

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**  
“JnanaSangama”, Belgaum -590014, Karnataka.



**LAB REPORT**  
on

**OBJECT ORIENTED JAVA PROGRAMMING  
LAB**

**Submitted by**

**DEEPAYAN DAS  
(1BM21CS049)**

in partial fulfillment for the award of the degree of  
**BACHELOR OF ENGINEERING**  
in  
**COMPUTER SCIENCE AND ENGINEERING**



**B.M.S. COLLEGE OF ENGINEERING**  
(Autonomous Institution under VTU)  
**BENGALURU-560019**  
**October-2022 to Feb-2023**

**B. M. S. College of Engineering,  
Bull Temple Road, Bangalore 560019  
(Affiliated To Visvesvaraya Technological University, Belgaum)  
Department of Computer Science and Engineering**



**CERTIFICATE**

This is to certify that the Lab work entitled "Object Oriented Java Programming Lab (22CS3PCOOJ)" carried out by DEEPAYAN DAS (1BM21CS049), who is bonafide student of B. M. S. College of Engineering. It is in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belgaum during the year 2022. The Lab report has been approved as it satisfies the academic requirements in respect of a Database Management Systems (22CS3PCDBM) work prescribed for the said degree.

**DR.NandhiniVineethDr.Jyothi S Nayak**

Professor and Head

Department of CSE  
BMSCE, Bengaluru

Department of CSE  
BMSCE, Bengaluru

### Lab Program 1:

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

Code:

```
import java.util.Scanner;
import java.lang.Math;
class qe
{
    public static void main(String xx[])
    {
        Scanner s=new Scanner (System.in);
        double a,b,c,d,r1,r2;
        System.out.println("Enter values of a,b and c");
        a=s.nextDouble();
        b=s.nextDouble();
        c=s.nextDouble();
        if(a==0)
        { System.out.println("a shouldn't be 0");}
        else
        {
            d=(b*b)-(4*a*c);
            if(d>0)
            {
                System.out.println("Roots are real and distinct");
                r1= (-b+Math.sqrt(d))/(2*a);
                r2= (-b-Math.sqrt(d))/(2*a);
            }
        }
    }
}
```

```
System.out.print("R1:"+r1+" R2:"+r2);
}

else if(d==0)
{
    System.out.println("Roots are real and equal");
    r1=(-b)/(2*a);

    System.out.println("Roots are "+r1);
}

else
{
    System.out.println("Roots are imaginary");
    r1=(-b)/(2*a);

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

    System.out.println("R1:"+r1+"+i"+r2+" R2:"+r1+"-i"+r2);
}

}
```

The screenshot shows a Notepad window titled "Lab1.java - Notepad". The code is a Java program that calculates the roots of a quadratic equation. It uses Scanner to read input values for a, b, and c. It then calculates the discriminant d = b\*b - 4\*a\*c. Based on the value of d, it prints different messages: "a shouldn't be 0", "Roots are real and distinct", "Roots are real and equal", or "Roots are imaginary". It also prints the roots r1 and r2.

```
import java.util.Scanner;
import java.lang.Math;
class qe
{
    public static void main(String xx[])
    {
        Scanner s=new Scanner (System.in);
        double a,b,c,d,r1,r2;
        System.out.println("Enter values of a,b and c");
        a=s.nextDouble();
        b=s.nextDouble();
        c=s.nextDouble();
        if(a==0)
        { System.out.println("a shouldn't be 0");}
        else
        {
            d=(b*b)-(4*a*c);
            if(d>0)
            {
                System.out.println("Roots are real and distinct");
                r1= (-b+Math.sqrt(d))/(2*a);
                r2= (-b-Math.sqrt(d))/(2*a);
                System.out.print("R1:"+r1+" R2:"+r2);
            }
            else if(d==0)
            {
                System.out.println("Roots are real and equal");
                r1=(-b)/(2*a);
                System.out.println("Roots are "+r1);
            }
            else
            { System.out.println("Roots are imaginary");
                r1=(-b)/(2*a);
                r2=Math.sqrt(Math.abs(d))/(2*a);
                System.out.println("R1:"+r1+"+"i"+r2+" R2:"+r1+"-"+i"+r2);
            }
        }
    }
}
```

