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.

Develop a Java program that print all the real solutions to quadratic equation $ax^2+bx+c=0$, Read a,b,c. If the discriminate b^2-4ac is negative, display there are no real solutions

```
import Java.util.Scanner;
class QuadraticEq
    {
        public static void main (String XX[])
        {
            double a;
            double b;
            double c;
            double root1, root2;
Scanner SS = new Scanner (System.in);
System.out.print (" Enter values of a, b, c");
a = SS.nextdouble();
b = SS.nextdouble();
c = SS.nextdouble();
double determinant = b*b - 4*a*c
  {
     if (a==0)
     System.out.print (" It is not a quadratic equation")
     else
        {
           if (determinant >0)
            {
               root1 = ((-b + Math.sqrt(determinant))/
                                                  2*a);
               root2 = ((-b+ Math.sqrt(determinant))/
                                                  2*a);
               System.out.println (" The root are
                  distinct and real ; "+root1+" and
                   "+root2);
```

```
if (determinant ==0)
    {
      root1 = root2 = -b/2*a
      System. out. print (" The roots are equal
                                  "+root1);
    }
    if (determinant <0)
      {
        root1 = ((-b + Math.abs ( Math.sqrt
                    (determinant))) /2*a);
        root 2 = ((-b - Math.abs (Math. sqrt
                    (determinant ))) /2*a);
        System.out. print (" The roots are
                    imaginary :"+"i"+root1+ "
                    + "i"+root2);
      }
    }
  }
}
```

```
:\Users\Admin\Desktop\1BM21CS011>javac QuadraticEq1.java

:\Users\Admin\Desktop\1BM21CS011>javac QuadraticEq1.java

:\Users\Admin\Desktop\1BM21CS011>javac QuadraticEq1
rror: Class names, 'QuadraticEq1', are only accepted if annotation processing is explicitly requested
 error

:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
rror: Could not find or load main class QuadraticEq1
aused by: java.lang.ClassNotFoundException: QuadraticEq1

:\Users\Admin\Desktop\1BM21CS011>javac QuadraticEq1.java

:\Users\Admin\Desktop\1BM21CS011>javac QuadraticEq1.java

:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c0 0 0
t is not a quadratic equation
:\Users\Admin\Desktop\1BM21CS011>1 1 1
1' is not recognized as an internal or external command,
perable program or batch file.

:\Users\Admin\Desktop\1BM21CS011>111
111' is not recognized as an internal or external command,
perable program or batch file.

:\Users\Admin\Desktop\1BM21CS011>1
1' is not recognized as an internal or external command,
perable program or batch file.

:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c1 1 1
he roots are imaginary:iNaN iNaN
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c1 4 3
he roots are distinct and real:-1.0and -3.0
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c0 1 2
t is not a quadratic equation
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c1 1 1
he roots are imaginary:iNaN iNaN
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c4 -4 1
he roots are equal:8.0
:\Users\Admin\Desktop\1BM21CS011>javac QuadraticEq1.java

:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c0 2 3
t is not a quadratic equation
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c1 1 1
he roots are imaginary:i0.3660254037844386 i-1.3660254037844386
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c4 -4 1
he roots are equal:8.0
:\Users\Admin\Desktop\1BM21CS011>java QuadraticEq1
nter the values of a,b,c1 4 3
he roots are distinct and real:-1.0 and -3.0
:\Users\Admin\Desktop\1BM21CS011>
```

3) 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.

