_cal(int n){
double agg=0.0,s=0.0;int cr=0,crsum=0;
for(int i=0;i<n;i++){
if(marks[i]>=90)
cr=10;
else if(marks[i]>=80 && marks[i]<90)
cr=9;
else if(marks[i]>=70 && marks[i]<80)
cr=8;
else if(marks[i]>=60 && marks[i]<70)
cr=7;
else if(marks[i]>=50 && marks[i]<60)
cr=6;
else if(marks[i]>=40 && marks[i]<50)
cr=5;
else
cr=0;
agg+=credits[i]*cr;
crsum+=credits[i];
s=(agg/crsum);}
return s;
}
}
class Sgpa_main
{
public static void main(String args[]){
Sgpa a=new Sgpa();
int n=a.accept();
a.display(n);
System.out.println("SGPA :"+(a.sgpa_cal(n)));}
}
```



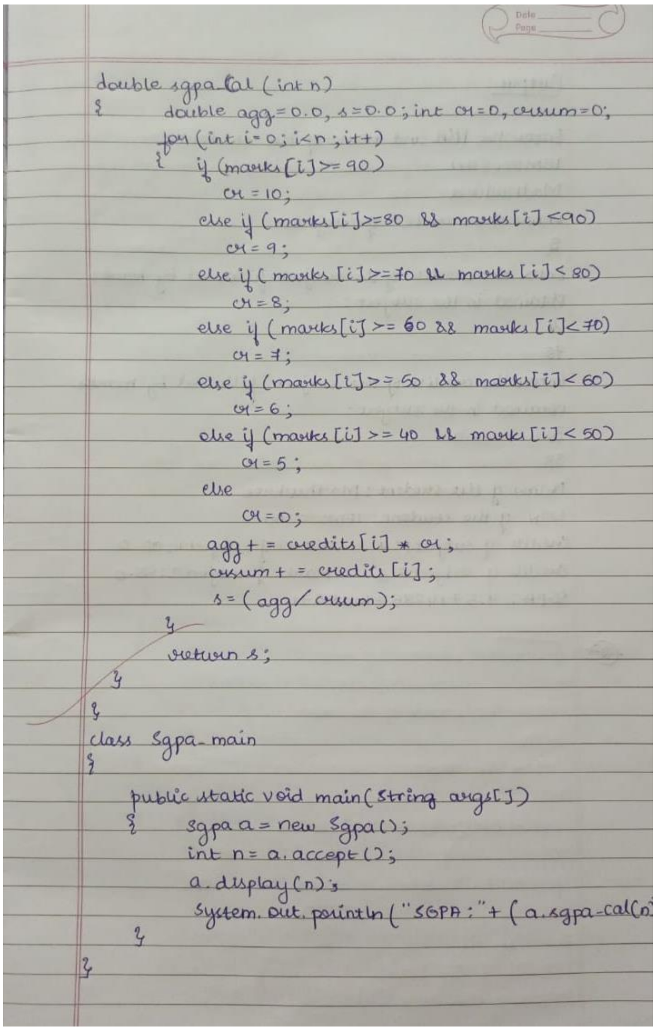
1/1/24

LAB -2

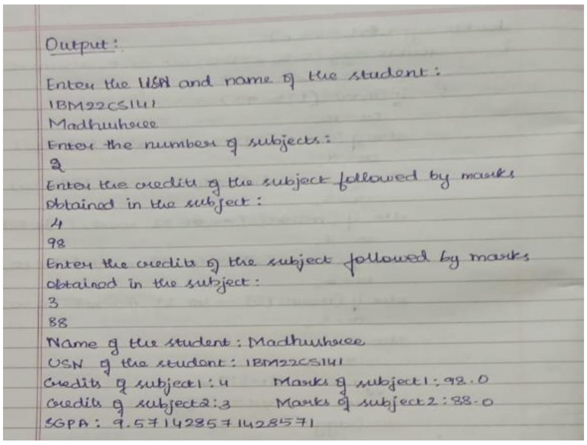
① Develop a java program to create a class student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

⇨

```
 import java.util.Scanner;
class Sgpa
{
 String usn, name; int credits[] = new int[10]; float
 marks[]= new float[100];
 void Sgpa() {}
 int accept()
 {
     Scanner sc= new Scanner(System.in);
     System.out.println("Enter the USN and name of
     the student:");
     usn= sc.next(); name=sc.nextLine();
     System.out.println("Enter the number of subjects:");
     int n= sc.nextInt();
     System.out.println("Enter the credits of the
     subject followed by marks obtained in the subject:");
    for (int i=0;i<n; i++)
   {   credits[i]= sc.nextInt(); marks[i]=sc.next[illegible]
    } return n;
 }
void display (int n)
{   System.out.println("Name of the student:"+
  name+ "\nUSN of the student:" +usn);
  for (int i=0; i<n; i++)
     System.out.println("Credits of subject"+(i+1)+":"+
     credits[i]+ "\t Marks of subject"+(i+1)+":"+
     marks[i]);
}
```



```
double sgpa_Cal (int n)
{      double agg=0.0, s=0.0; int cr=0, crsum=0;
       for (int i=0; i<n; i++)
       {   if (marks[i]>=90)
              cr =10;
           else if (marks[i]>=80 && marks[i]<90)
              cr=9;
           else if (marks[i]>=70 && marks[i]<80)
              cr=8;
           else if (marks[i]>= 60 && marks[i]<70)
              cr = 7;
           else if (marks[i]>=50 && marks[i]<60)
              cr=6;
           else if (marks[i] >= 40 && marks[i]<50)
              cr=5;
           else
              cr=0;
           agg += credits[i] * cr;
           crsum += credits[i];
           s=(agg/crsum);
       }
       return s;
   }
}
class Sgpa_main
{
    public static void main(String args[])
    {    Sgpa a=new Sgpa();
         int n= a.accept();
         a.display(n);
         System.out.println("SGPA:"+(a.sgpa_cal(n
    }
}
```

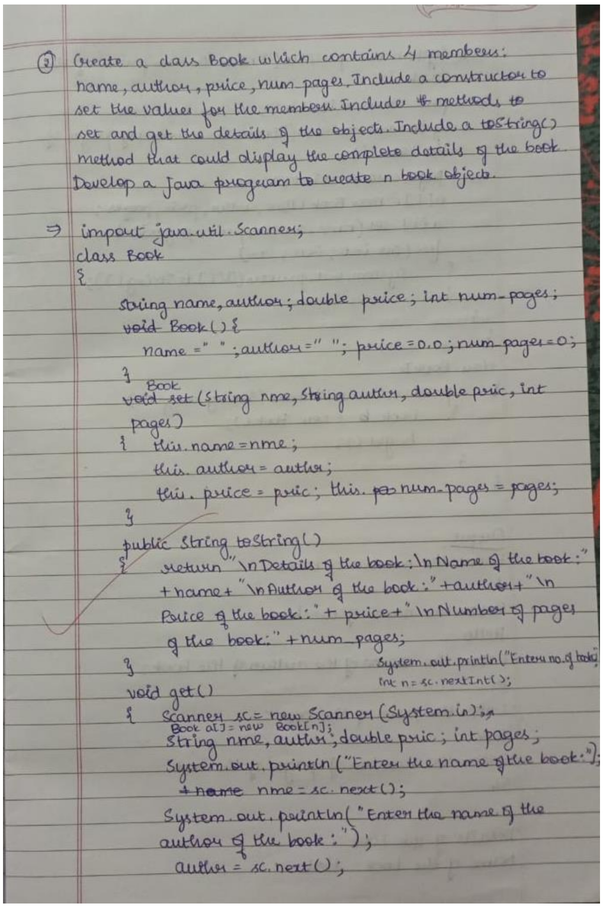


Output:

```
Enter the USN and name of the student:
1BM22CS141
Madhushree
Enter the number of subjects:
2
Enter the credits of the subject followed by marks
obtained in the subject:
4
98
Enter the credits of the subject followed by marks
obtained in the subject:
3
88
Name of the student: Madhushree
USN of the student: 1BM22CS141
Credits of subject1: 4      Marks of subject1: 98.0
Credits of subject2: 3      Marks of subject2: 88.0
SGPA: 9.571428571428571
```

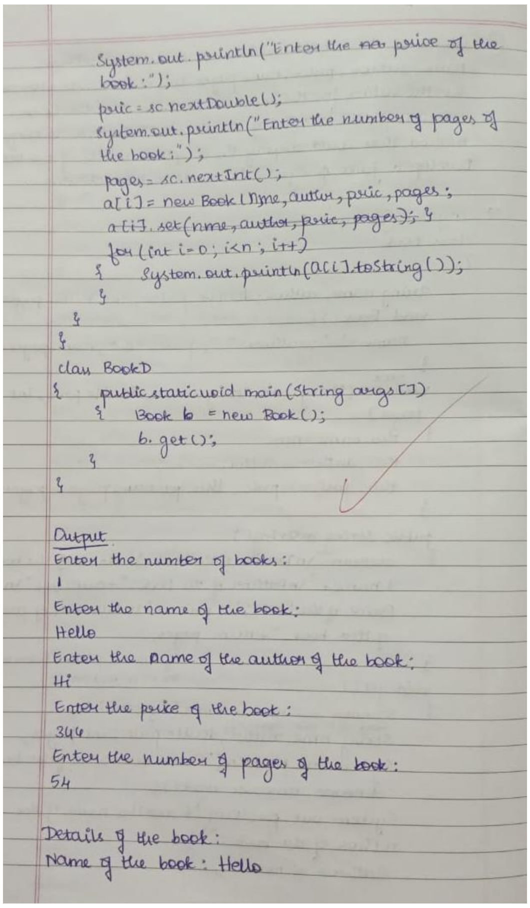
3. Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program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

