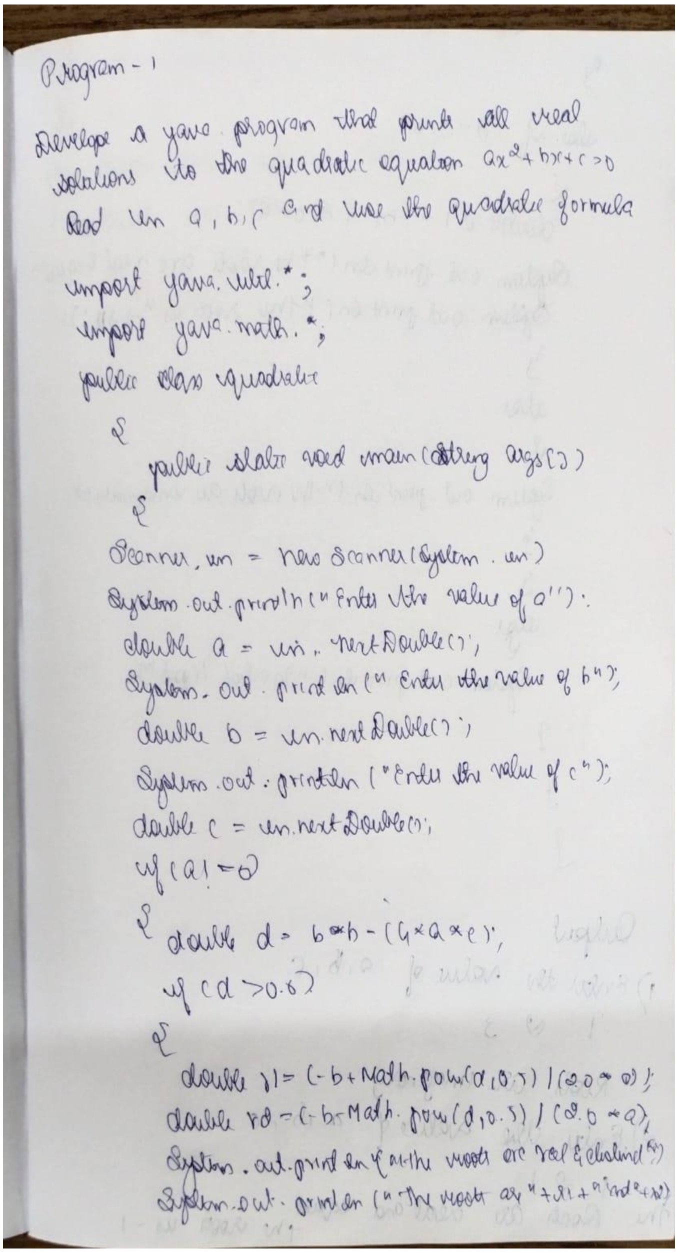
LAB PROGRAM -1

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.

CODE

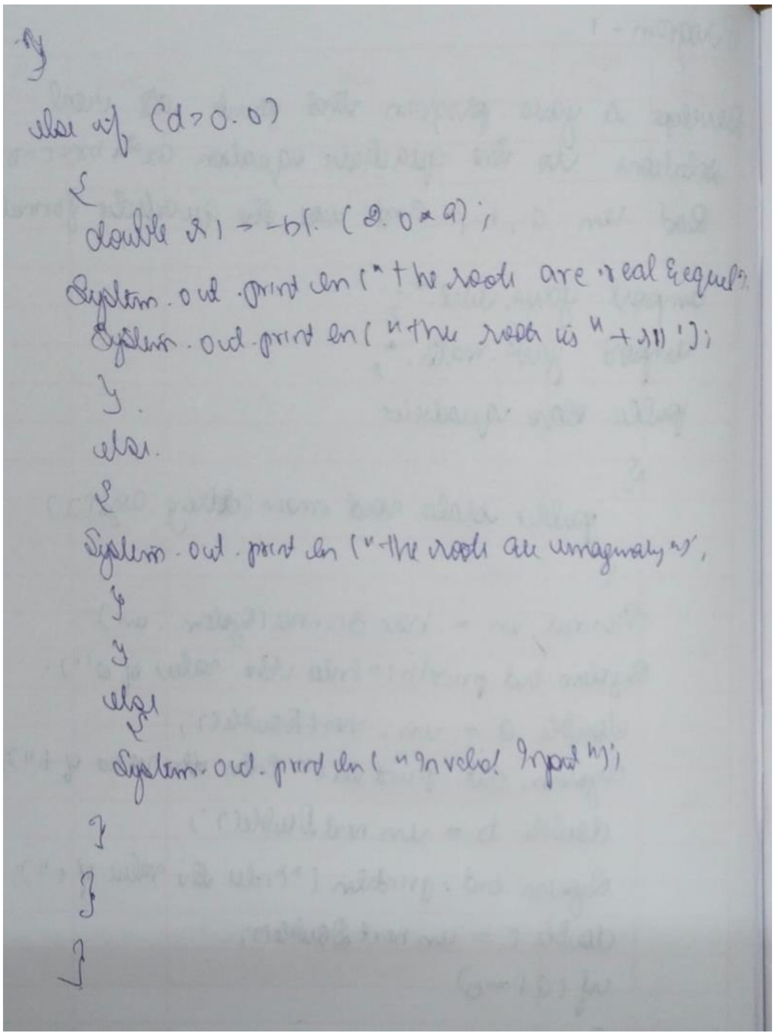
:



Program - 1

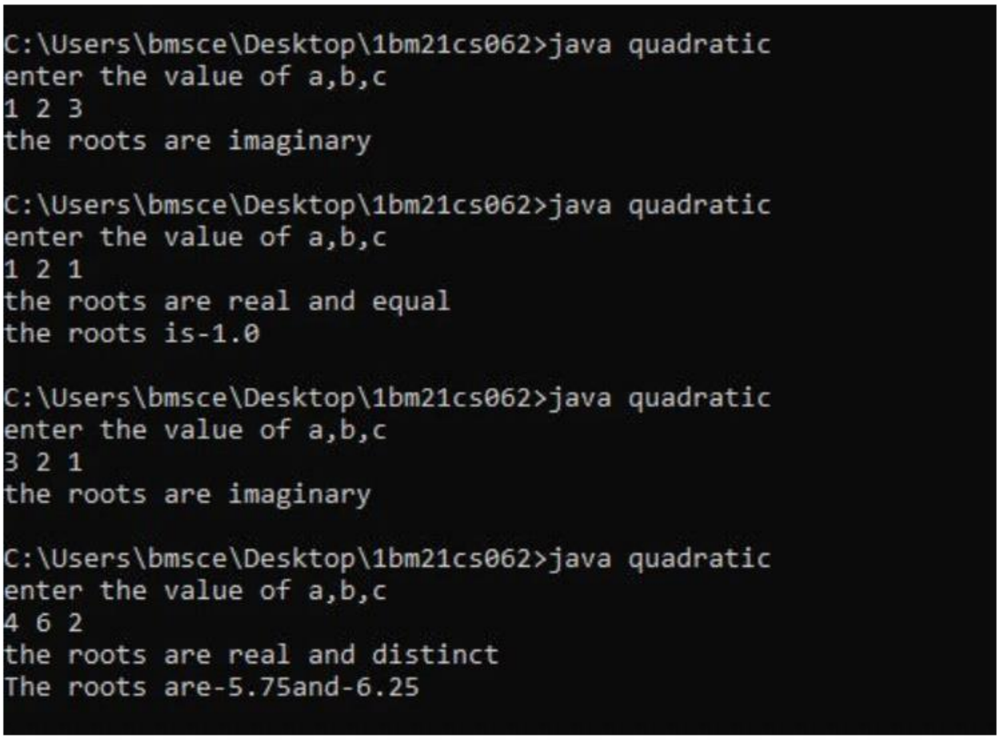
Develope a java program that prints all real solutions to the quadratic equation $ax^2+bx+c>0$ Read in a, b, c and use the quadratic formula

```
import java.util.*;
 import java.math.*;
 public class quadratic
   {
      public static void main(String args[])
    {
  Scanner, in = new Scanner(System.in)
  System.out.println("Enter the value of a"):
   double a = in..nextDouble();
   System.out.println(" Enter the value of b");
    double b = in.nextDouble();
    System.out:println("Enter the value of c");
   double c = in.nextDouble();
   if(a!=0)
   { double d= b*b-(4*a*c);
     if (d>0.0)
     {
       double r1= (-b+Math.pow(d,0.5))/(2.0*a);
      double r2=(-b-Math.pow(d,0.5))/(2.0*a);
      System.out.println("The roots are real & distinct");
      System.out.println("The roots are "+r1+" and "+r2)
```



```
}
else if (d>0.0)
  {
  double r1 = -b/ (2.0*a);
 System.out.println("the roots are real Equal");
  System.out.println("the root is "+r1");
  }.
  else.
  {
  System.out.println("the roots are imaginary");
   }
   }
   else
   {
   System.out.println("Invalid Input");
  }
  }
 }
```

OUTPUT:

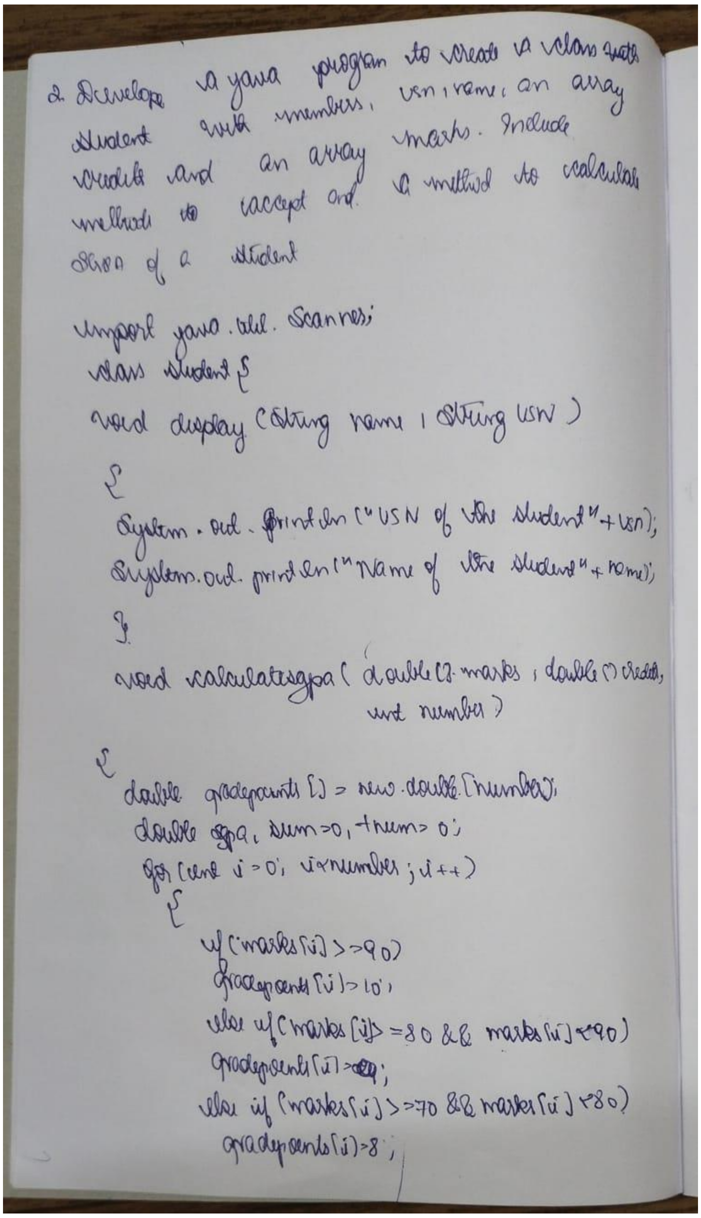


```
C:\Users\bmsce\Desktop\1bm21cs062>java quadratic
enter the value of a,b,c
1 2 3
the roots are imaginary

C:\Users\bmsce\Desktop\1bm21cs062>java quadratic
enter the value of a,b,c
1 2 1
the roots are real and equal
the roots is-1.0

C:\Users\bmsce\Desktop\1bm21cs062>java quadratic
enter the value of a,b,c
3 2 1
the roots are imaginary

C:\Users\bmsce\Desktop\1bm21cs062>java quadratic
enter the value of a,b,c
4 6 2
the roots are real and distinct
The roots are-5.75and-6.25
```