Q) Create a class Book which contain 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 object. Include a toString() method that could display the complete details of the book. Develop a Java program to create a book of objects.

```
import java.util.Scanner;

class student {
    string USN;
    string name;
    int credits[];
    int marks[];
    double sgpa = 0;
    int crdts = 0;
    int Totcred = 0;

void acceptor (string USN, string name, int
        credits[], int marks[]){
    this.USN = USN;
    this.name = name;
    this.credits = credits;
    this.marks = marks
}
```

```
student ( int credsize){
        credits = new int [credsize];
        marks = new int [credsize];
    }
void display() {
   system.out.print (" usn ; " + usn + "\n" + "
      name : " + name + "\n" + sgpa : " +
      sgpa + " \n");
    }
   void sgpacalc(){
      for ( int i=0 ; i<marks.length ; i++ )
      { if (marks[i] >= 90) crts=10;
        else if ( marks [i] > = 80) crts = 9;
        else if (marks[i] > = 70) crts = 8;
        else if (marks [i] >= 60) crts=7;
        else if (marks [i] >= 50) crts=6;
        else if (marks[i] >= 40) crts 5;
        else if (marks[i] >= 30) crts4;
        sgpa t = crts * credits [i];
        totcred t = credits [i]* 10;
      }
      Sgpa = (sgpa / totcred) * 10;
}   }
}
```

```
class lab-3{
    public static void main (String [] args){
        Scanner s = new Scanner (System.in
        System.out.println (" Enter the no o
                                cources ");
        int n = s.nextInt();
        int credits [] = new int [n];
        int marks[] = new int[n];
        student s1 = new student(n);
        system.out.println("Enter the
                                credits of cources:
    for (int i = 0; i<n; i++){
            credits [i]= s.nextInt();
    }
    system.out.println (" Enter your USN,
                                name ");
        string USN =s.next();
        string skiplines = s.nextline()
        string name = s.nextline()
        system.out.println (" Enter your
                                marks in each subject ");
    }
```

```
for (int i=0 ; i<n ; i++) {
        System.out.print(" Enter the
                marks obtained in
" +i+" course (credits = " + credits[i]+")
                : ");
        marks[i] = S.nextInt();
    }
S1.acceptor (USN, name, credits, marks);
S1. sgpacalc();
S1. display();
S1. close();
    }
}
```

```
C:\Users\BMSCECSEIL74\Documents\aadi>java lab_3
Enter the number of courses
5
Enter the creadits of the courses:
3
3
3
3
4
Enter your  usn,name
ABHINAV KUMAR
1BM21CS003
Enter your marks in each subject
enter the marks obtained in 0 course (credits= 3) :80
enter the marks obtained in 1 course (credits= 3) :89
enter the marks obtained in 2 course (credits= 3) :93
enter the marks obtained in 3 course (credits= 3) :94
enter the marks obtained in 4 course (credits= 4) :95
usn:ABHINAV
name:1BM21CS003
sgpa:9.625
```

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 Java program to create n book objects.

Lab - program 5

```
import java.util.Scanner;
class Book {
    int num_pages;
    double price;
    string name;
    string author;
        Book(){
            num_pages = 0;
            price = 0.0;
            name = "some_book";
            author = "some_author";
        }
    Book (int num_pages, double price, string name,
          string author)
    {
        this.num_pages = num_pages;
        this.price = price;
        this.name = name;
        this.author = author;
    }
    void set_data (int num_pages, double prince,
                   string name string author)
    {
        this.num_pages = num_pages;
        this.price = price;
        this.name = name;
```

```
        this.author = author;
    }
Void get_data() {
        System.out.println("Book detail\n
        name:" + name + "\n author:
        + author + "\n number of pages" +
        num_pages + "\n price : " +
        price);
    System.out.println("\n ---- \n")
Public string tostring() {
    return ("Book detail is name: " + name
        +"\n author : " + author \n no of
        pages " + num_pages + "\n
        price : " + price);
    }
}
class lab_2 {
    public static void main (string[] args)
    {
        Book b1 = new Book();
        Scanner s = new Scanner (system
                                .in);
```

```
system.out.print("enter the name of book:");
string name = s.nextline();
system.out.print("enter author's name:");
string author = s.nextline();
system.out.print("enter the no of pages:");
num_pages = s.nextInt();
system.out.print("enter the price:")
double price = s.next double();
System.out.println();
b1.set_data(num_pages, price, name,
author)
book b2 = new book(20, 87.65, "Deception
point", "Dan Bron");
b1.get_data();
b2.get_data();
system.out.println(b1);
system.out.println(b2);
s.close()
}
```

```
C:\Program Files\Java\jdk1.8.0_231\bin>java lab_2
eneter the name of the book: dark
eneter the author's name: james
eneter the number of pages in the book: 456
eneter the price of the book: 370

Book details
name: dark
author: james
nmber of pages: 456
price: 370.0

--------------------------------

Book details
name: deception point
author: dan brown
nmber of pages: 20
price: 87.65

--------------------------------

Book details
name: dark
author: james
nmber of pages: 456
price: 370.0
--------------------------------

Book details
name: deception point
author: dan brown
nmber of pages: 20
price: 87.65
--------------------------------
```