```
import java.util.Scanner;
class Book
{
String name,author;double price;int num_pages;
Book(){
name="";author="";price=0.0;num_pages=0;}

Book(String nme,String authr,double pric,int pages){
this.name=nme;this.author=authr;this.price=pric;this.num_pages=pages;}

/*void set(String nme,String authr,double pric,int pages){
this.name=nme;this.author=authr;this.price=pric;this.num_pages=pages;}*/

/*void display(){
System.out.println("Name of the book:"+name);
System.out.println("Author of the book:"+author);
System.out.println("Price of the book:"+price);
System.out.println("Number of pages of the book:"+num_pages);
}*/

public String toString()
{
 return "\nDetails of the book:\nName of the book:"+name+"\nAuthor of the
book:"+author+"\nPrice of the book:"+price+"\nNumber of pages of the
book:"+num_pages;
}

void get(){
Scanner sc=new Scanner(System.in);
String nme;String authr;double pric;int pages;
System.out.println("Enter the number of books:");
int n=sc.nextInt();
Book a[]=new Book[n];

for(int i=0;i<n;i++){
System.out.println("Enter the name of the book:");
nme=sc.next();
System.out.println("Enter the name of the author of the book:");
authr=sc.next();
System.out.println("Enter the price of the book:");
pric=sc.nextDouble();
System.out.println("Enter the number of pages of the book:");
pages=sc.nextInt();
a[i]=new Book(nme,authr,pric,pages);
/*a[i].set(nme,authr,pric,pages);*/}
for(int i=0;i<n;i++){
System.out.println(a[i].toString());
}
}
}


class BookD
{
public static void main(String args[])
{
Book b=new Book();
b.get();
}
}
```



② Create a class Book which contains 4 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.

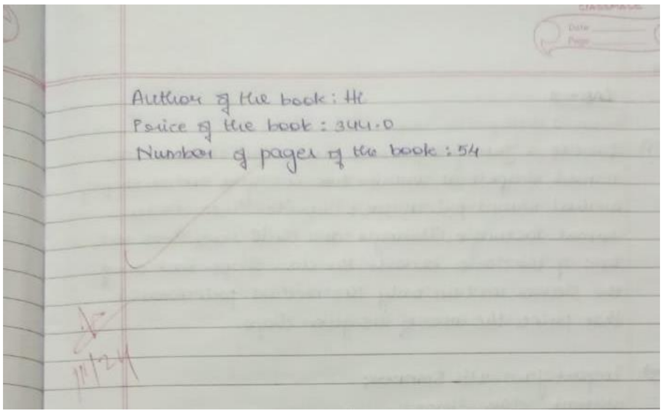
```
⇒ import java.util.Scanner;
class Book
{
    string name, author; double price; int num-pages;
    ~~void~~ Book(){
        name=" "; author=" "; price=0.0; num_pages=0;
    }
       Book
    ~~void set~~ (String nme, String auther, double pric, int
     pages)
    {   this.name=nme;
        this. author = auther;
        this. price = pric; this. ~~pa~~ num-pages = pages;
    }
    public String toString()
    {   return "\nDetails of the book:\nName of the book:"
        +name+ "\nAuthor of the book:"+author+"\n
        Price of the book:"+ price+"\nNumber of pages
        of the book:"+num_pages;
    }                          System.out.println("Enter no.of books");
                               int n=sc.nextInt();
    void get()
    {   Scanner sc= new Scanner(System.in);
         Book a[]=new Book[n];
        String nme, auther; double pric; int pages;
        System.out.println("Enter the name of the book:");
          ~~+name~~ nme=sc.next();
        System.out.println("Enter the name of the
        author of the book:");
         auther = sc.next();
```



```
        System.out.println("Enter the ~~ne~~ price of the
        book:");
        pric=sc.nextDouble();
        System.out.println("Enter the number of pages of
        the book:");
         pages= sc.nextInt();
         a[i]= new Book(nme, auther, pric, pages;
         ~~a[i].set(nme, author, pric, pages);~~ }
         for (int i=0; i<n; i++)
        {   System.out.println(a[i].toString());
        }
    }
}
class BookD
{   public static void main(String args[])
    {   Book b = new Book();
        b.get();
    }
}
```

Output

```
Enter the number of books:
1
Enter the name of the book:
Hello
Enter the name of the author of the book:
Hi
Enter the price of the book:
344
Enter the number of pages of the book:
54

Details of the book:
Name of the book: Hello
```

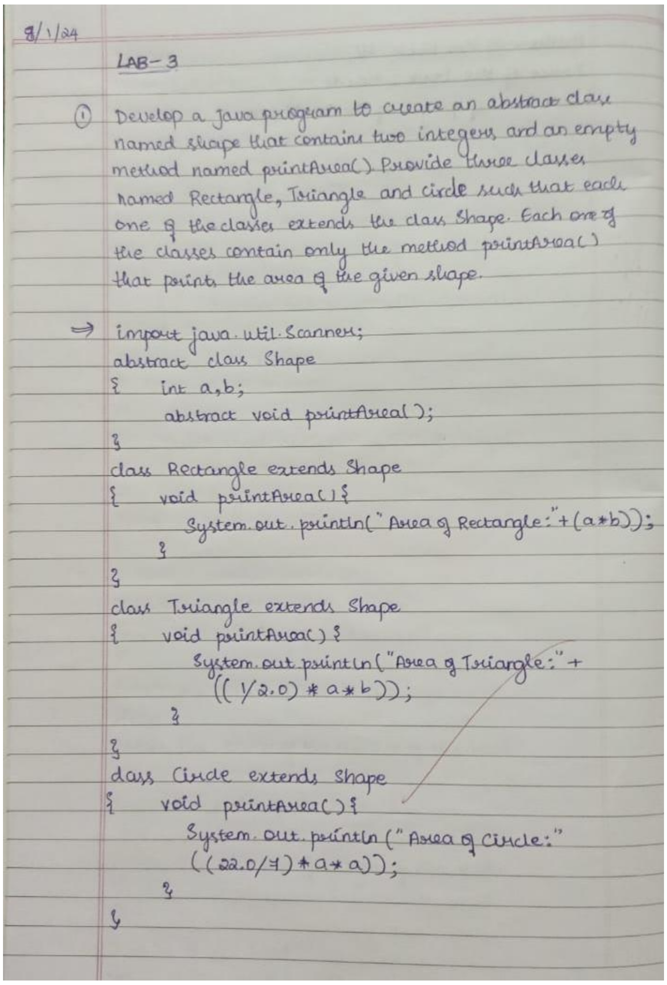


```
Author of the book: Hi
Price of the book : 344.0
Number of pages of the book : 54
```

11/24

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.

```
import java.util.Scanner;
abstract class Shape
{
int a,b;
abstract void printArea();
}
class Rectangle extends Shape
{ void printArea(){
 System.out.println("Area of Rectangle:" + (a*b));}
}
class Triangle extends Shape
{ void printArea(){
System.out.println("Area of Triangle:" + (.5*a*b));}
}
class Circle extends Shape
{ void printArea(){
System.out.println("Area of Circle:" +((22.0/7)*a*a));}
}

class Abs
{
public static void main(String args[])
{ Scanner sc= new Scanner(System.in);
Rectangle aa=new Rectangle();
Triangle bb=new Triangle();
Circle cc=new Circle();
System.out.println("Enter 1.Area of Rectangle\n2.Area of Triangle\n2.Area 
of Circle");
int n=sc.nextInt();
switch(n)
{case 1:
System.out.println("Enter length and breadth of rectangle:");
aa.a=sc.nextInt();
aa.b=sc.nextInt();
aa.printArea();
break;

case 2:
System.out.println("Enter length and height of triangle:");
bb.a=sc.nextInt();
bb.b=sc.nextInt();
bb.printArea();
break;