LAB PROGRAM -2

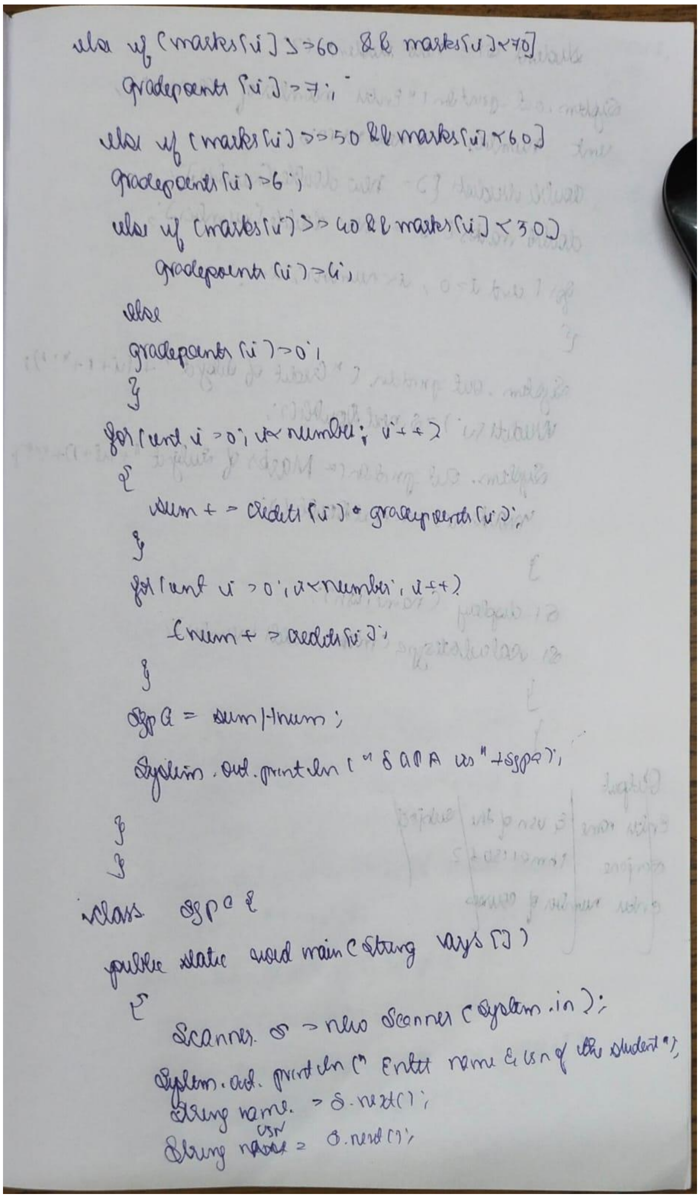
Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

<u>CODE</u>

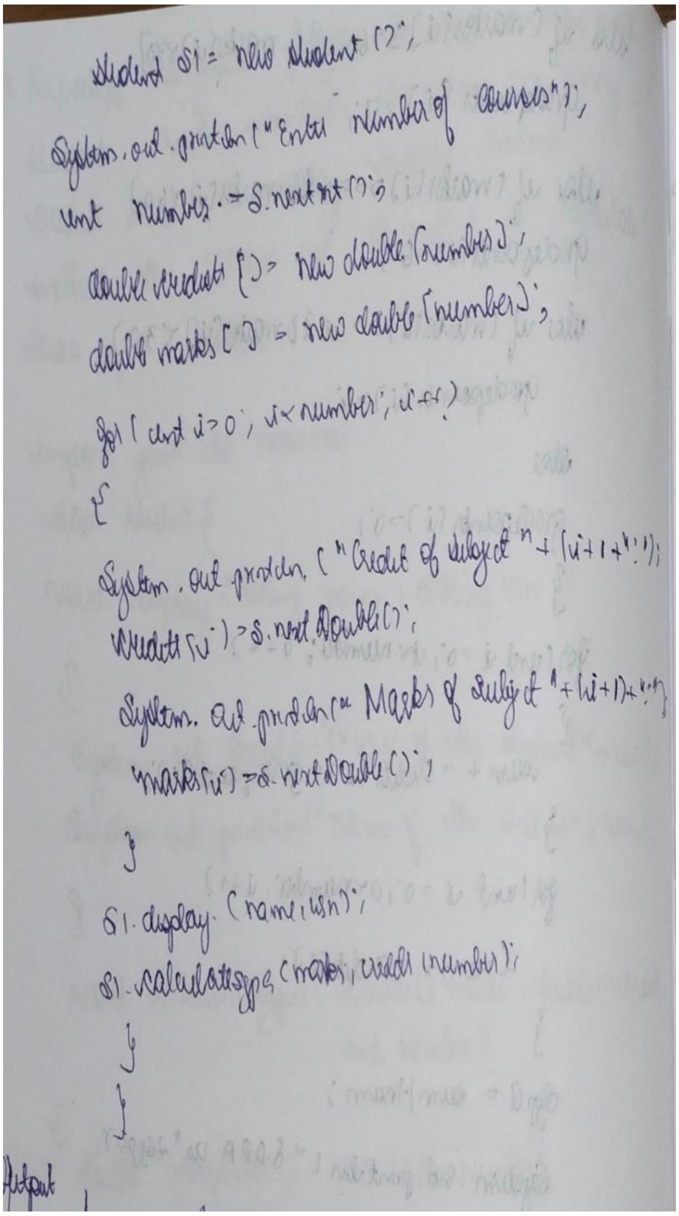


2. Develop a java program to create a class with Student with members, usn, name, an array credits and an array marks. Include methods to accept and a method to calculate SGPA of a student

```
import java.util. Scanner;
class Student {
void display (String name , String usn )
 {
  System.out.println ("USN of the student"+ usn);
  System.out.println("Name of the student"+ name);
 }
 void calculatesgpa ( double [] marks , double [] credits,
                      int number )
 {
  double gradepoints [] = new double [number];
  double sgpa, sum=0, tnum= 0;
  for (int i = 0; i<number ; i++)
   {
      if (marks[i] >=90)
       gradepoints [i]= 10;
      else if (marks [i]>=80 && marks[i]<90)
       gradepoints[i]= 9;
     else if (marks[i] >=70 && marks[i] <80)
       gradepoints[i]=8 ;
```



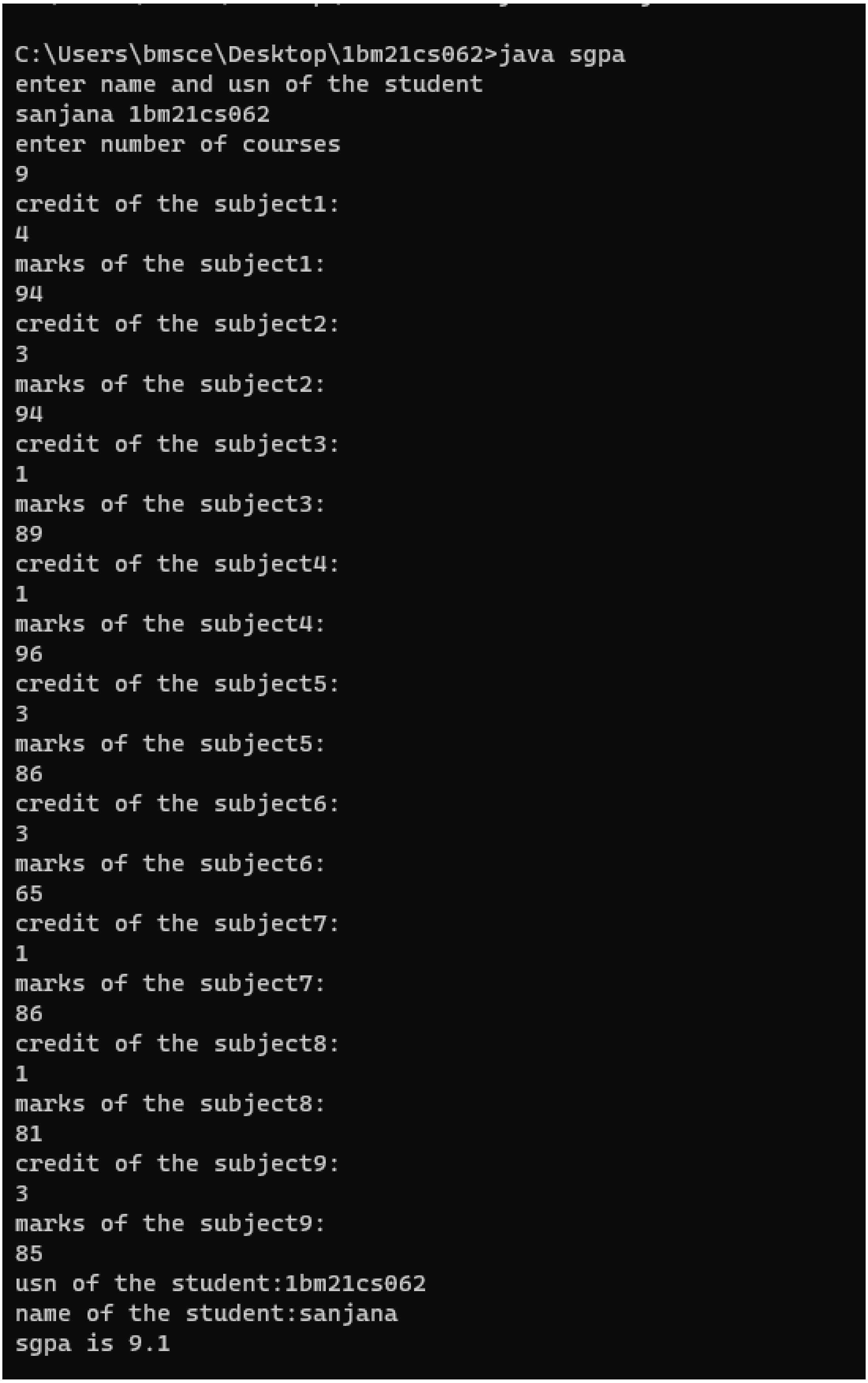
```
else if (marks[i] >=60  && marks[i]<70]
    gradepoints [i] =7;
 else if (marks[i] >=50 && marks[i]<60]
  gradepoints[i] =6;
  else if (marks[i] >= 40 && marks[i] <50]
      gradepoints [i] =4;
  else
  gradepoints [i] =0;
  }
 for(int i =0; i<number; i++)
  {
    sum + = credits [i] * gradepoints [i];
   }
   for(int i =0; i<number; i++)
       tnum + = credits[i];
   }
   sgpa = sum/tnum;
   System.out.println (" SGPA is" +sgpa);

 }
 }
class sgpa {
public static void main(String args[])
 {
    Scanner s = new Scanner (System.in);
  System.out. println (" Enter name & usn of the student");
   String name. = s.next();
                usn
   String name = s.next();
```



```
  Student s1 = new Student ();
System.out.println("Enter number of courses");
int number. = s.nextInt();
 double credits [] = new double [number];
 double marks [ ] = new double [number];

 for ( int i=0; i<number; i++)
 {
  System.out.println ("Credit of subject" + (i+1+":"));
  credits[i] = s.nextDouble();
  System.out.println(" Marks of subject "+(i+1)+":");
  marks[i] =s.nextDouble();

  }
 s1.display (name,usn);
 s1.calculatesgpa (marks, credits, number);
  }
  }
```