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.

4) 9-12-22 LAB program - 4

Develop a Java program to create abstract class named shape that contain 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 print the area of the given shape

```
import java.util.Scanner();
import java.lang.Math.*;

abstract class Shape{
        public int a;
        public int b;
        abstract public void printArea();
        Scanner s = new Scanner(System.in);

class rectangle extends Shape{
        public void printArea(){
        System.out.print("Enter length and
                                breadth");
        float a = s.nextFloat()
        float b = s.nextFloat();
        float area = a*b;
        System.out.println("Area = " +
            area + " sq. units");
        }
}
```

```
class tring extends shape{
    public void printArea()
    System.out.print("Enter the three sides")
    float a = s.nextfloat();
    float b = s.nextfloat();
    float c = s.nextfloat();
    float d = (a+b+c)/2;
    double area = Math.sqrt(d*(d-a)*
                  (d-b)*(d-c));
    System.out.println("Area =" +area+"
                        sq.units");
    }
}
class figure{
    public static void main(String args[]){
class circle extends shape{
    public void printArea(){
System.out.print("Enter the radius of
                  circle");
    float a = s.nextfloat()
    float area = 22/7 *a*a;
    System.out.println("Area =" +area+"
                        sq.units");
    }
}
```

```
class figure {
public static void main { string args[
        shape r = new rectangle();
        shape t = new tringle();
        shape c = new circle();
    for(int i=0; i<100; i++){
system.out.println(" \n 1) Triangle \n 2)
                    Rectange \n 3) circle
 System.out.println(" Enter your choice;
 Scanner S = new Scanner(System.in);
     int ch = S.nextInt();
      switch (ch) {
          case 1 : t.printArea();
                   break;
          case 2; r.printArea();
                   break;
          case ;  c.PrintArea();
                   break;
          default; System.out.println
                   ("Invalid choice");
                }
            }
        }
    }
```

Output

1) Triangle
2) rectangle
3) Circle
Enter your choice : 1
Enter three sides of triangle : 4 5 6
Area = 9.921567411 sq.units

1> Triangle
2> rectangle
3> Circle
Enter your choice : 2
Enter the lengths and breadth of rectangle : 34 5
Area = 170.0 sq units

1> Triangle
2> rectangle
3> circle
Enter your choice : 3
Enter the radius of circle : 10
Area = 300.0 sq.units

9/12/22

```
C:\javaprograms>javac figure.java

C:\javaprograms>java figure

1)Triangle
2)Rectangle
3)Circle
Enter your choice:
1
Enter three sides of triangle: 2 3 5
Area=0.0sq.units

1)Triangle
2)Rectangle
3)Circle
Enter your choice:
2
Enter length and breadth of rectangle: 2 4
Area=8.0sq.units

1)Triangle
2)Rectangle
3)Circle
Enter your choice:
3
Enter radius of circle: 4
Area=48.0sq.units

1)Triangle
2)Rectangle
3)Circle
Enter your choice:
```

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.

Q) one saving and other current acc. The saving account provides compound interest and withdrawl but no cheque books. current account provide cheque book facility but no interest. current account should hold a min balance below this level charge is imposed.