case 3:
System.out.println("Enter radius of circle:");
cc.a=sc.nextInt();
cc.printArea();
break;
default:System.out.println("Invalid choice");
}
}
}
```

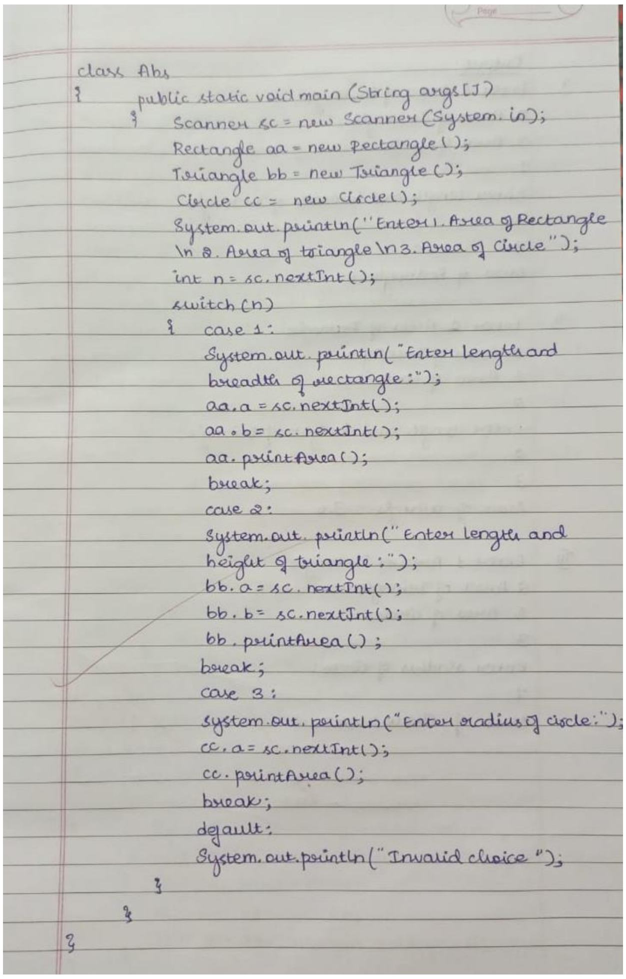


8/1/24

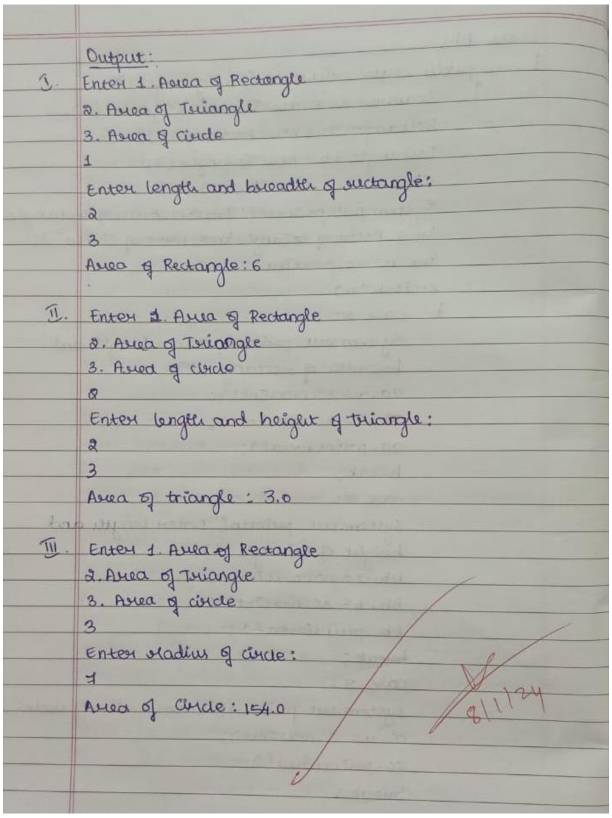
LAB-3

(1) 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.

```
⇒ import java.util.Scanner;
abstract class Shape
{    int a,b;
     abstract void printArea();
}
class Rectangle extends Shape
{    void printArea(){
       System.out.println("Area of Rectangle:"+(a*b));
     }
}
class Triangle extends Shape
{    void printArea(){
       System.out.println("Area of Triangle:"+
         ((1/2.0) * a*b));
      }
}
class Circle extends Shape
{    void printArea(){
       System.out.println("Area of Circle:"
       ((22.0/7)*a*a));
     }
}
```



```
class Abs
{      public static void main(String args[])
       {   Scanner sc = new Scanner(System.in);
           Rectangle aa = new Rectangle();
           Triangle bb = new Triangle();
           Circle cc = new Circle();
           System.out.println("Enter 1. Area of Rectangle
           \n 2. Area of triangle \n 3. Area of Circle");
           int n = sc.nextInt();
           switch(n)
           {  case 1:
              System.out.println("Enter length and
              breadth of rectangle:");
              aa.a = sc.nextInt();
              aa.b = sc.nextInt();
              aa.printArea();
              break;
              case 2:
              System.out.println("Enter length and
              height of triangle:");
              bb.a = sc.nextInt();
              bb.b = sc.nextInt();
              bb.printArea();
              break;
              case 3:
              system.out.println("Enter radius of circle:");
              cc.a = sc.nextInt();
              cc.printArea();
              break;
              default:
              System.out.println("Invalid choice");
           }
       }
}
```



Output:

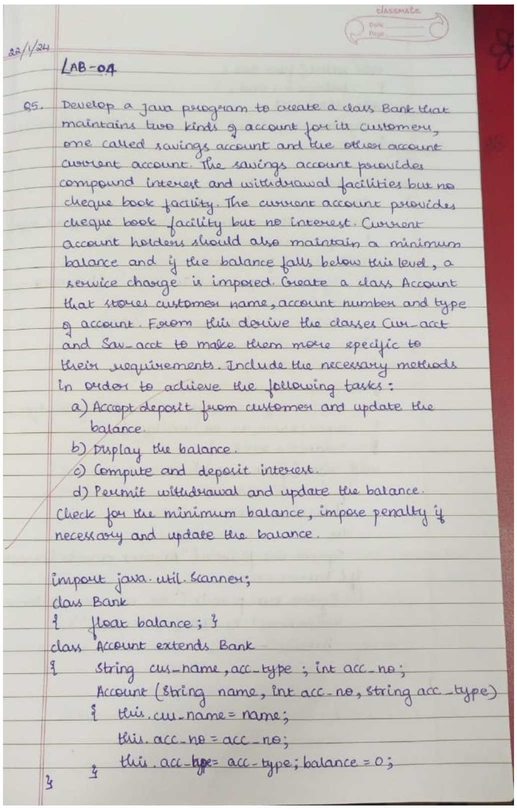
```
I.  Enter 1. Area of Rectangle
    2. Area of Triangle
    3. Area of Circle
    1
    Enter length and breadth of rectangle:
    2
    3
    Area of Rectangle: 6

II. Enter 1. Area of Rectangle
    2. Area of Triangle
    3. Area of circle
    2
    Enter length and height of triangle:
    2
    3
    Area of triangle : 3.0

III. Enter 1. Area of Rectangle
    2. Area of Triangle
    3. Area of circle
    3
    Enter radius of circle:
    7
    Area of Circle: 154.0
```

8/1/24

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 Bank
{
    float balance;
}

class Account extends Bank
{
    String cus_name,acc_type;int acc_no;
    Account(String name,int acc_no,String acc_type)
    {

this.cus_name=name;this.acc_no=acc_no;this.acc_type=acc_type;balance=0.0f;
    }
    void setBal(float amt)
    {
        balance+=amt;
    }
    void disBal(){
    System.out.println("Balance:Rs "+this.balance);
    }
}

class Sav_acct extends Account
{
    float comp_int,withdrawal;
    Sav_acct(String name,int acc_no,String acc_type)
    {
        super(name,acc_no,acc_type);
    }
    void inter(float rate)
    {
        System.out.println("Interest:"+(balance*rate/100));
        balance+=(balance*rate/100);
    }
}

class Cur_acct extends Account
{
    float comp_int,withdrawal;
    Cur_acct(String name,int acc_no,String acc_type)
    {
        super(name,acc_no,acc_type);
```

```
      balance=0.0f;
    }
    void with(float amt){
        if(amt<balance)
        balance-=amt;
        else
        System.out.println("Amount exceeds balance!");
    if(balance<500)
        {
            System.out.println("No minimum balance maintained!Rs.500 fine
levied.");
            balance-=500;
        }
    }
}

class LabQ5
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        int k=0,j=0;int ch;boolean t=true,t1=true;
        System.out.println("Enter the number of customers:");
        int n=sc.nextInt();
        Sav_acct a[]=new Sav_acct[n];Cur_acct b[]=new Cur_acct[n];

        for(int i=0;i<n;i++)
        {
            System.out.println("Enter the name, Account type and Account
number of user" +(i+1)+":");
            String name=sc.next();
            String acc_type=sc.next();
            int acc_no=sc.nextInt();

            if(acc_type.equalsIgnoreCase("savings"))
            {
                a[k]=new Sav_acct(name,acc_no,acc_type);
                while(t)
                {
                    System.out.println("Enter 1.Update balance.\n2.Display
balance.\n3.Compute and deposit interest.\n4.End");
                    ch=sc.nextInt();
                    switch(ch){
                    case 1:System.out. println("Enter the deposit
amount:");
                    a[k].setBal(sc.nextFloat());
                    break;
                    case 2:
                    a[k].disBal();
                    break;
                    case 3:System.out.println("Enter the rate of
interest:");
                    a[k].inter(sc.nextFloat());
                    break;
                    case 4:t=false;break;
                    default:System.out.println("Invalid choice");
                    }
                }
                k++;
            }

            if(acc_type.equalsIgnoreCase("current"))
            {
                b[j]=new Cur_acct(name,acc_no,acc_type);
                while(t1)
                {
                    System.out.println("Enter 1.Update balance.\n2.Display
balance.\n3.withdraw and update balance.\n 4.End");
                    ch=sc.nextInt();
                    switch(ch){
                    case 1:System.out. println("Enter the deposit
amount:");
                    b[j].setBal(sc.nextFloat());
                    break;
                    case 2:
                    b[j].disBal();
                    break;
                    case 3:System.out. println("Enter the withdraw
amount:");
                    b[j].with(sc.nextFloat());
                    break;
                    case 4:t1=false;break;
                    default:System.out.println("Invalid choice");
                    }
                }j++;
            }
        }
    }
}
```