Output

OUTPUT:

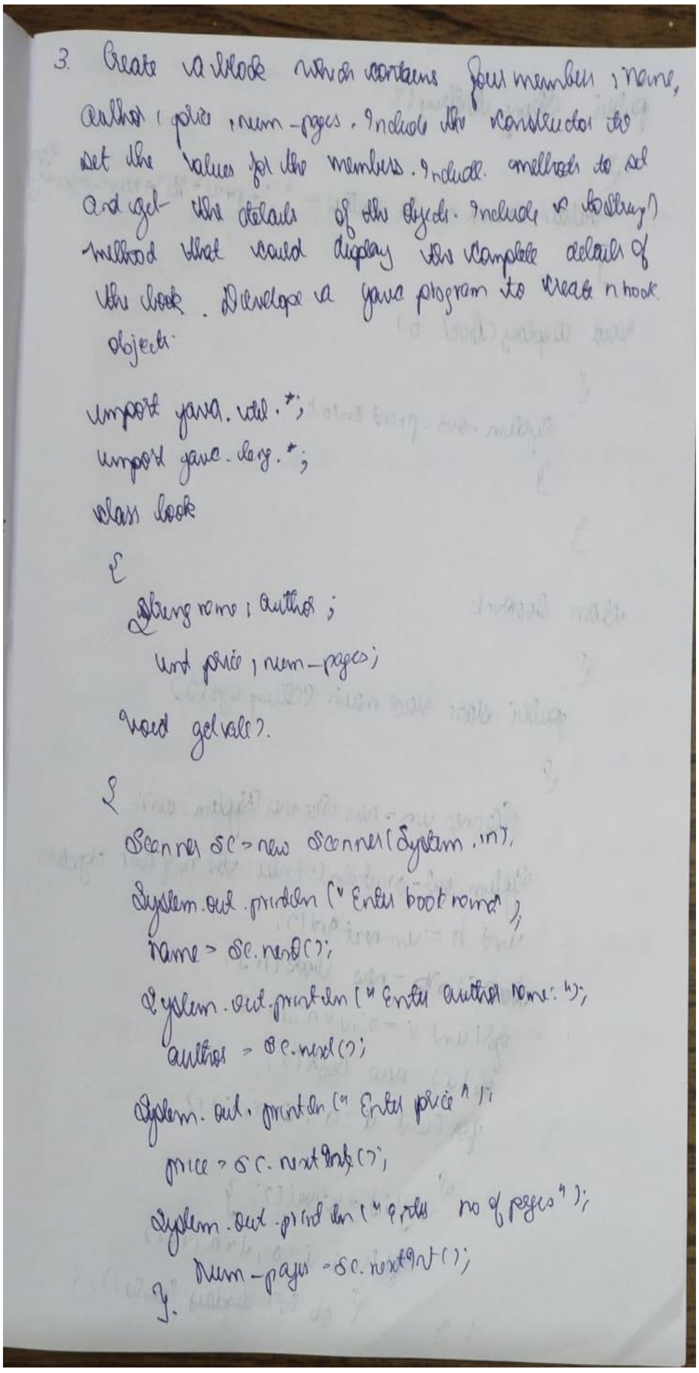


```
C:\Users\bmsce\Desktop\1bm21cs062>java sgpa
enter name and usn of the student
sanjana 1bm21cs062
enter number of courses
9
credit of the subject1:
4
marks of the subject1:
94
credit of the subject2:
3
marks of the subject2:
94
credit of the subject3:
1
marks of the subject3:
89
credit of the subject4:
1
marks of the subject4:
96
credit of the subject5:
3
marks of the subject5:
86
credit of the subject6:
3
marks of the subject6:
65
credit of the subject7:
1
marks of the subject7:
86
credit of the subject8:
1
marks of the subject8:
81
credit of the subject9:
3
marks of the subject9:
85
usn of the student:1bm21cs062
name of the student:sanjana
sgpa is 9.1
```

LAB PROGRAM -3

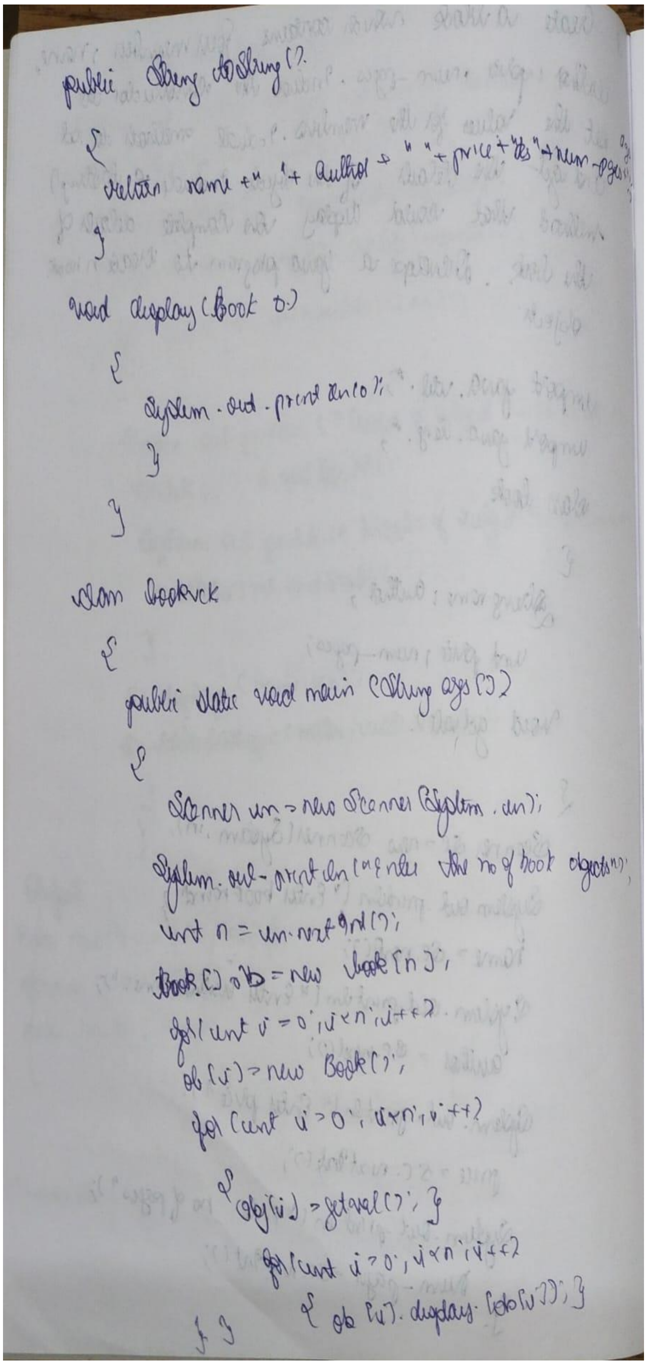
Create a class Book which contains four members: name, author, price, num pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString() method that could display the complete details of the book. Develop a Java program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

<u>CODE</u>



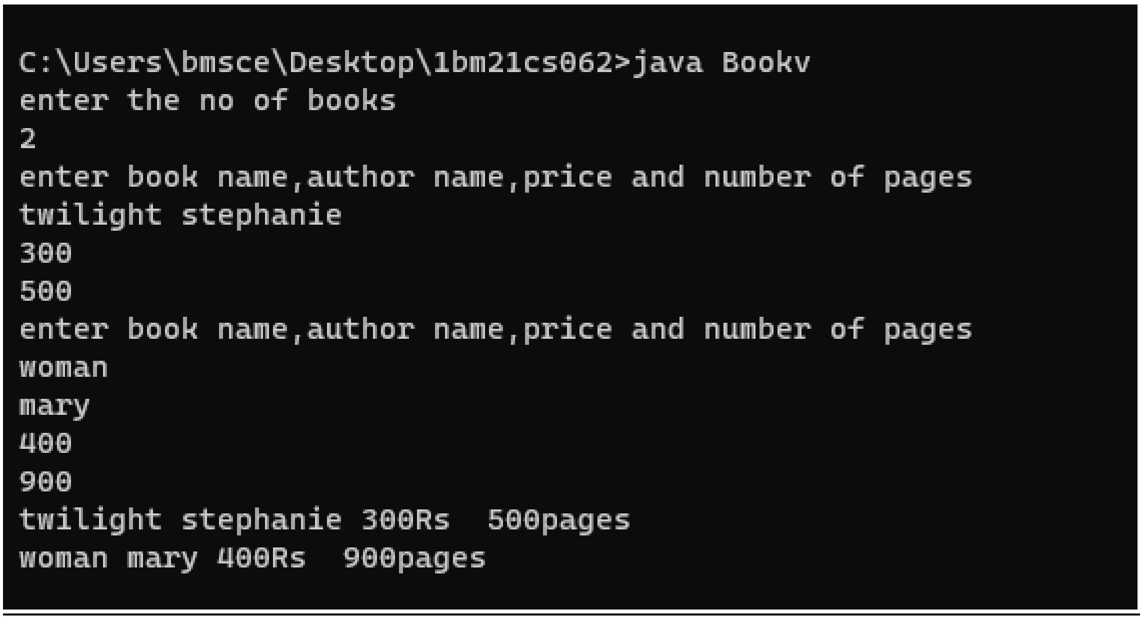
3. Create a book which contains four members : name, author, price, num-pages. Include the constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString() method that could display the complete details of the book. Develop a java program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

```
import java.util.*;
import java.lang.*;
class book
 {
  String name, author;
   int price, num-pages;
 void getval().
 {
 Scanner sc = new Scanner(System.in),
 System.out.println ("Enter book name");
  name = sc.next();
 System.out.println ("Enter author name: ");
   author = sc.next();
 System.out.println (" Enter price ");
   price = sc.nextInt();
 System.out.println (" Enter no of pages ");
    Num-pages = sc.nextInt();
 }.
```



```
public String toString().
 {
  return name +" "+ author + " "+ price + " "+ num-pages;
 }
void display (Book o)
  {
    System.out.println(o);
    }
  }
class Bookrck
 {
  public static void main (String args[])
   {
     Scanner in = new Scanner(System.in);
    System.out.println("Enter the no of book objects");
    int n = in.nextInt();
    Book[] ob = new book[n];
     for(int i = 0; i<n; i++)
     ob[i] = new Book();
      for (int i = 0; i<n; i++)
       { obj[i] = getval(); }
         for(int i = 0; i<n; i++)
           { ob[i]. display (ob[i]); }
    } }
```

OUTPUT:

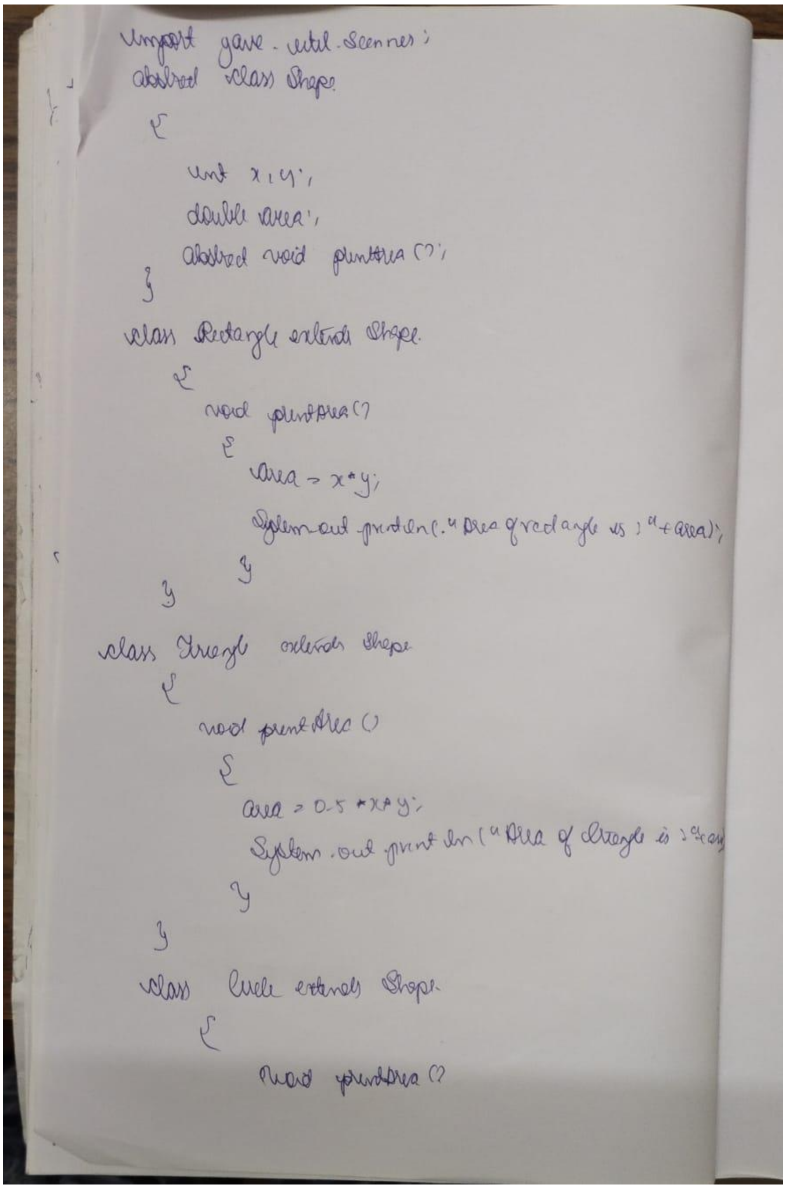


```
C:\Users\bmsce\Desktop\1bm21cs062>java Bookv
enter the no of books
2
enter book name,author name,price and number of pages
twilight stephanie
300
500
enter book name,author name,price and number of pages
woman
mary
400
900
twilight stephanie 300Rs  500pages
woman mary 400Rs  900pages
```

OUTPUT:

Q) 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 extend the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

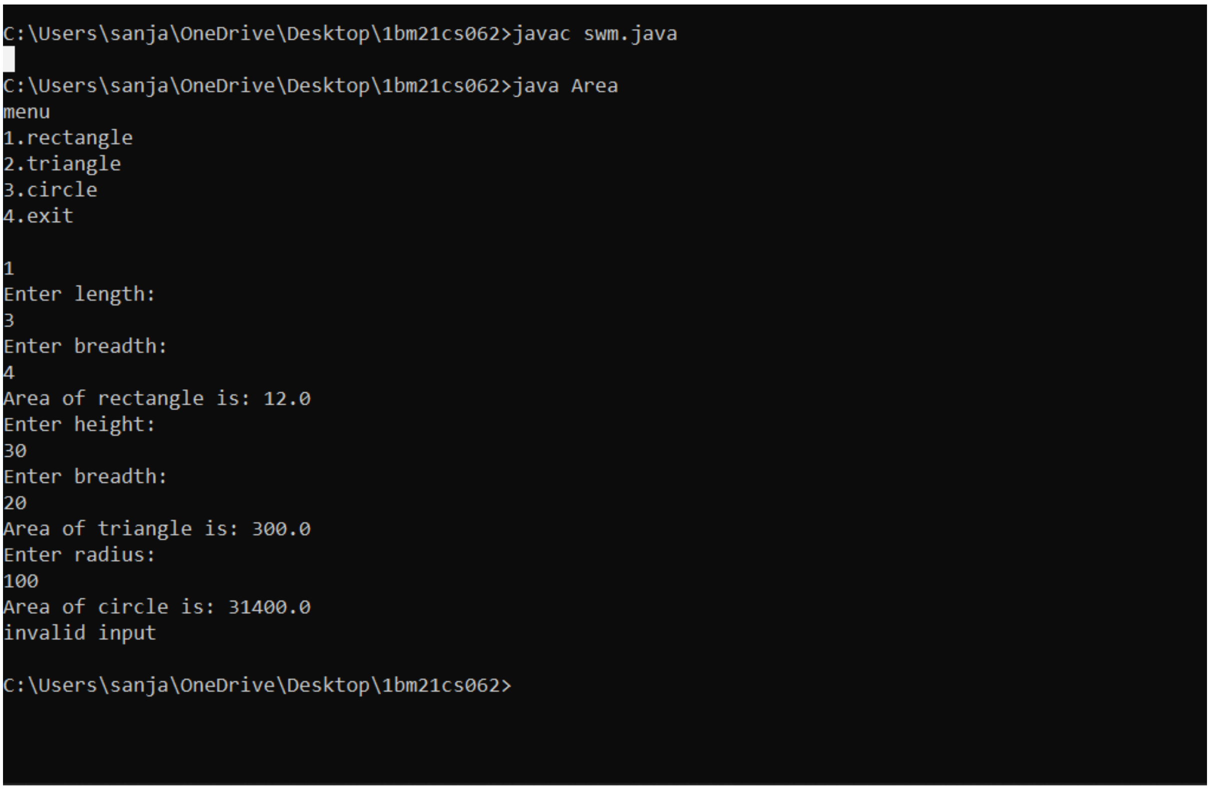
CODE:



```
import jave.util.Scanner;
abstract class Shape
 {
    int x,y;
    double area;
    abstract void printArea();
 }
class Rectangle extends Shape
  {
    void printArea()
     {
       area = x*y;
       System.out.println("Area of rectangle is : "+area);
     }
  }
class Triangle extends Shape
  {
    void printArea()
     {
       area = 0.5*x*y;
       System.out.println("Area of triangle is : "+area
     }
  }
class Circle extends Shape
  {
     void printArea()
```

```
        s{
            Area = 3.14*x*x;
            System.out.println ("Area of circle is "+area);

        }
  }
class Area
{
    p.s.v.m (String args[])
     {
        Rectangle rec = new Rectangle();
        Triangle t = new Triangle ();
        Circle c = new Circle();
        int ch;
        Scanner s = new Scanner( System.in);
       S.O.P ("Menu \n1. Rectangle \n2. Triangle \n3. Circle
              \n 4. Exit \n");
        Ch = s.nextInt();
       switch (ch)

   {
     case 1: S.O.P ( "Enter length & breadth ");
             rec . x = s.nextInt ();
             rec. y = s.nextInt ();
              rec.. printArea ();
    case 2: S.O.P ("Enter breadth & height
            t.x = s.nextInt(); t.y = s.nextInt();
              t.printArea ();
    case 3 : S.O.P ("Enter radius");
               c.x = s.nextInt(); c.printArea();
      default : S. .O.P(" Invalid input ");
     }
}  }
```

OUTPUT:



```
C:\Users\sanja\OneDrive\Desktop\1bm21cs062>javac swm.java

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>java Area
menu
1.rectangle
2.triangle
3.circle
4.exit

1
Enter length:
3
Enter breadth:
4
Area of rectangle is: 12.0
Enter height:
30
Enter breadth:
20
Area of triangle is: 300.0
Enter radius:
100
Area of circle is: 31400.0
invalid input

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>
```

Q) 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.

CODE:

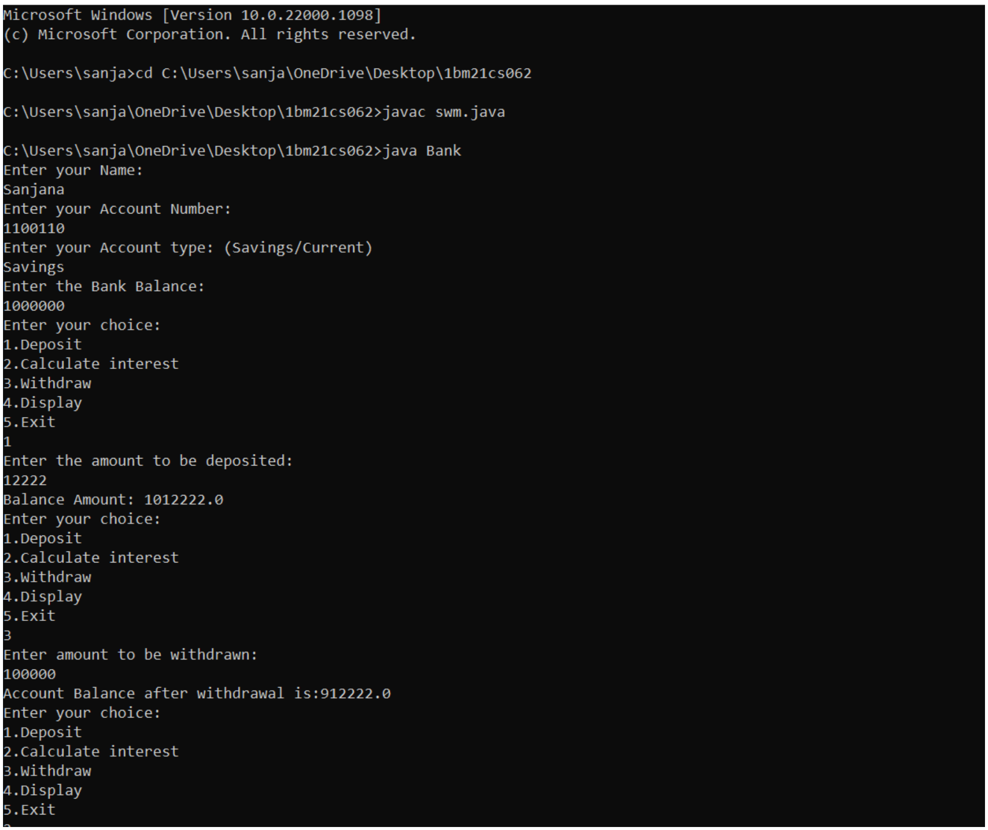
5. Develope a Java program to create a clan Bank that maintain two kinds of account for its customers, one called savings account and the other current account.

```
import java.util.Scanner.
import java.lang.Math;
class Account
  {
     String name, acc_type;
     int acc_no;
      double bal, dep;
     Scanner ss = new Scanner(System.in)
  void set ()
  {
   System.out.println("Enter your name");
      name = ss.next()
    System.out.println(" Enter your account number");
      acc_no = ss.nextInt();
   System.out.println(" Enter your account type =
      (savings, current)");
      acc_type = ss.next();
    ~~S.P;~~ S.O.P("Enter the Bank balance");
          bal = ss.nextDouble();

     }
      void disp()
      {
        S.O.P (" Name:"+name);
         S.O.P (" Account no:"+acc_no);
         S.O.P (" Account type:"+ acc_type);
  S.o.P ("Current Balance is" + bal);
}
void deposit()
  {
   S.O.P ("Enter the amount to be deposited");
   dep = ss.nextDouble();
     bal+=dep;
   S.O.P ("Balance amount:"+bal);

  }

  boolean acc(String acc_type)
      {
          if(acc_type == "Savings")
          return true
        else if (acc_type == "current"){
          return false;
      else
          return true;

      }

 }
 class cur_acct extends Account
      {
        int penal()
         {
             double min ,pen;
            S.O.P ("Enter min amount
          & penalty amount if not followed");
min = ss.nextDouble();
    pen = ss.nextDouble();

  if (bal < min)
    {
      bal -= pen;
    S.O.P ("Penalty imposed for having insufficient
              balance");}

        else
         return 1;
    }
   void withdrawl()
     {
       double amt;
     S.O.P ("Enter amt to be withdrawn");
    amt = ss.nextDouble();
       int a = penal();
    if (a == 1)
     {  if (bal >= amt)
        bal -= amt;
        System.o.p ("Account balance after
         withdrawn is :"+bal;}
     }
   else
   S.O.P("The amount can't be withdrawn");
    }
 }
```