## Outputs:

The screenshot shows a Windows Command Prompt window. The user navigates to the directory "C:\Users\bmsce\Desktop\1BM21CS047\week-1". They compile the Java program with "javac qe.java" and then run it with "java qe". For each run, they enter values for a, b, and c. The program outputs the roots based on the discriminant's value.

```
Microsoft Windows [Version 10.0.19045.2251]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bmsce>cd C:\Users\bmsce\Desktop\1BM21CS047\week-1

C:\Users\bmsce\Desktop\1BM21CS047\week-1>javac qe.java

C:\Users\bmsce\Desktop\1BM21CS047\week-1>java qe
Enter values of a,b and c
0 20 30
a shouldn't be 0

C:\Users\bmsce\Desktop\1BM21CS047\week-1>java qe
Enter values of a,b and c
1 2 1
Roots are real and equal
Roots are -1.0

C:\Users\bmsce\Desktop\1BM21CS047\week-1>java qe
Enter values of a,b and c
1 2 3
Roots are imaginary

C:\Users\bmsce\Desktop\1BM21CS047\week-1>java qe
Enter values of a,b and c
1 5 3
Roots are real and distinct
R1:-0.6972243622680054 R2:-4.302775637731995
C:\Users\bmsce\Desktop\1BM21CS047\week-1>
```

```
cmd Select Command Prompt
Microsoft Windows [Version 10.0.19045.2251]
Copyright Microsoft Corporation. All rights reserved.

C:\Users\bmsce>cd C:\Users\bmsce\Desktop\1BM21CS047\week-1

C:\Users\bmsce\Desktop\1BM21CS047\week-1>javac qe.java

C:\Users\bmsce\Desktop\1BM21CS047\week-1>java qe
Enter values of a,b and c
*1 2 3
Roots are imaginary
R1:-1.0+i1.4142135623730951 R2:-1.0-i1.4142135623730951

C:\Users\bmsce\Desktop\1BM21CS047\week-1>
-M
-D
-C
-/C
-C
-M
-b
-t
-n
```

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

if ( $a \neq 0$ )

{  
if ( $d = 0$ )

{  
system.out.println ("Roots are real and

equal");

$$x_1 = (-b + \text{math.sqrt}(d, 0.5)) / (2 * a);$$

$$x_2 = (-b - \text{math.sqrt}(d, 0.5)) / (2 * a);$$

~~$x_1 =$~~  system.out.println ("Roots are

real and unequal");

$$(2 * a) x_1 = (-b + \text{math.sqrt}(d, 0.5)) / (2 * a)$$

$$x_2 = (-b - \text{math.sqrt}(d, 0.5)) / (2 * a);$$

system.out.println ("Root 1" + x\_1);

system.out.println ("Root 2" + x\_2);

else

{

system.out.println ("Roots are real and")

$a_1 = -b / (2 * a);$

$a_2 = \text{Math}\cdot\text{Pow}(-d, 0.5) / (2 * a);$

`System.out.println("root1" + a1 + " + " + d1 + a2);`

`System.out.println("root2" + a1 + " - " + d1 + a2);`

`} // methods of A9N3 below of both`

~~else~~ else

{

`System.out.println("wrong input");`

}

`public void`

`process2()`

`{`

`String`

`process2`

`(String)`

`)`

`= null`

real solutions to the quadratic equation

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

Code:

```
import java.util.Scanner;  
public class QuadraticEquation  
{  
    public static void main (String [] args)  
    {  
        Scanner s = new Scanner (System.in);  
        double a, b, c, d, r1, r2;  
        System.out.println ("Enter variables a, b, c  
respectively");  
        a = s.nextDouble();  
        b = s.nextDouble();  
        c = s.nextDouble();  
        d = b * b - 4 * a * c;
```

$a = s.nextDouble();$

$b = s.nextDouble();$

$c = s.nextDouble();$

Lab 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.