22/1/24

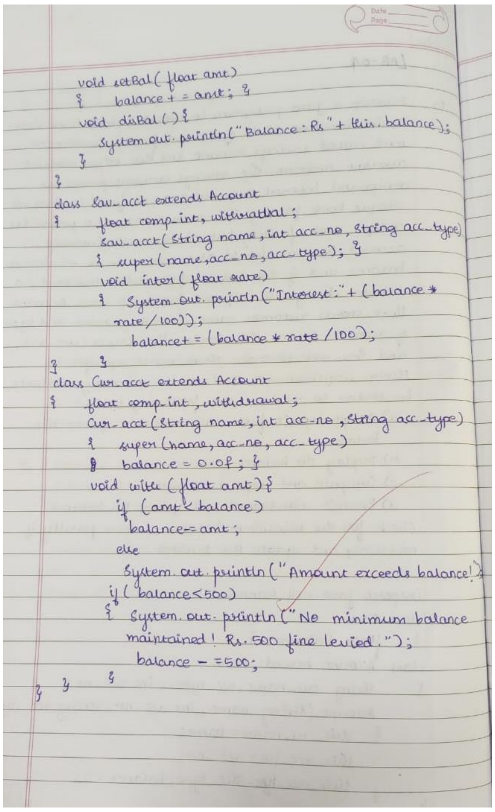
LAB-04

Q5. 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 account 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:

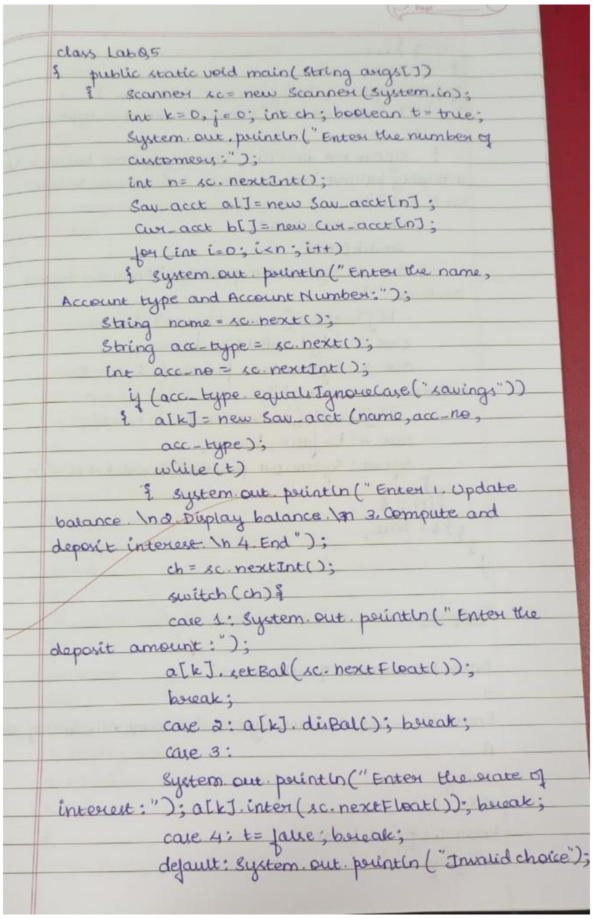
a) Accept deposit from customer and update the balance.
b) Display the balance.
c) Compute and deposit interest.
d) Permit withdrawal and update the balance.

Check for the minimum balance, impose penalty if necessary and update the balance.

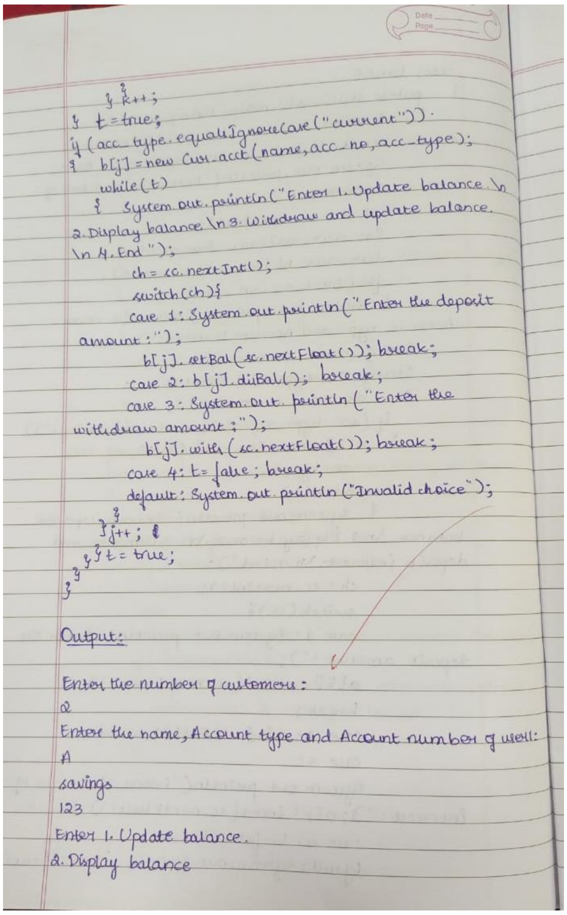
```
import java.util.Scanner;
class Bank
{   float balance; }
class Account extends Bank
{   String cus_name, acc_type ; int acc_no;
    Account (String name, int acc_no, String acc_type)
    {  this.cu_name = name;
       this.acc_no = acc_no;
    }  this.acc_~~type~~= acc_type; balance = 0;
}
```



```
    void setBal( float amt)
    {    balance + = amt; }
    void disBal (){
      System.out.println("Balance : Rs" + this.balance);
    }
}
class Sav_acct extends Account
{    float comp_int, withratbal;
     Sav_acct(String name, int acc_no, String acc_type)
     { super(name,acc_no,acc_type); }
     void inter( float rate)
     {  System.out.println("Interest :"+ (balance *
        rate / 100));
          balance+ = (balance * rate /100);
}    }
class Cur_acct extends Account
{   float comp_int , withdrawal;
    Cur_acct (String name, int acc_no , String acc_type)
    {   super (name, acc_no, acc_type)
    ~~{~~   balance = 0.0f; }
    void with (float amt){
        if (amt < balance)
         balance-= amt;
        else
         System.out.println("Amount exceeds balance!");
       if ( balance<500)
       {  System.out.println("No minimum balance
          maintained! Rs. 500 fine levied.");
           balance - =500;
}   }      }
```



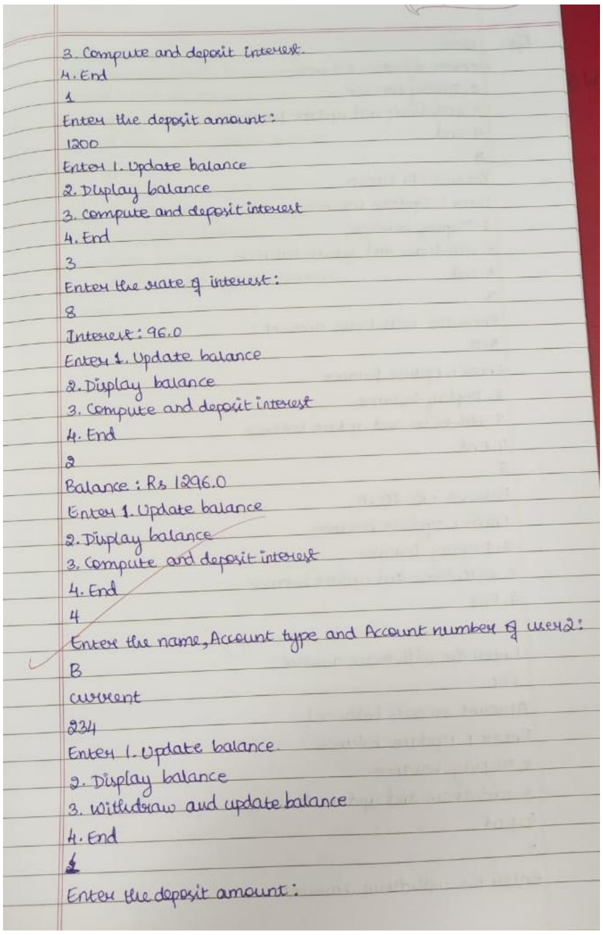
```
class LabQ5
{   public static void main( String args[])
    {     Scanner sc= new Scanner(System.in);
          int k=0, j=0; int ch; boolean t= true;
          System.out.println("Enter the number of
          customers:");
          int n= sc.nextInt();
          Sav_acct a[]= new Sav_acct[n] ;
          Cur_acct b[]= new Cur_acct[n];
          for (int i=0; i<n; i++)
         { system.out.println("Enter the name,
Account type and Account Number:");
      String  name = sc.next();
      String  acc_type = sc.next();
      int  acc_no = sc.nextInt();
         if (acc_type.equalsIgnoreCase("savings"))
        {  a[k]= new Sav_acct (name,acc_no,
             acc_type);
             while(t)
           {  system.out.println("Enter 1. Update
balance. \n2. Display balance. \n 3. Compute and
deposit interest. \n 4. End");
              ch = sc.nextInt();
              switch (ch){
              case 1: System.out.println("Enter the
deposit amount :");
              a[k].setBal(sc.nextFloat());
              break;
              case 2: a[k].disBal(); break;
              case 3:
              System.out.println("Enter the rate of
interest :"); a[k].inter(sc.nextFloat()); break;
              case 4: t= false; break;
              default: System.out.println("Invalid choice");
```