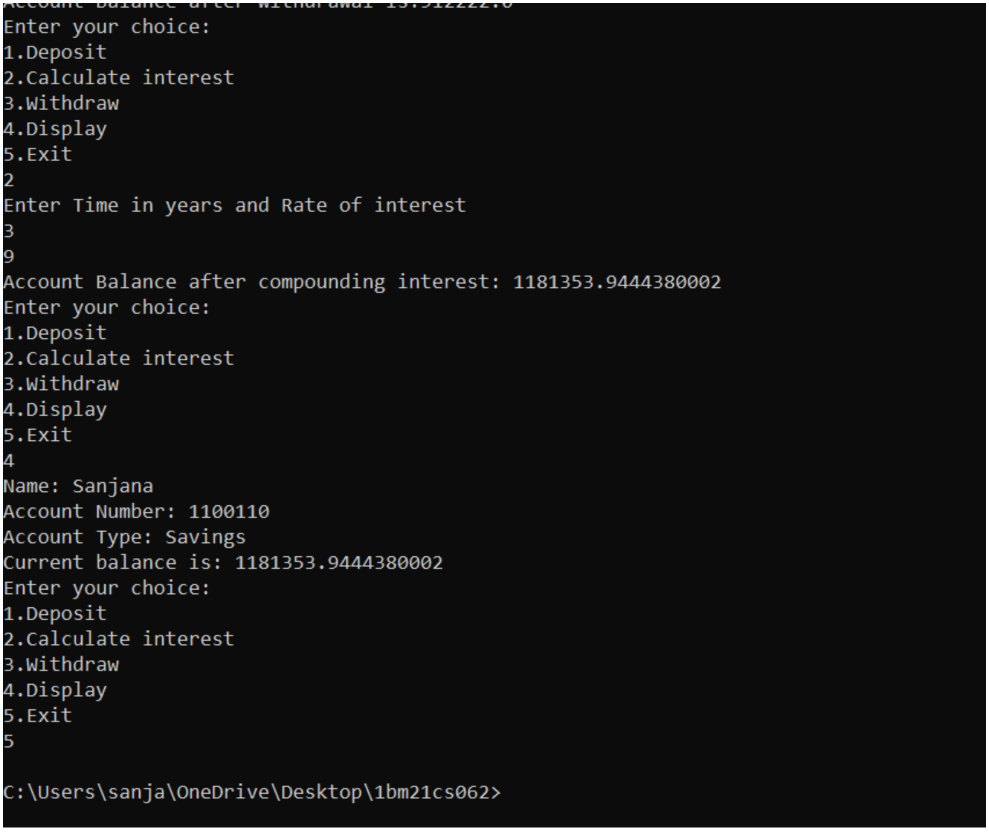
```
class sav_acct extends Account
 {
  void calc_interest()
   {
     S.O.P ("Enter time & rate of interest");
   double t = ss.nextDouble();
   double r = ss.nextDouble();
   double CI = bal * Math.pow((1+r/100),t);
   S.O.P ("Compound interest is" +CI);
    bal += CI;
   S.O.P ("Balance amount:" +bal);
 }
 void withdrawal()
    {
        double amt;
       S.O.P ("Enter amount to be withdrawn");
        amt = ss.nextDouble();
          if (bal >= amt)
           {
              bal -= amt;
  S.O.P ("Account balance after withdrawal"
            +bal); }

     }.
```

```
else
  S.o.p ("the amount can't be withdrawn");
   }
}
class Bank
  {
     p s v m ( String args[]) ;
   {
    Scanner ss = new Scanner(System.in);
      Account a1 = new Account ();
      a1.add();
  if (a1.acc (a1.acc_type) == true)
{
 sav_acc s1 = new sav_acc();
  s1.name = a1.name;
   s1.acno = a1.ac_no;
  s1.ac_type = a1ac_type;
   s1.bal = a1.bal;
 S.o.p("Enter your choice : /n 1.Deposit ; 2.Calculat
       3.withdrawn 4. Display 5.Exit ");
      int ch = ss.nextInt();
      switch (ch)
   {
      case 1: s1.deposit(); break;
    case 2: s1.cal_interest (); break;
    case 3: s1.withdrawal (); break;
     case 4: s1.disp(); break;
     case 5: exit(0); break;
```

OUTPUT:



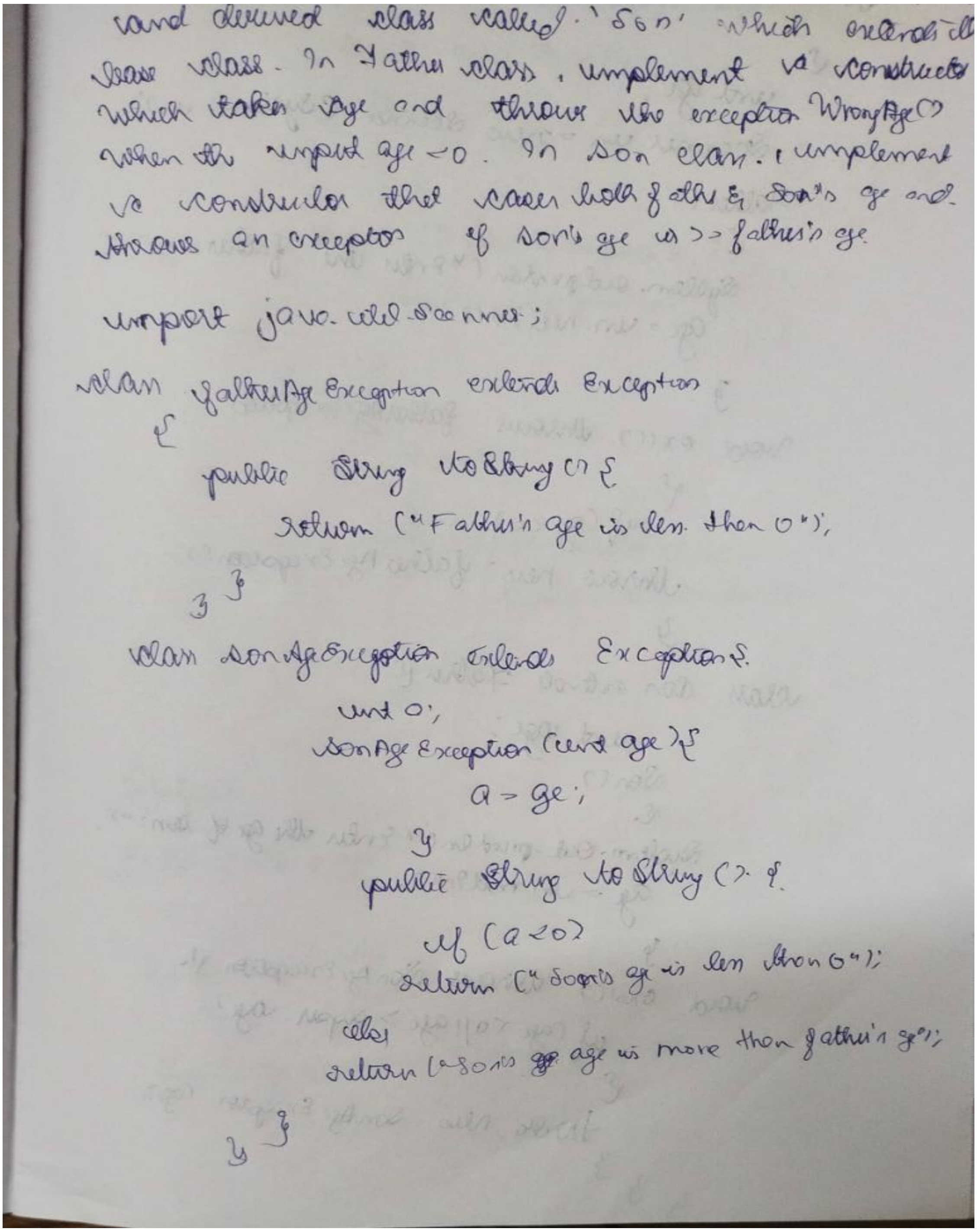
```
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sanja>cd C:\Users\sanja\OneDrive\Desktop\1bm21cs062

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>javac swm.java

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>java Bank
Enter your Name:
Sanjana
Enter your Account Number:
1100110
Enter your Account type: (Savings/Current)
Savings
Enter the Bank Balance:
1000000
Enter your choice:
1.Deposit
2.Calculate interest
3.Withdraw
4.Display
5.Exit
1
Enter the amount to be deposited:
12222
Balance Amount: 1012222.0
Enter your choice:
1.Deposit
2.Calculate interest
3.Withdraw
4.Display
5.Exit
3
Enter amount to be withdrawn:
100000
Account Balance after withdrawal is:912222.0
Enter your choice:
1.Deposit
2.Calculate interest
3.Withdraw
4.Display
5.Exit
```



```
Account Balance after withdrawal is:912222.0
Enter your choice:
1.Deposit
2.Calculate interest
3.Withdraw
4.Display
5.Exit
2
Enter Time in years and Rate of interest
3
9
Account Balance after compounding interest: 1181353.9444380002
Enter your choice:
1.Deposit
2.Calculate interest
3.Withdraw
4.Display
5.Exit
4
Name: Sanjana
Account Number: 1100110
Account Type: Savings
Current balance is: 1181353.9444380002
Enter your choice:
1.Deposit
2.Calculate interest
3.Withdraw
4.Display
5.Exit
5

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>
```

Q) **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.**

CODE:



and derived class called 'Son' which extends the base class. In Father class, implement a constructor which takes age and throws the exception WrongAge() when the input age <0. In Son class, implement a constructor that cases both father & son's age and throws an exception if son's age is >= father's age

```
import java.util.Scanner;

class fatherAgeException extends Exception
  {
     public String toString() {
          return ("Father's age is less than 0");
   }
  }

class sonAgeException extends Exception {
          int a;
        sonAgeException (int age){
                 a = age;
          }
        public String toString() {
             if (a<0)
             return (" Son's age is less than 0");
           else
         return ("Son's age is more than father's age");
    }
  }
```

```
class Father
{
   int age;
   Scanner in = new Scanner (System.in);
  Father()
   {
    System.out.println("Enter the father's age: ");
     age = in.nextInt()

   }
 void ex() throws fatherAgeException
      {
        if (age<0)
      throw new fatherAgeException();
      }
class Son extends Father {
       int age;
      Son()
      {
   System.out.println("Enter the age of son:");
      age = in.nextInt();

      }
 void ex() throws SonAgeException {
        if (age <0|| age > super.age)
        {
        throw new SonAgeException (age);
      }
   } }
```