Include necessary methods (a). Accept deposit from customer (b) display balance, (c) compute and deposit, (d) Permit withdra check for minimum balance and if necessary penalty update balance.

```
Import Java.util.*;
class Bank
    {   public string name;
        public int acc_no;
        public float bal;
        public float si;
    public void accept()
        {
            Scanner s = new Scanner (system.in
            System.out.print("\n Enter name of
                                the acc holder:");
            name = s.next()
            system.out.print("Enter the account
                                number:");
            acc_no = s.nextInt();
            system.out.print("Enter the accou
                                balance:");
            bal=s.nextfloat();
        }
    public void display()
        {
            System.out.println("Details");
        System.out.println("Name:"+Name+
    \n Account number:"+acc_no+"\n
```

```
public void simple_interest()
    {
        System.out.println("\n Rate of Interest
                                = 8%");
        si = (bal * 8)/100;
        System.out.println(" simple interest
                        (for one year) = Rs"+si);
    }
}
class savings extends bank
{
    public void cheque()
    {
        System.out.println("No cheque services");
    }
    public void withdrawal()
    {
        float amount;
        Scanner a = new Scanner(System.in);
        System.out.println(" No minimum
                            balance
                            required ");
        System.out.print(" Enter amout
                        to be withdrawn: ");
        amount = a.nextfloat();
        if (amount > super.bal)
        {
            System.out.println(" Balance
                        is insufficient");
        }
        else
        {
```

```
super.bal = super.bal - amount;
System.out.println("amount + "
                    withdrawn");
system.out.println(" Available balance
                    = "+super.bal);
    }
  }
}
class current extends bank
{
    public void cheque()
    {
        System.out.println(" Cheques servic
                    available ");
    }
    public void withdrawal()
    {
        float amount;
        Scanner a = new Scanner (System.in
                    );
        System.out.println(" Minimum
                    balance = RS
                    1000.00");
        if (super.bal < 1000)
    {   System.out.println (Balance is in sufficient
                    to withdraw");
        float servic_charge;
        service_charge = (1* super.bal)/100;
        super.bal = super.bal - servic_charge;
system.out.println (" Servic charge of RS " + servic
                    charge +" is
                    added");
} system.out.println (" Availabic balance = RS " +
                    super.bal);}
```

```
else
{
    system.out.print("Enter amount to be
                        withdrawn: ");
    amount = a.nextFloat();
    if (amount > (super.bal - 1600))
    {
        system.out.println("Balance is
                        insufficient");
    }
    else
    {   super.bal = super.bal - amount;
        system.out.println(amount+"
                        withdrawn");
        System.out.println("Available
          balance = " + super.bal);
    }
}
}
}
class Main {
    public static void main(String args[])
    {
        Savings obj1[] = new Savings[3];
        Current obj2[] = new Current[3];
        System.out.print("Enter the
                        no of accounts: ");
        Scanner x = new Scanner(System.in);
        int n = x.nextInt()
        int i = 0
        int j = 0
        int k = 0
        while (i<n)
    {   System.out.println("\nAccount"
                        +(i+1));
        system.out.println("1) Savings
                        2) Current");
```

```
System.out.print(" Enter the type of account",);
int ch = x.nextInt();
if (ch==1)
    {
        obj1[j] = new savings();
        obj1[j].accept();
        obj2[j].display();
        obj1[j].cheque();
        obj1[j].simple interest();
        obj1[j].withdrawal();
        j++;
    }
else
    {
        obj2[k] = new current();
        obj2[k].accept();
        obj2[k].display();
        obj2[k].cheque();
        obj2[k].withdrawl();
        k++;
    }
i++;
}
}
}
```

output

Enter the number of account : 1

Account 1

1> Savings

2> Current

Enter the type of account : 1

Enter the ~~tot~~ name of account holder : Abhi

Enter the account number : 290478

Enter the account balance : 7000

* Details *

Name : Abhi

Account number : 2904768

Balance : 7000.0

No cheque services

Rate of interest = 8%

simple interest < for one year > = Rs 560.0

No minimum balance required

Enter the amount to be withdrawn : 3000

3000.0 withdraw successfully

Available balance = 4000.0.