Code:

```
import java.util.Scanner;

class student
{
    student(){}
    String name;
    String usn;
    double result=0;
    int credits[]={};
    int marks[]={};
    int total=0;

    void accept()
    {
        Scanner s=new Scanner(System.in);

        System.out.println("Enter your Name ");
        name=s.nextLine();
        System.out.println("Enter USN ");
        usn=s.nextLine();
        System.out.println("Enter credits and marks of each subject respectively ");
        for(int i=0;i<3;i++)
        {
            this.credits[i]=s.nextInt();
            this.marks[i]=s.nextInt();
        }
        result=(credits[0]*marks[0]+credits[1]*marks[1]+credits[2]*marks[2])/3;
    }

    void display()
    {
        System.out.println("Name : "+name);
        System.out.println("USN : "+usn);
        System.out.println("Credits : "+credits[0]+","+credits[1]+","+credits[2]);
        System.out.println("Marks : "+marks[0]+","+marks[1]+","+marks[2]);
        System.out.println("SGPA : "+result);
    }
}
```

```
this.marks[i]=s.nextInt();
}
}

void calculate()
{
for(int i=0;i<3;i++)
{
if(marks[i]>=90 && marks[i]<=100)
result+=10*credits[i];
if(marks[i]>=80 && marks[i]<90)
result+=9*credits[i];
if(marks[i]>=70 && marks[i]<80)
result+=8*credits[i];
if(marks[i]>=60 && marks[i]<70)
result+=7*credits[i];
if(marks[i]>=50 && marks[i]<60)
result+=6*credits[i];
if(marks[i]>=40 && marks[i]<50)
result+=5*credits[i];
else result+=0*credits[i];
}
for(int i=0;i<3;i++)
total+=credits[i];
result=result/total;
}

void display()
{
System.out.println("Name:"+name+" USN:"+usn);
System.out.println("credits Marks");
}
```

```

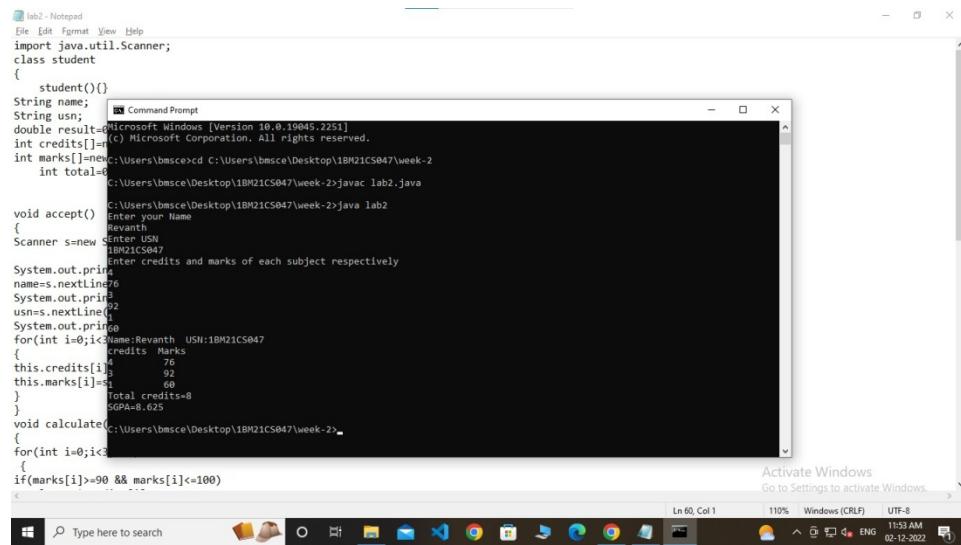
for(int i=0;i<3;i++)
System.out.println(credits[i]+      "+marks[i]);
System.out.println("Total credits="+total);
System.out.println("SGPA="+result);
}
}

```

```

class lab2
{
public static void main(String[] args)
{
    student s1=new student();
    s1.accept();
    s1.calculate();
    s1.display();
}
}

```



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.

code

```
import java.util.Scanner  
class student  
{  
    String name;  
    String USN;  
    double result = 0;  
    int credits[] = new
```

```
marks [] = new  
int [3];  
int total = 0;  
void accept()  
{  
    Scanner out.println("Enter your Name");  
    Name = s.nextLine();  
    System.out.println("Enter USN");  
    USN = s.nextLine();  
    System.out.print("Enter credits and marks of  
each subject respectively");  
    for(int i=0;i<3;i++)  
    {  
        this.credits[i] = s.nextInt();  
        this.marks[i] = s.nextInt();  
    }  
}  
void calculate()  
{
```