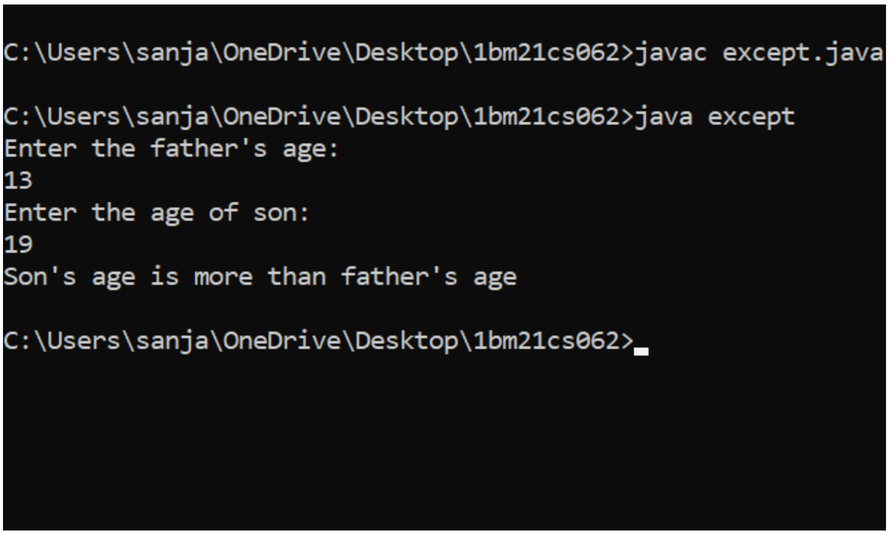
```
class Except {
    public static void main (String[] args) {
            Son s = new Son();
            try {
                s.ex1();
              }
            catch (FatherAgeException e ) {
                System.out.print(e);
              }
            try {
                s.ex2();
              }
              catch (SonAgeException e) {
                        System.out.println(e);
                     }
                 }
              }
```

Output

```
Enter father's age:
  73
Enter son's age:
  99
Son's age is more than father's age.
```

30/12/2022

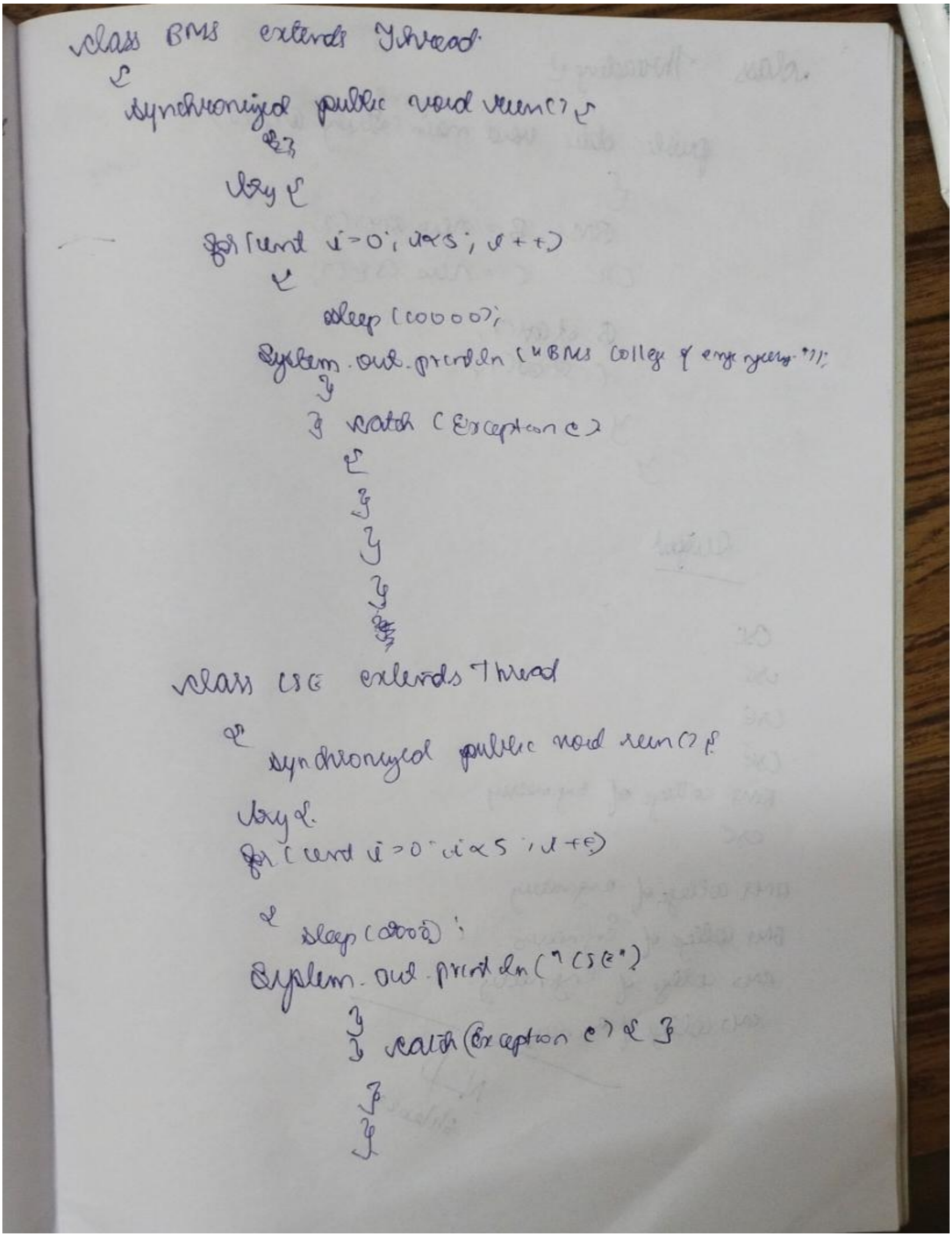
OUTPUT



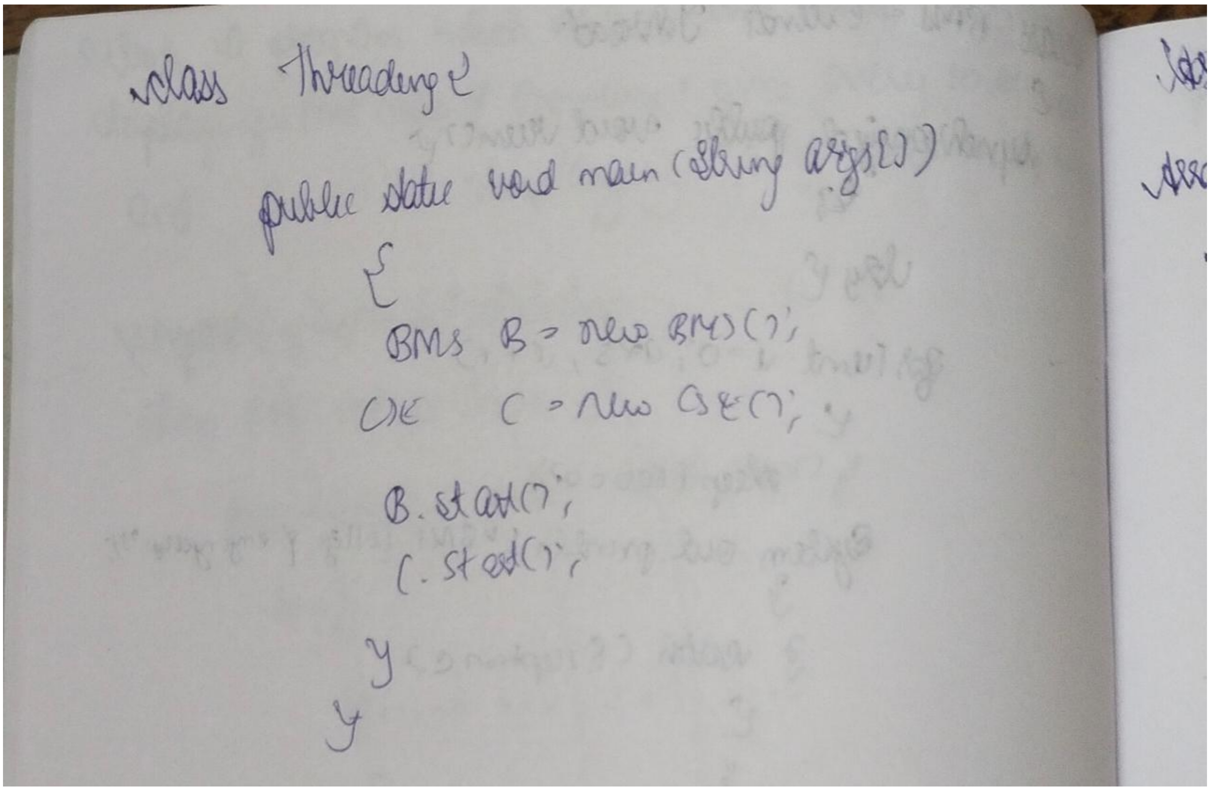
```
C:\Users\sanja\OneDrive\Desktop\1bm21cs062>javac except.java

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>java except
Enter the father's age:
13
Enter the age of son:
19
Son's age is more than father's age

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>
```

Q) **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.**

CODE:

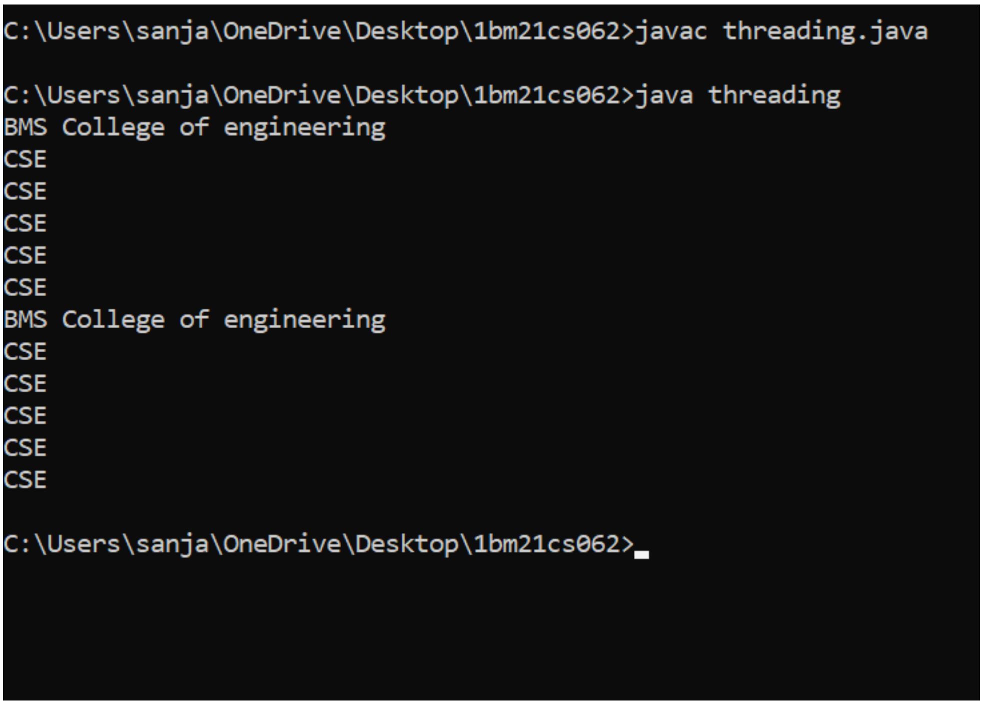


```
class BMS extends Thread
 {
  synchronized public void run() {
          {}
        try {
      for (int i=0; i<5; i++)
         {
            sleep(10000);
       System.out.println("BMS College of engineering");
          }
          } catch (Exception e)
            {
            }
            }
            }
            }
     class CSE extends Thread
      {
        synchronized public void run() {
      try {
      for (int i=0; i<5; i++)
        {
          sleep(2000);
      System.out.println("CSE")
              }
              } catch(Exception e) { }
              }
              }
```



```
class Threading{
    public static void main(String args[])
    {
        BMS B = new BMS();
        CSE C = new CSE();

        B.start();
        C.start();
    }
}
```