Note all possible o/p options

```
C:\javaprograms>java Main
Enter the number of accounts: 1

Account 1

1)Savings
2)Current
Enter the type of account: 2

Enter the name of the account holder: abhi
Enter the account number: 129067
Enter the account balance: 12000
*Details*
Name: abhi
Account number: 129067
Balance: 12000.0

Cheque services available

Rate of interest= 8%
Simple interest(for one year)= Rs960.0

Minimun balance= Rs.1000.00
Enter the amount to be withdrawm: 1200
1200.0 withdrawm
Available balance= 10800.0
```

```
C:\javaprograms>java Main
Enter the number of accounts: 1

Account 1

1)Savings
2)Current
Enter the type of account: 1

Enter the name of the account holder: kani
Enter the account number: 2980167
Enter the account balance: 2000
*Details*
Name: kani
Account number: 2980167
Balance: 2000.0

No cheque services

Rate of interest= 8%
Simple interest(for one year)= Rs160.0
```

6) 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.

Lab program -6

Q) A program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class "Son" which extends the base class. In Father class implement a constructor which takes the age and throw the exception. WrongAge() when the input is age<0. In Son class, implement a constructor that cases both father and son's age and throwns an exception if son age is >= father's age

A)

```
import java.util.*;
class FatherAgeException Extends Exception
        { public string tostring(){
            return("Father age less than0")
        }
}
class sonAgeException extends Exception{
        int a;
        sonAgeException(int age){
            a = age;
        }
        public string tostring(){
            if (a<0)
              return("son's age is less
                      than0");
            else
              return("son's age is
                      more than father's
                      age");
        }
}
```

```
class Father{
        Public int age1;
        Scanner s = new Scanner (System.in)
         father() {
                    System.out.println(" Enter Father's
                                                age : ");
                     age1 = s.nextInt();
                  }
                void ex1() throws FatherAge Exception{
                        if (age1<0)
                          throw new fatherAge Exception
                                                    ();
                } }
        }
Class son  extends father{
        public int age2;
          son() {   system.out.print(" Enter son's age:
                                                  ");
                      age 2 = s. next In();
          Void ex2() throws SonAge Exception{
             if ( age2<0 || age 2 >super.age1)
                throw new sonAge Exception(age 2);
          } }
    }
Class Father son{
          Public static void main (string args[])
          { father f = new father();
            son s = new son();
              try {
                    s1.ex1();
                    f1. ex1();
              }
              catch ( Father AgeException e)
                    { system.out.println(e); }
          try { s. ex2();}
          catch ( son Age exception e) {
                        System.out.print(e);
          }
} } }
```

Output

```
Enter father's age: -2
Enter son's age: 5
Father age is less than 0
Son's age is more than father age

Enter father's age: 40
Enter son's age: 50
Son's age is more than father's age

Enter father's age : 40
Enter son's age : 15
```

```
C:\javaprograms>java fatherson
Enter father's age: 40
Enter son's age: 15

C:\javaprograms>java fatherson
Enter father's age: -3
Enter son's age: 12
Father's age is less than 0
Son's age is more than father's age

C:\javaprograms>java fatherson
Enter father's age: 30
Enter son's age: 40
Son's age is more than father's age

C:\javaprograms>java fatherson
Enter father's age: 40
Enter son's age: 40
```

7) 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.

(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.