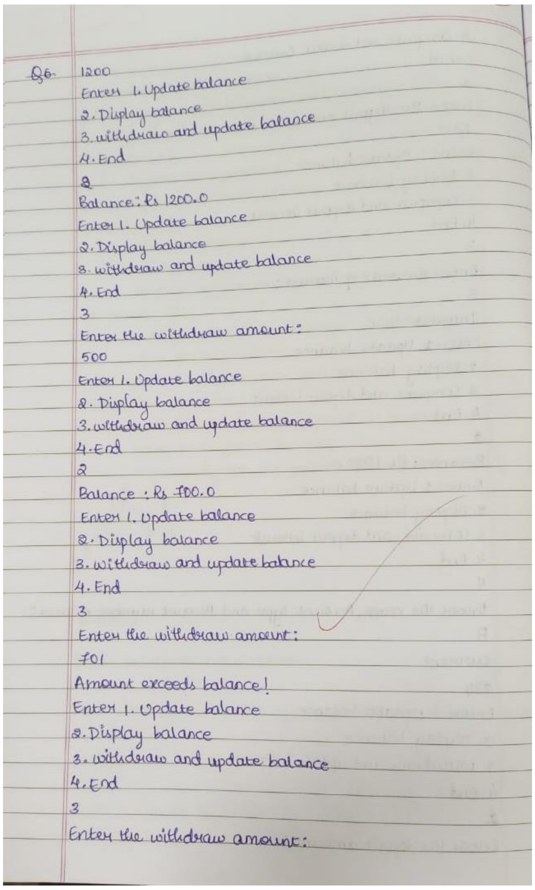
```
      } }
        k++;
    } t=true;
    if (acc_type.equalsIgnoreCase("current")).
    {  b[j]=new Cur_acct(name,acc_no,acc_type);
       while(t)
      {  System.out.println("Enter 1. Update balance.\n
2. Display balance \n 3. Withdraw and update balance.
\n 4. End");
           ch = sc.nextInt();
           switch(ch){
           case 1: System.out.println("Enter the deposit
amount :");
             b[j].setBal(sc.nextFloat()); break;
           case 2: b[j].disBal(); break;
           case 3: System.out.println("Enter the
withdraw amount :");
             b[j].with(sc.nextFloat()); break;
           case 4: t= false; break;
           default: System.out.println("Invalid choice");
        }
      } j++;
    } } t= true;
  }
}
```

Output:

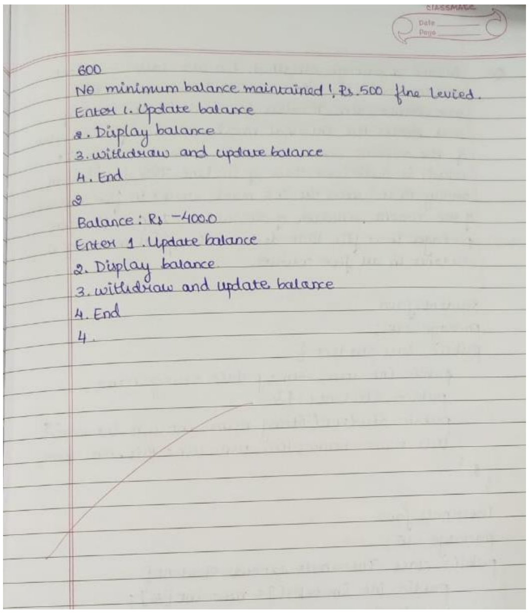
```
Enter the number of customers:
2
Enter the name, Account type and Account number of user1:
A
savings
123
Enter 1. Update balance.
2. Display balance
```



```
3. Compute and deposit interest.
4. End
1
Enter the deposit amount:
1200
Enter 1. Update balance
2. Display balance
3. compute and deposit interest
4. End
3
Enter the rate of interest:
8
Interest: 96.0
Enter 1. Update balance
2. Display balance
3. Compute and deposit interest
4. End
2
Balance : Rs 1296.0
Enter 1. Update balance
2. Display balance
3. Compute and deposit interest
4. End
4
Enter the name, Account type and Account number of user2:
B
current
234
Enter 1. Update balance.
2. Display balance
3. Withdraw and update balance
4. End
1
Enter the deposit amount:
```



```
Q6. 1200
    Enter 1. Update balance
    2. Display balance
    3. withdraw and update balance
    4. End
    2
    Balance: Rs 1200.0
    Enter 1. Update balance
    2. Display balance
    3. withdraw and update balance
    4. End
    3
    Enter the withdraw amount:
    500
    Enter 1. Update balance
    2. Display balance
    3. withdraw and update balance
    4. End
    2
    Balance : Rs 700.0
    Enter 1. Update balance
    2. Display balance
    3. withdraw and update balance
    4. End
    3
    Enter the withdraw amount:
    701
    Amount exceeds balance!
    Enter 1. Update balance
    2. Display balance
    3. withdraw and update balance
    4. End
    3
    Enter the withdraw amount:
```



```
600
NO minimum balance maintained !, Rs.500 fine levied.
Enter 1. Update balance
2. Display balance
3. withdraw and update balance
4. End
2
Balance : Rs -400.0
Enter 1. Update balance
2. Display balance
3. withdraw and update balance
4. End
4.
```

6. Create a package CIE which has two classes- Student and Internals. The class Personal 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.

Student.java

```
package CIE;
public class Student{
public int usn,sem;public String name;
Student(){}
Student(String name,int usn,int sem)
{this.name=name;this.usn=usn;this.sem=sem;
}
}
```

Internals.java

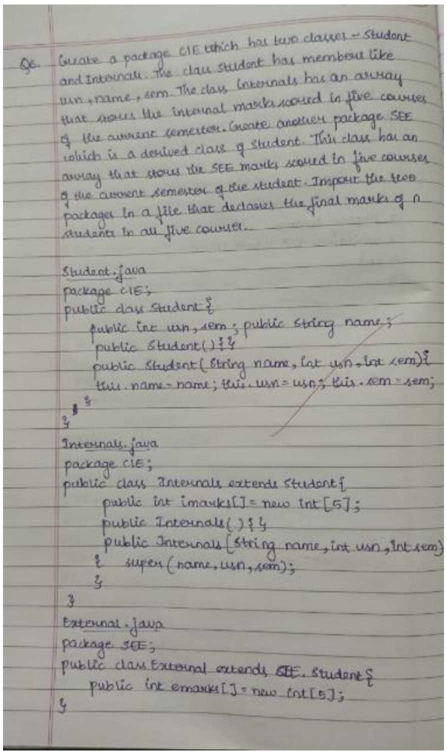
```
package CIE;
public class Internals extends Student{
public int imarks[]=new int[5];
Internals(){}
Internals(String name,int usn,int sem)
{super(name,usn,sem);
}
}
```

External.java

```
package SEE;
/*import CIE.Student;*/
public class External extends CIE.Student{
public int emarks[]=new int[5];
}
```

LabQ6.java

```
import java.util.Scanner;
import CIE.Student;import CIE.Internals;import SEE.External;

class LabQ6{
public static void main(String args[]){
Scanner sc=new Scanner(System.in);
String name;int usn,sem;
System.out.println("Enter the number of students:");
int n=sc.nextInt();
Internals in[]=new Internals[n];
External ex[]=new External[n];
for(int i=0;i<n;i++)
{
System.out.println("Enter the name of the student "+(i+1)+":");
name=sc.next();
System.out.println("Enter the usn of the student "+(i+1)+":");
usn=sc.nextInt();
System.out.println("Enter the sem of the student "+(i+1)+":");
sem=sc.nextInt();
in[i]=new Internals(name,usn,sem);
ex[i]=new External();
System.out.println("Enter the internal marks of the student in 5 subjects:");
for(int j=0;j<5;j++)
in[i].imarks[j]=sc.nextInt();
System.out.println("Enter the external marks of the student in 5 subjects:");
for(int j=0;j<5;j++)
ex[i].emarks[j]=sc.nextInt();
}

for(int i=0;i<n;i++)
{
System.out.println("\nDetails of student "+(i+1)+":");
System.out.print("Name:"+in[i].name+"\t");
System.out.print("USN:"+in[i].usn+"\t");
System.out.print("Sem:"+in[i].sem+"\t");
System.out.println("\nInternal marks:");
for(int j=0;j<5;j++)
System.out.print("Subject "+(j+1)+":"+in[i].imarks[j]+"\t");
System.out.println("\nExternal marks:");
for(int j=0;j<5;j++)
System.out.print("Subject "+(j+1)+":"+ex[i].emarks[j]+"\t");
System.out.println("Final marks:");
for(int j=0;j<5;j++)
System.out.println("Subject "+(j+1)+":"+(in[i].imarks[j]+ex[i].emarks[j]));
}
}
}
```



Q6. 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. Create another package SEE 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.