OUTPUT



```
C:\Users\sanja\OneDrive\Desktop\1bm21cs062>javac threading.java

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>java threading
BMS College of engineering
CSE
CSE
CSE
CSE
CSE
BMS College of engineering
CSE
CSE
CSE
CSE
CSE

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>
```

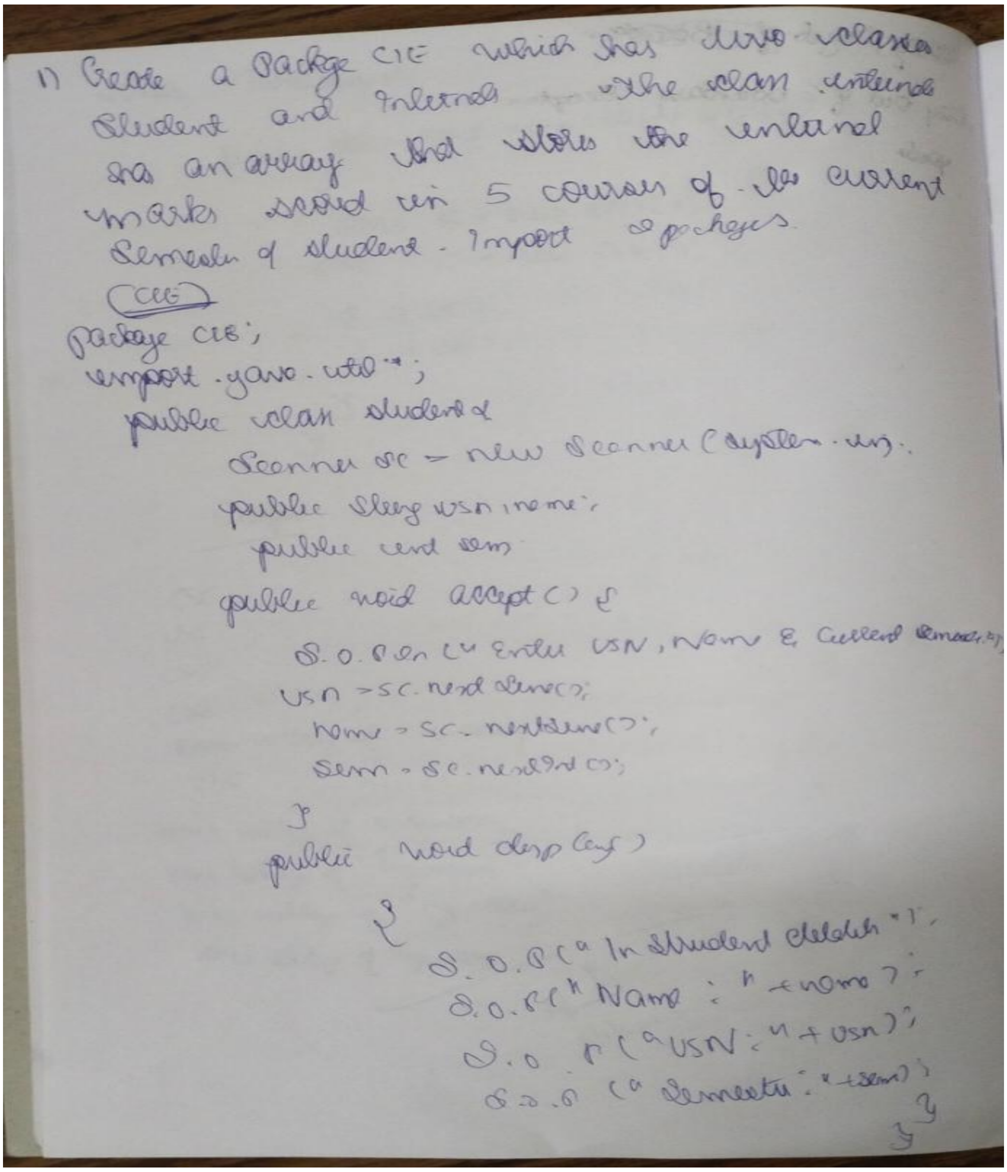
Q) Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal

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.

CODE:



1) Create a Package CIE which has two classes Student and Internals. The class internals has an array that stores the internal marks scored in 5 courses of the current Semester of student. Import packages.

(CIE)

```
package CIE;
import java.util.*;
  public class student{
        Scanner sc = new Scanner(System.in);
      public String usn,name;
        public int sem
      public void accept() {
          S.O.P.ln("Enter USN, Name & Current Semester");
        usn=sc.nextLine();
          name=sc.nextLine();
          sem=sc.nextInt();
        }
      public void display()
           {
               S.O.P("\n Student details");
               S.O.P("Name: "+name);
               S.O.P("USN: "+usn);
               S.O.P("Semester: "+sem);
                                      }
                                    }
```

```
package CIE;
 import jave util.*;
  public class internals extends CIE.student {
      Scanner sc = new Scanner(System.in);
      public int aem = new int[5];
       public void accept()
       {
         int i
        for(i=0;i<5;i++)
         {
          System.out.print("Enter cie marks of
                subject "+(i-1));
           aem[i]=sc.nextInt();
            }
       } }
```

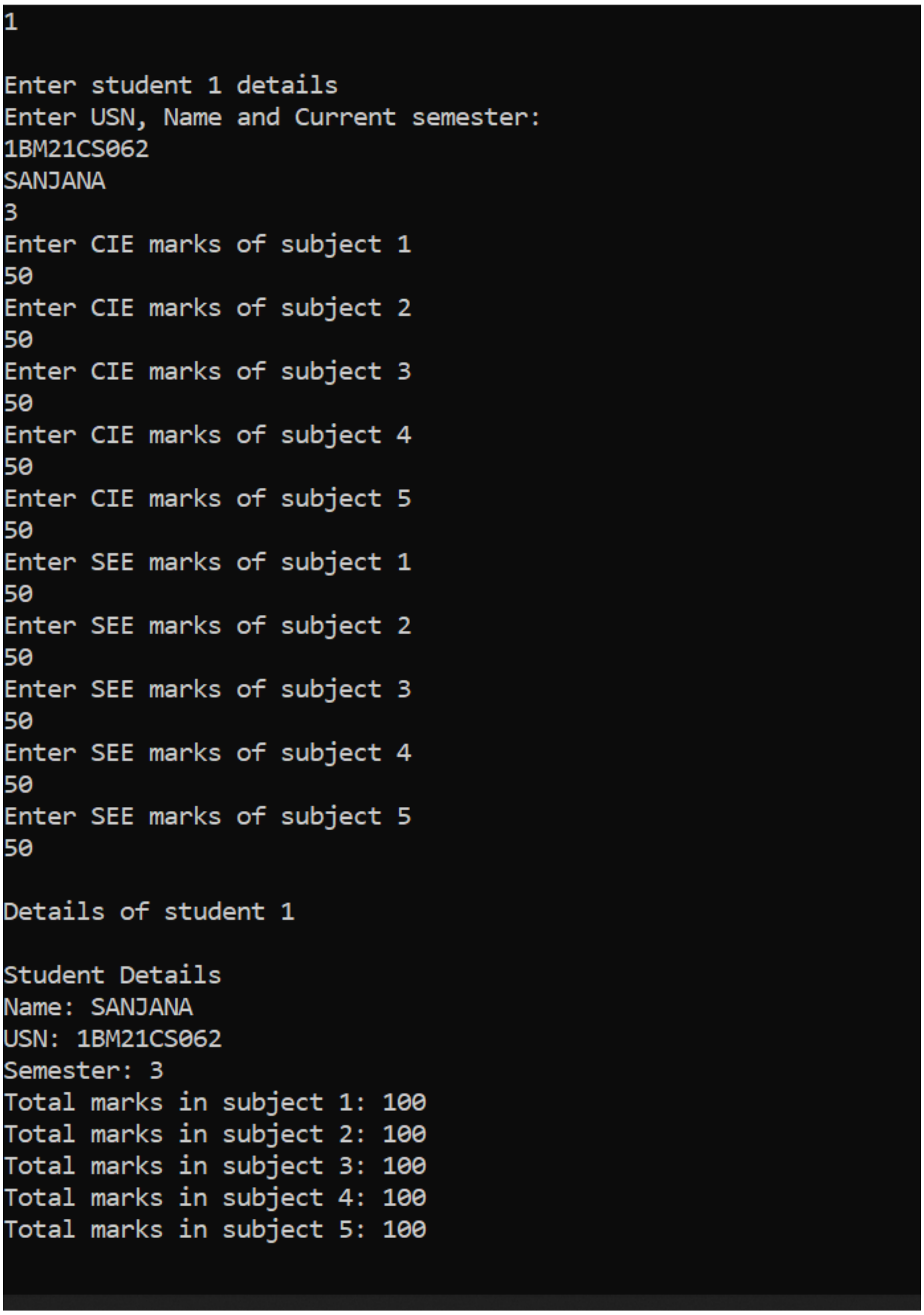
SEE

```
package SEE
 import CIE.*;
  import java.util.*;
  public class externals extends CIE.student {
     Scanner sc = new Scanner(System.in);
     public int seem[] = new int[5];
       public void accept()
  {
```

```
for(int i=0; i<5; i++)
    {
      S.O.P("Enter SEE marks of subject "+(i+1));
      seem[i]=sc.nextInt();
      }
   }
}
```

```
import CIE.*;
import SEE.*;
 import java.util.*;
 class total{
public static void main(String args[])
   {
     int i,j,n;
   Scanner sc= new Scanner(System.in);
   int total[]= new int[5];
   S.O.P("Enter Number of students:");
      n=sc.nextInt();
      CIE.Student s[]= new CIE.Student[n];
      CIE.internals ci[]= new CIE.internals[n];
      SEE.externals se[]= new SEE.externals
      for(i=0;i<n;i++)

       {
         System.out.print("Enter student "+(i+1)+"
                                    details");
s[i] = new CIE.student();
 s[i].accept();
ci[i] = new CIE.internals();
ci[i].accept()
 se[i] = new SEE.externals();
  se[i].accept();
}
for(i=0; i<n; i++)
{
   S.O.Pln("Details of student"+(i+1));
 s[i].display();
 for(j=0; j<5; j++)
{
  total[j] = ci[i].cim[j]+(se[i].seem[j]/2);
  S.O.P("Total marks of student "+(j+1)+": "+total[j]);
   }
   S.O.();
  }
 }
}
```