~~user = [ ]~~  
for (int i = 0; i < 3; i++)

{  
if (marks[i] >= 90 and marks[i] < 100)

result += 10 \* credits[i];

if (marks[i] >= 80 and marks[i] < 90)

result += 9 \* credits[i];

if (marks[i] >= 70 and marks[i] < 80)

result += 8 \* credits[i];

if (marks[i] >= 60 and marks[i] < 70)

result += 7 \* credits[i];

if (marks[i] >= 50 and marks[i] < 60)

result += 6 \* credits[i];

if (marks[i] >= 40 and marks[i] < 50)



```
System.out.println("credit  
marks");  
    }  
    F] { [i] tibers * 2 = + tibers  
for (int i = 0; i < 3, i++)  
System.out.println("Total  
credits = " + total);  
System.out.println("SGPA = " +  
result);  
}  
}  
class lab 2
```

```
{  
public static void main (String [] args)  
{  
Student S1 = new
```



student();  
s. I accept();  
s. I. calcul  
ate();  
s. I. displa  
y();  
{  
}  
{  
}

class  
carro  
{  
}  
{  
}

Lab 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.

Code:

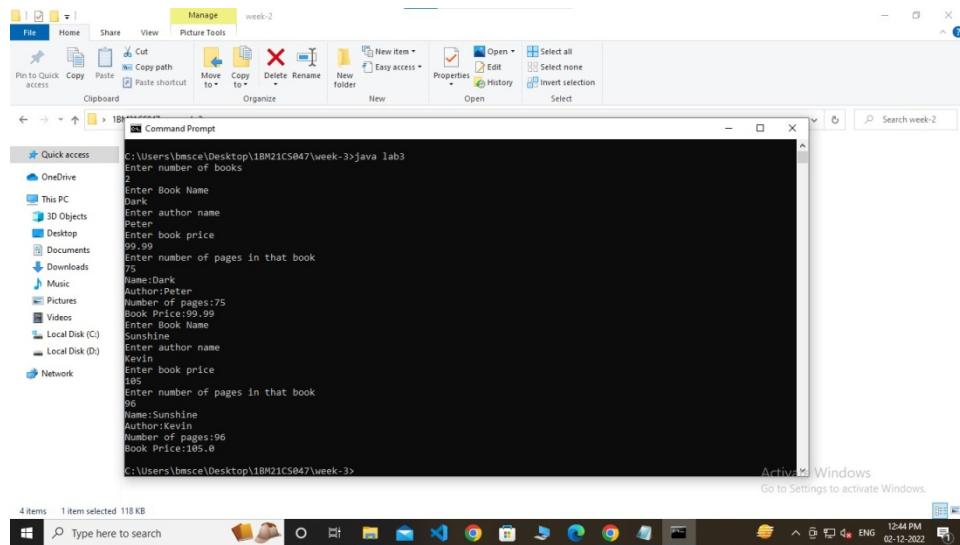
```
import java.util.Scanner;

class book
{
    String name,author;
    double price;
    int pages;
    book()
    {}
    Scanner s=new Scanner(System.in);
    void set()
    {
        System.out.println("Enter Book Name ");
        name=s.nextLine();
        System.out.println("Enter author name ");
        author=s.nextLine();
        System.out.println("Enter book price ");
        price=s.nextDouble();
        System.out.println("Enter number of pages in that book");
        pages=s.nextInt();
    }
}
```

```
public String toString()
{
    return("Name:"+name+"\nAuthor:"+author+"\nNumber of pages:"+pages+"\nBook Price:"+price);
}

}

class lab3
{
    public static void main(String xx[])
    {
        Scanner ss=new Scanner(System.in);
        int n;
        System.out.println("Enter number of books ");
        n=ss.nextInt();
        book b[]=new book[n];
        for(int i=0;i<n;i++)
        {
            b[i]=new book();
            b[i].set();
            System.out.println(b[i]);
        }
    }
}
```



```
double price;
int pages;
book()
{
    Scanner s = new Scanner (System.in);
    void set()
    {
        System.out.print("Enter book price");
        price = s.nextDouble();
        System.out.print("Enter number of pages in that book");
        pages = s.nextInt();
    }
}
```



public String toString()

```
{  
    return ("Name: " + name + " In Author: " +  
            author + " In Number of  
            pages: " + pages + " In Book Price " + price);  
}
```

class lab 3

```
{  
    public static void main(String xx[])
```

```
{  
    Scanner ss = new
```

```
Scanner (System.in); int n;
```

```
System.out.print("Enter")
```

Lab 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.