Student.java

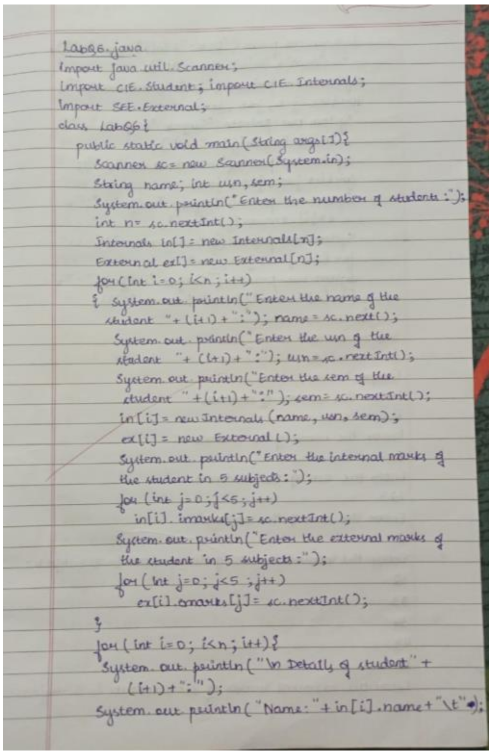
```
package CIE;
public class Student {
    public int usn, sem; public String name;
    public Student() { }
    public Student( String name, int usn, int sem){
    this.name = name; this.usn = usn; this.sem = sem;
  }
}
```

Internals.java

```
package CIE;
public class Internals extends Student{
      public int imarks[] = new int[5];
      public Internals( ) { }
      public Internals( String name, int usn, int sem)
    {   super( name, usn, sem);
    }
 }
```

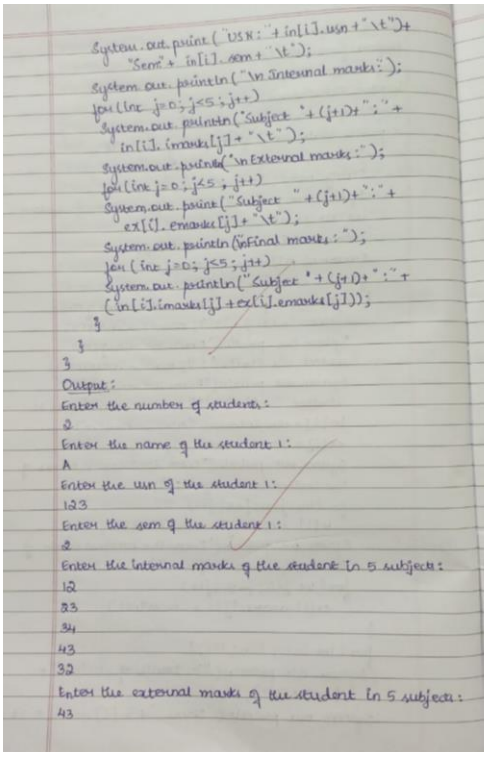
External.java

```
package SEE;
public class External extends SEE. Student{
    public int emarks[] = new int[5];
}
```



LabQ6.java

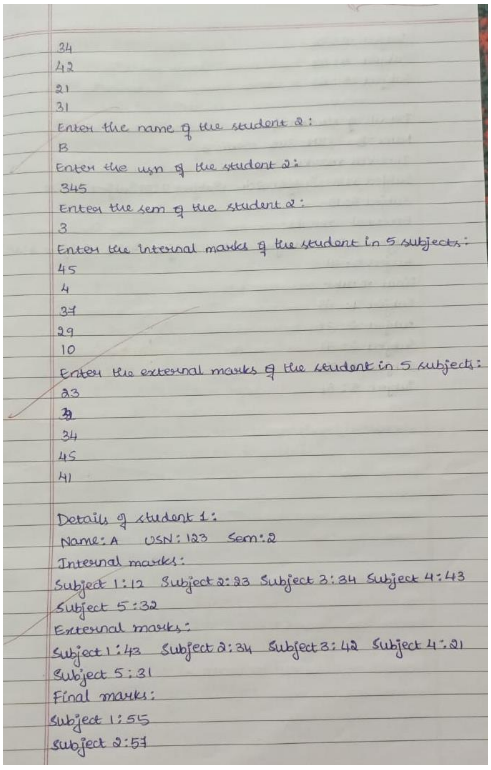
```
Import java.util.Scanner;
Import CIE.Student; import CIE.Internals;
Import SEE.External;
class LabQ6{
  public static void main( String args[]){
     Scanner sc = new Scanner(System.in);
     String name; int usn, sem;
     System.out.println("Enter the number of students :");
     int n = sc.nextInt();
     Internals in[] = new Internals[n];
     External ex[] = new External[n];
     for(int i=0; i<n; i++)
     { System.out.println("Enter the name of the
       student "+ (i+1) + ":"); name = sc.next();
       System.out.println("Enter the usn of the
       student "+ (i+1) + ":"); usn = sc.nextInt();
       System.out.println("Enter the sem of the
        student "+(i+1)+":"); sem = sc.nextInt();
       in[i] = new Internals (name, usn, sem);
       ex[i] = new External ();
       System.out.println("Enter the internal marks of
       the student in 5 subjects :");
       for (int j=0; j<5; j++)
         in[i]. imarks[j] = sc.nextInt();
       System.out.println("Enter the external marks of
       the student in 5 subjects :");
       for ( int j=0; j<5 ; j++)
          ex[i].emarks[j] = sc.nextInt();
     }
     for ( int i=0; i<n; i++){
      System.out.println( "\n Details of student" +
        (i+1) + ":");
     System.out.println( "Name: " + in[i].name + "\t");
```



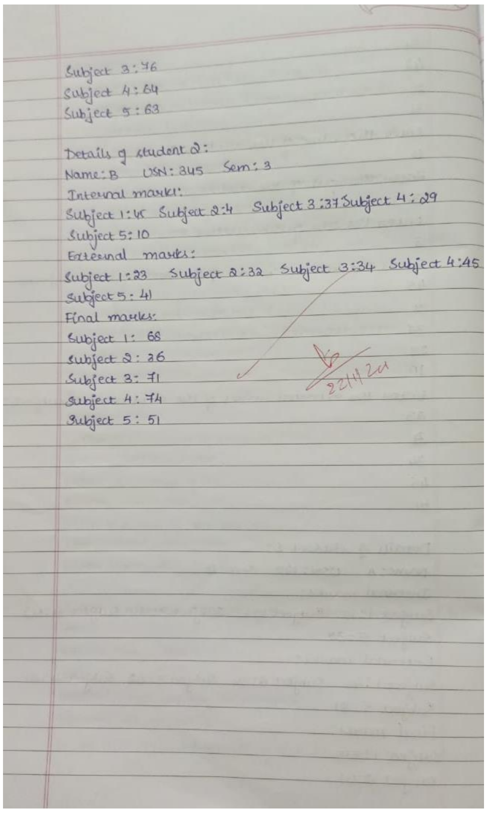
```
System.out.print("USN:"+in[i].usn+"\t")+
      "Sem:"+ in[i].sem+"\t");
System.out.println("\n Internal marks:");
for(int j=0;j<5;j++)
 System.out.println("Subject "+(j+1)+":"+
    in[i].imarks[j]+"\t");
 System.out.println("\n External marks:");
 for(int j=0;j<5;j++)
 System.out.print("Subject "+(j+1)+":"+
    ex[i].emarks[j]+"\t");
 System.out.println("\nFinal marks:");
 for(int j=0;j<5;j++)
 System.out.println("Subject "+(j+1)+":"+
 (in[i].imarks[j]+ex[i].emarks[j]));
 }
  }
}
```

Output:

```
Enter the number of students:
2
Enter the name of the student 1:
A
Enter the usn of the student 1:
123
Enter the sem of the student 1:
2
Enter the internal marks of the student in 5 subjects:
12
23
34
43
32
Enter the external marks of the student in 5 subjects:
43
```



```
34
42
21
31
Enter the name of the student 2:
B
Enter the usn of the student 2:
345
Enter the sem of the student 2:
3
Enter the internal marks of the student in 5 subjects:
45
4
37
29
10
Enter the external marks of the student in 5 subjects:
23
32
34
45
41

Details of student 1:
Name: A   USN: 123   Sem: 2
Internal marks:
Subject 1: 12  Subject 2: 23  Subject 3: 34  Subject 4: 43
Subject 5: 32
External marks:
Subject 1: 43  Subject 2: 34  Subject 3: 42  Subject 4: 21
Subject 5: 31
Final marks:
Subject 1: 55
Subject 2: 57
```