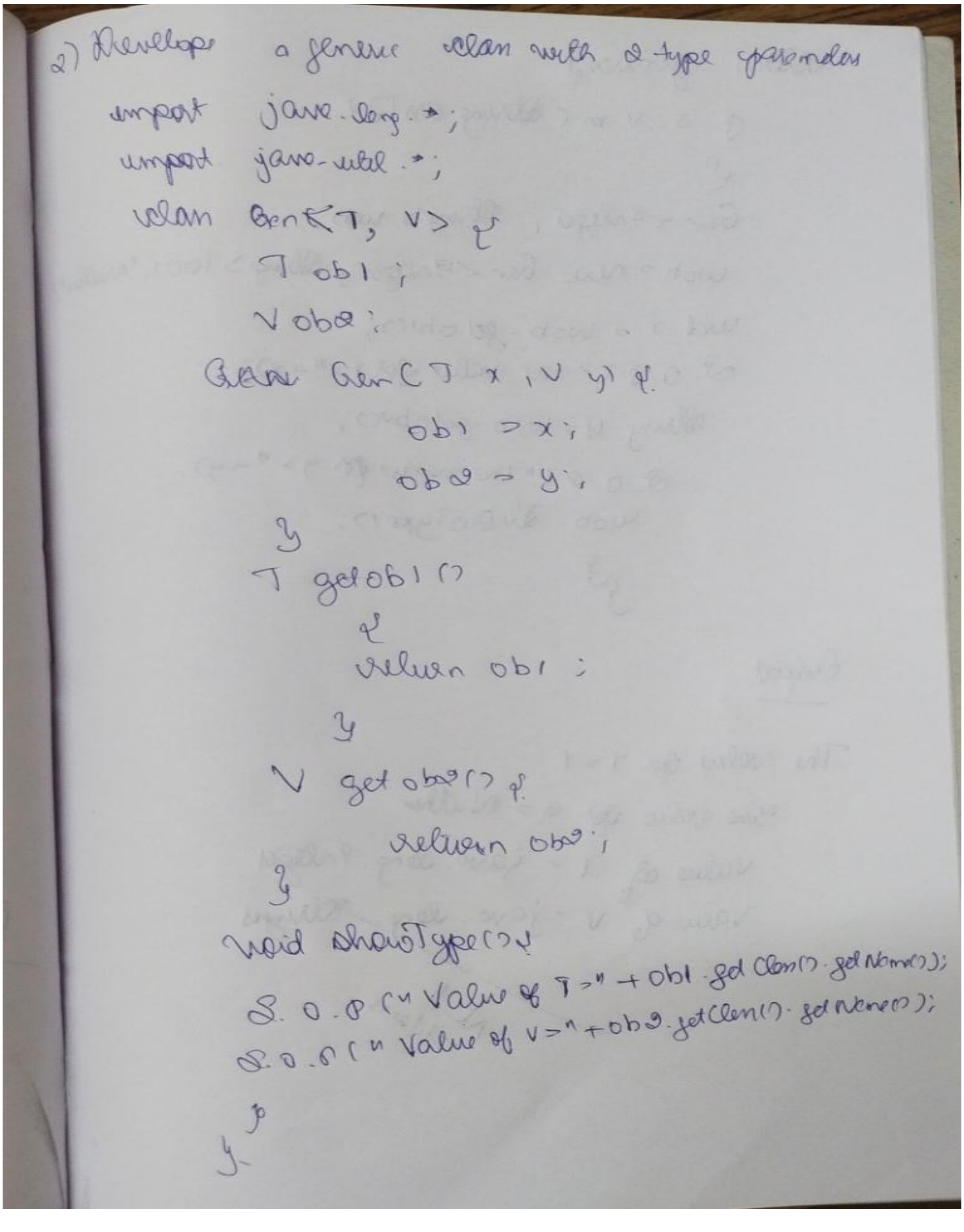
OUTPUT



```
1

Enter student 1 details
Enter USN, Name and Current semester:
1BM21CS062
SANJANA
3
Enter CIE marks of subject 1
50
Enter CIE marks of subject 2
50
Enter CIE marks of subject 3
50
Enter CIE marks of subject 4
50
Enter CIE marks of subject 5
50
Enter SEE marks of subject 1
50
Enter SEE marks of subject 2
50
Enter SEE marks of subject 3
50
Enter SEE marks of subject 4
50
Enter SEE marks of subject 5
50

Details of student 1

Student Details
Name: SANJANA
USN: 1BM21CS062
Semester: 3
Total marks in subject 1: 100
Total marks in subject 2: 100
Total marks in subject 3: 100
Total marks in subject 4: 100
Total marks in subject 5: 100
```

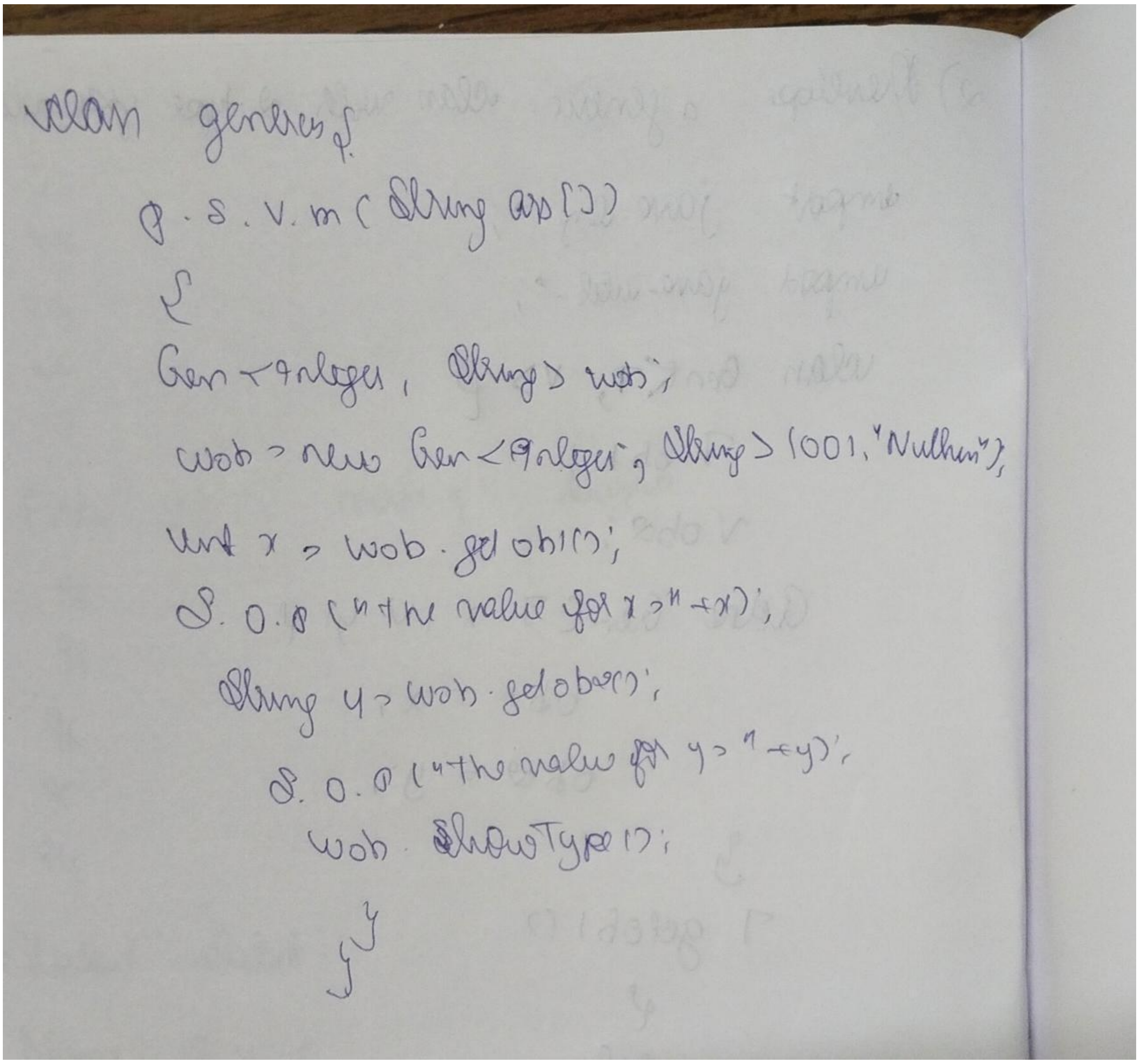
Q) Develop a Generic Class with Two Type Parameters.

CODE:



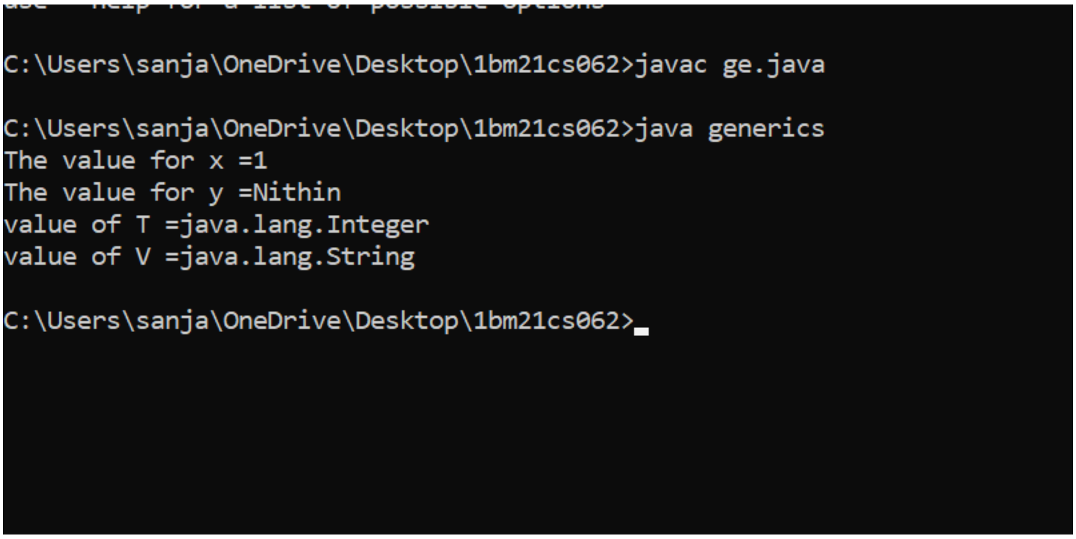
2) Develope a generic class with 2 type paramdes

```
import java.lang.*;
import java.util.*;
class Gen<T, V> {
    T ob1;
    V ob2;
  ~~Gen~~ Gen(T x, V y) {
        ob1 = x;
        ob2 = y;
  }
  T getob1()
     {
     return ob1;
     }
  V get ob2() {
       return ob2;
  }
void showType() {
  S.O.P("Value of T>" + ob1.get Class().getName());
S.O.P("Value of V>" + ob2.getClass().getName());

  }
}.
```



```
class generics {
  p.s.v.m ( String args[])
  {
  Gen<Integer, String> wob;
   wob = new Gen<Integer, String>(001, "Nithin");
  int x = wob.getob1();
  S.o.p("the value for x =" +x);
   String y = wob.getob2();
    S.o.p("the value for y =" +y);
     wob.showType();
      }
  }
```

OUTPUT



```
use --help for a list of possible options

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>javac ge.java

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>java generics
The value for x =1
The value for y =Nithin
value of T =java.lang.Integer
value of V =java.lang.String

C:\Users\sanja\OneDrive\Desktop\1bm21cs062>
```