A)

```
class bms implements Runnable {
        Thread t1;
        bms() {
            t1 = new Thread (this, "bms");
        }
    public void run() {
        try {
            for (int i=5; i>0; i--){
            System.out.println (" BMS
                college of Engineering ");
                Thread.sleep (10000);
            }
        }
        catch (InterruptedException e){
            system.out.println (" Bms interrupted");
        }
        System.out.println ("Exiting :" t1);
    }
}
class csc implements Runnable {
        Thread t2;
        csc() {
            t2 = new thread (this, "csc");
        }
    public void run(){
        try {
            for(int i=5, i>0; i--)
            system.out.println
                ("CSE");
            Thread.sleep(2000);
        }
    }
```

```
        catch (InterruptedException e){
        System.out.println ("CSE interrupted\n");
            }
        System.out.println("Exiting : " +t2);
    }
}
class threadprg {
    public static void main (String args[])
        {
        bms obj1 = new bms();
        cse obj2 = new cse();
        obj1.t1.start();
        obj2.t2.start();
    }
}
```

output:

```
Bms
CSE
CSE
CSE
CSE
CSE
CSE
Bms
Exiting: Thread [cse,5,main]
Bms
Bms
Bms
Exiting: Thread [bms,5,main
```

```
C:\Users\BMSCECSEIL74\Desktop\1bm21cs003>java threadprg
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting: Thread[cse,5,main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting: Thread[bms,5,main]

C:\Users\BMSCECSEIL74\Desktop\1bm21cs003>
```

8) 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.

Q)

Packages

Lab program - 8

Create a packages 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 internals marks scored in five cources 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 cources 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 cources.

A)

```
public CIE ;
import java.util*;
public class student.
        {
            scanner sc = new Scanner ( system.in)
            public string usN, name;
            public int sem;
            public void getdata()
                {
                System.out.println ("enter USN, name
                                        and sem");
                usn = sc.nextline()
                name= sc.nextline();
                sem = sc.nextInt();
        }
```

```
public void disp()
{
    System.out.println("Name, "+name);
    System.out.println("USN "+USN);
    System.out.println("SEM "+sem);
}
}
import java.util.*;
public class internal extend student
{
    Scanner sc= new Scanner(Sys
    public int imarks[] = new int[
    public void getdata()
    { super.getdata();
      System.out.println(" enter mar
                          of SCIE");
      for(int i=0, i<5, i++)
      {
          imarks[i] = sc.nextInt()
      }
}
public void disp()
{
    System.out.print("CIE mark
    for(int i=0, i<5, i++)
    {
    System.out.println(imarks[i]+
     System.out.println()
}
}
```

```
package SEE ;
import CIE.internal
import java.util.*;
public class external ~~int~~ extends internal
    {
        Scanner sc = new Scanner (system.in);
        public int emarks[] = new int [5]
            public void getdata()
                {
                    super.getdata()
                System.out.println (" Enter 5
                subject marks for see ");
                for (int i=0 ; i<5 ; i++)
                    {
                    } emarks [i] = sc.nextInt();
            }
        public void disp()
            {
                system.out.print ("See marks
                for (int i=0 ; i<5, i++)
                {
                System.~~print~~out.print
                    (emarks [i] + " ");
                }
                system.out.println()
            }
    }
```

```
import CIE.student;
import CIE.internal;
import SEE.external;
import java.util.*;
class exam
{
    public static void main(String args[])
    {
    Scanner sc = new Scanner
                    (System.in);
    System.out.println("Enter number
                    of students");
        int n = sc.nextInt();
        int sum[] = new int[5]

    external e[] = new external[n];
     for (int i=0; i<n; i++)
      {
      e[i] = new external();
      e[i] = getdata();
    system.out.println("Final marks of
                    student "+(i+1)+"
                    ");
    for(int i=j=0, j<5, j++)
      {
        sum[i] = e[i].emarks[j]+
                    e[i].imarks[j];
} } } System.out.println(sum[i]+" ");
```

Output:

1
Enter student 1 details

Enter USN, name and SEM:

1BM21CS003
Abhinav
3
Enter CIE marks of 5 subjects:

40
70
50
60
80

Enter SEE marks of 5 subjects:

70
60
35
45
83

Final marks of student 1 of student in CIE:
300.