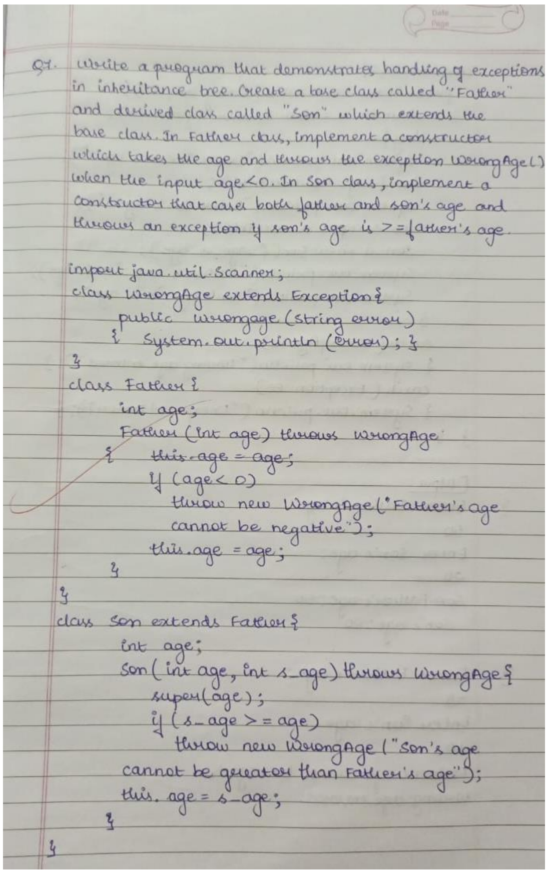


```
Subject 3: 76
Subject 4: 64
Subject 5: 63

Details of student 2:
Name: B   USN: 345   Sem: 3
Internal marks:
Subject 1: 45  Subject 2: 4  Subject 3: 37 Subject 4: 29
Subject 5: 10
External marks:
Subject 1: 23  Subject 2: 32  Subject 3: 34  Subject 4: 45
Subject 5: 41
Final marks:
Subject 1: 68
Subject 2: 36
Subject 3: 71
Subject 4: 74
Subject 5: 51
```

22/4/24

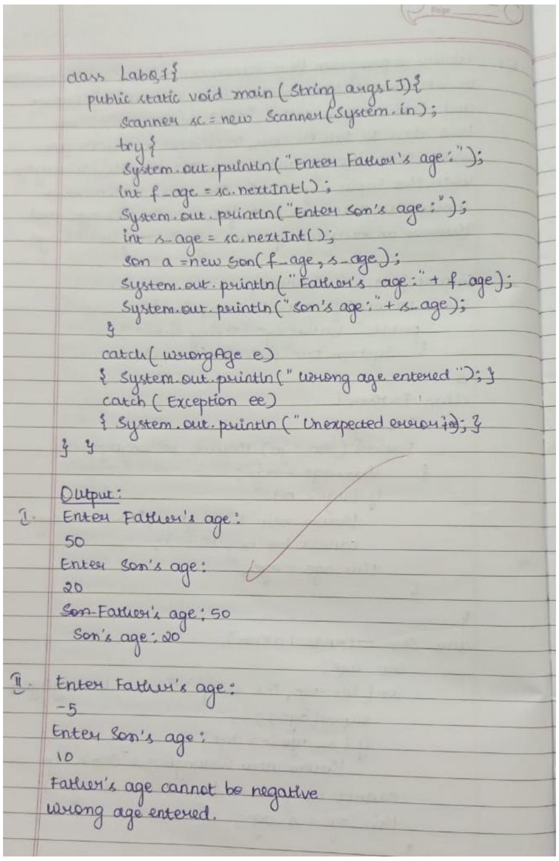
7. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age=father’s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

```
import java.util.Scanner;
class WrongAge extends Exception{
WrongAge(String error)
{     System.out.println(error);
}
}

class Father
{
int age;
Father(int age) throws WrongAge{
if(age<0)
throw new WrongAge("Father's age cannot be negative");
this.age=age;
}
}


class Son extends Father{
int age;
Son(int age,int s_age) throws WrongAge{
super(age);
if(s_age>=age)
throw new WrongAge("Son's age connot be greater than Father's age");
this.age=s_age;
}
}


class LabQ7{
public static void main(String args[]){
Scanner sc=new Scanner(System.in);
try{
System.out.println("Enter Father's age:");
int f_age=sc.nextInt();
System.out.println("Enter Son's age:");
int s_age=sc.nextInt();
Son a=new Son(f_age,s_age);

System.out.println("Father's age:"+f_age);
System.out.println("Son's age:"+s_age);
}
catch(WrongAge e)
{System.out.println("Wrong age entered");}
catch(Exception ee)
{System.out.println("Unexpected error"+ee);}
}}
```



Q7. 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<0. In Son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >=father's age.

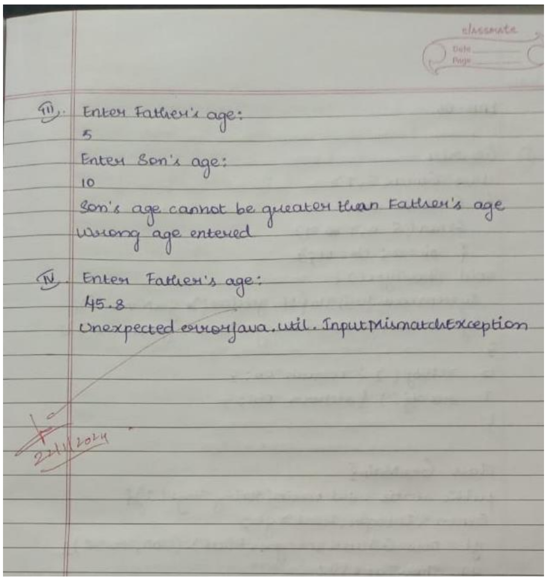
```
import java.util.Scanner;
class WrongAge extends Exception{
      public  Wrongage (string error)
    {  System.out.println (error); }
}
class Father {
      int age;
      Father (int age) throws WrongAge
    {     ~~this.age = age;~~
         if (age< 0)
           throw new WrongAge("Father's age
           cannot be negative");
         this.age = age;
    }
}
class Son extends Father{
      int age;
      Son( int age, int s_age) throws WrongAge{
         super(age);
         if (s_age >= age)
            throw new WrongAge ("Son's age
      cannot be greater than Father's age");
      this. age = s_age;
    }
}
```



```
class LabQ7{
  public static void main (String args[]){
     Scanner sc = new Scanner(System.in);
     try{
     System.out.println("Enter Father's age:");
     int f_age = sc.nextInt();
     System.out.println("Enter Son's age:");
     int s_age = sc.nextInt();
     Son a =new Son(f_age, s_age);
     System.out.println("Father's age:"+ f_age);
     System.out.println("Son's age:"+ s_age);
    }
    catch( WrongAge e)
    { System.out.println("Wrong age entered"); }
    catch ( Exception ee)
    { System.out.println ("Unexpected error"+ee); }
} }
```

Output:

```
I.  Enter Father's age:
    50
    Enter Son's age:
    20
    Son-Father's age: 50
     Son's age: 20

II. Enter Father's age:
    -5
    Enter Son's age:
    10
    Father's age cannot be negative
    Wrong age entered.
```

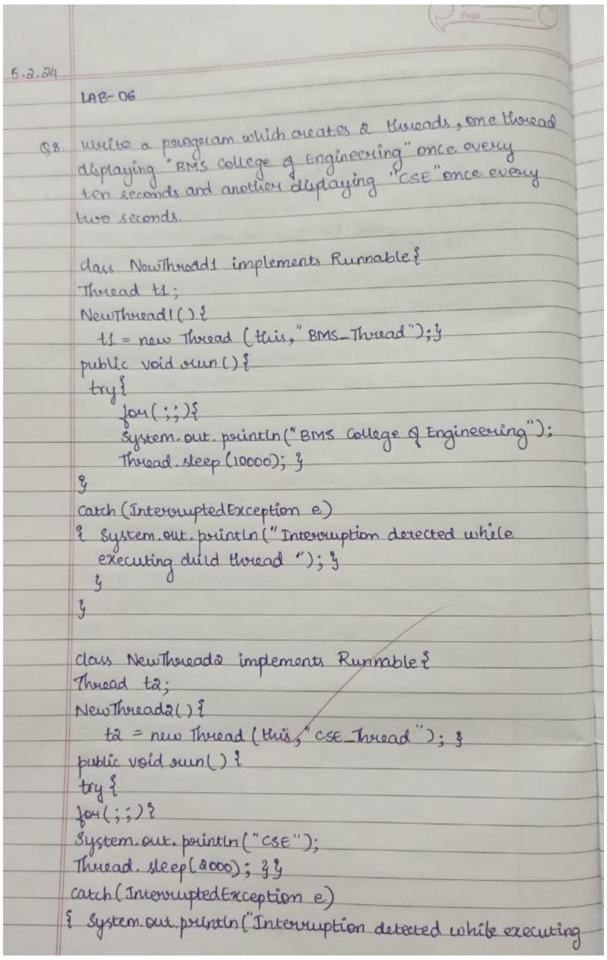


```
III. Enter Father's age:
     5
     Enter Son's age:
     10
     Son's age cannot be greater than Father's age
     Wrong age entered

IV.  Enter Father's age:
     45.8
     Unexpected errorJava.util.InputMismatchException
```

22/1/2024

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 NewThread1 implements Runnable{
Thread t1;
NewThread1(){
t1=new Thread(this,"BMS_Thread");
}
public void run(){
try{
for(;;){
System.out.println("BMS College of Engineering");
Thread.sleep(10000);}
}
catch(InterruptedException e)
{System.out.println("Interruption detected while executing child thread
:");
}
}
}

class NewThread2 implements Runnable{
Thread t2;
NewThread2(){
t2=new Thread(this,"CSE_Thread");
}
public void run(){
try{
for(;;){
System.out.println("CSE");
Thread.sleep(2000);}
}
catch(InterruptedException e)
{System.out.println("Interruption detected while executing child thread
:");
}
}
}


class ThreadDemo{
public static void main(String args[]){
NewThread1 a=new NewThread1();
NewThread2 b=new NewThread2();
a.t1.start();
b.t2.start();
}
}
```

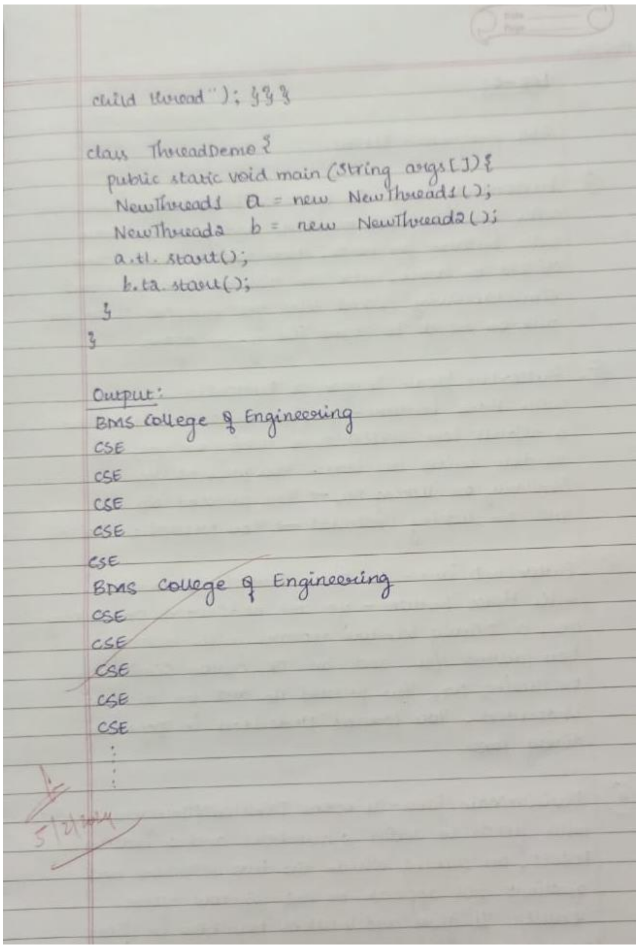


5.2.24

LAB-06

Q8. write a program which creates 2 threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

```
class NewThread1 implements Runnable{
Thread t1;
NewThread1(){
  t1 = new Thread (this,"BMS_Thread");}
public void run(){
 try{
     for(;;){
     System.out.println("BMS College of Engineering");
     Thread.sleep(10000); }
}
catch (InterruptedException e)
{ System.out.println("Interruption detected while
  executing child thread ");}
  }
}

class NewThread2 implements Runnable{
Thread t2;
NewThread2(){
   t2 = new Thread (this,"CSE_Thread"); }
public void run(){
try{
for(;;){
System.out.println("CSE");
Thread.sleep(2000); }}
catch(InterruptedException e)
{ System.out.println("Interruption detected while executing
```



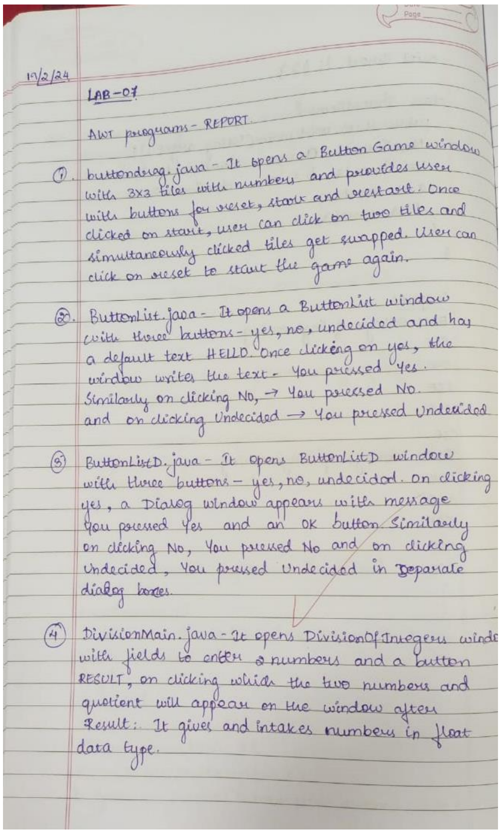
```
child thread"); }}}

class ThreadDemo{
 public static void main (String args[]){
  NewThreads1 a = new NewThreads1();
  NewThreads2 b = new NewThread2();
  a.t1.start();
   b.t2.start();
 }
}
```

Output:

```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
 :
 :
```

5/2/2024

AWT Programs Observations:



19/2/24

LAB-07

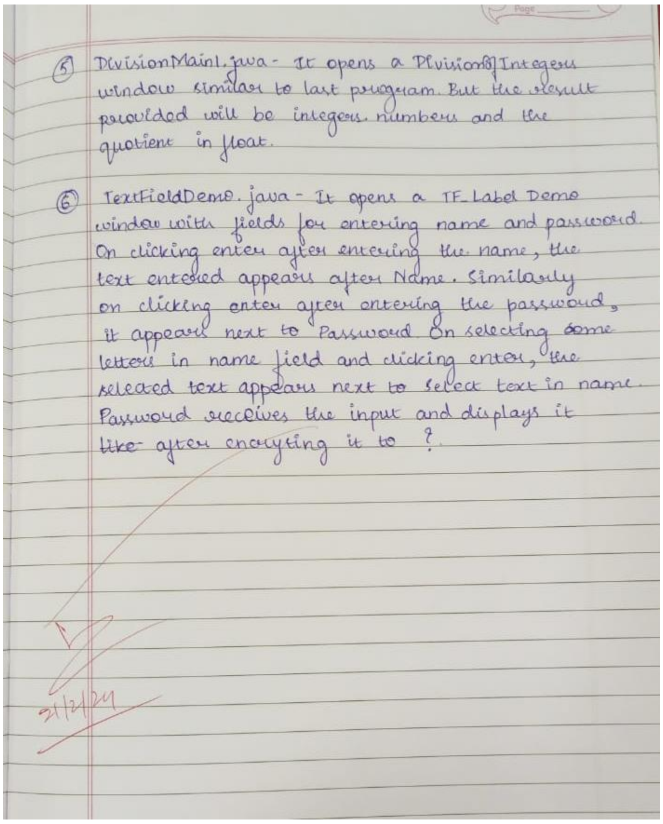
AWT programs - REPORT.

1. buttondrag.java - It opens a Button Game window and provides user with 3x3 tiles with numbers and with buttons for reset, start and restart. Once clicked on start, user can click on two tiles and simultaneously clicked tiles get swapped. User can click on reset to start the game again.

2. ButtonList.java - It opens a ButtonList window with three buttons - yes, no, undecided and has a default text HELLO. Once clicking on yes, the window writes the text - You pressed Yes. Similarly on clicking No, → You pressed No. and on clicking Undecided → You pressed Undecided

3. ButtonListD.java - It opens ButtonListD window with three buttons - yes, no, undecided. On clicking yes, a Dialog window appears with message You pressed Yes and an OK button. Similarly on clicking No, You pressed No and on clicking Undecided, You pressed Undecided in separate dialog boxes.

4. DivisionMain.java - It opens DivisionOfIntegers window with fields to enter 2 numbers and a button RESULT, on clicking which the two numbers and quotient will appear on the window after Result: It gives and intakes numbers in float data type.



5. DivisionMain1.java - It opens a DivisionOfIntegers window similar to last program. But the result provided will be integers. numbers and the quotient in float.

6. TextFieldDemo.java - It opens a TF_Label Demo window with fields for entering name and password. On clicking enter after entering the name, the text entered appears after Name. Similarly on clicking enter after entering the password, it appears next to Password. On selecting some letters in name field and clicking enter, the selected text appears next to Select text in name. Password receives the input and displays it like after encrypting it to ?.

21/2/24