Code:

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

```
class book
```

```
{
```

```
String
```

```
name, author;}
```



`n = ss. nextInt();`

`book B [] = new`

`book [n];`

`for(int`

`i=0; i < n; i++)`

`{ book b = new`

`b[i] = new`

`b[i].set();`

`System.out.print`

`lin(b[i]);`

`}`

`}`

`}`

`[]`

Lab4. 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.

Code:

```
import java.util.Scanner;

abstract class shape
{
    shape(){}
    int h,b;
    abstract void printArea();
}

class rectangle extends shape
{
    Scanner s=new Scanner(System.in);
    void printArea()
    {
        System.out.println("Enter height and width of rectangle");
        h=s.nextInt();
        b=s.nextInt();
        System.out.println("Area of Rectangle is "+b*h);
    }
    rectangle(){}
}

class triangle extends shape
{
    Scanner s=new Scanner(System.in);
```

```
void printArea()
{
    System.out.println("Enter height and base of Triangle");
    h=s.nextInt();
    b=s.nextInt();
    System.out.println("Area of Triangle is "+0.5*b*h);
}
triangle(){}
}
```

```
class circle extends shape
{
    Scanner s=new Scanner(System.in);
    void printArea()
    {
        System.out.println("Enter radius of Circle");
        h=s.nextInt();
        System.out.println("Area of Circle is "+3.14*h*h);
    }
    circle(){}
}
```

```
class Lab4
{
    public static void main(String xx[])
    {
        rectangle r=new rectangle();
        r.printArea();
    }
}
```

```

triangle t=new triangle();
t.printArea();

circle c=new circle();
c.printArea();

}

}

```

The screenshot shows a Windows desktop environment. In the foreground, there is a 'Notepad' window titled 'Lab4 - Notepad' containing Java code. Below it is a 'Command Prompt' window showing the execution of the code and its output.

```

Lab4 - Notepad
File Edit Format View Help
Scanner s=new Scanner(System.in);
void printArea()
{
    Scanner s=new Scanner(System.in);
    void printArea()
    {
        Scanner s=new Scanner(System.in);
        void printArea()
        {
            Scanner s=new Scanner(System.in);
            void printArea()
            {
                Scanner s=new Scanner(System.in);
                void printArea()
                {
                    Scanner s=new Scanner(System.in);
                    void printArea()
                    {
                        Scanner s=new Scanner(System.in);
                        void printArea()
                        {
                            Scanner s=new Scanner(System.in);
                            void printArea()
                            {
                                Scanner s=new Scanner(System.in);
                                void printArea()
                                {
                                    Scanner s=new Scanner(System.in);
                                    void printArea()
                                    {
                                        Scanner s=new Scanner(System.in);
                                        void printArea()
                                        {
                                            Scanner s=new Scanner(System.in);
                                            void printArea()
                                            {
                                                Scanner s=new Scanner(System.in);
                                                void printArea()
                                                {
                                                    Scanner s=new Scanner(System.in);
                                                    void printArea()
                                                    {
                                                        Scanner s=new Scanner(System.in);
                                                        void printArea()
                                                        {
                                                            Scanner s=new Scanner(System.in);
                                                            void printArea()
                                                            {
                                                                Scanner s=new Scanner(System.in);
                                                                void printArea()
                                                                {
                                                                    Scanner s=new Scanner(System.in);
                                                                    void printArea()
                                                                    {
                                                                        Scanner s=new Scanner(System.in);
                                                                        void printArea()
                                                                        {
                                                                            Scanner s=new Scanner(System.in);
                                                                            void printArea()
                                                                            {
                                                                                Scanner s=new Scanner(System.in);
                                                                                void printArea()
                                                                                {
                                                                                    Scanner s=new Scanner(System.in);
                                                                                    void printArea()
                                                                                    {
                                                                                        Scanner s=new Scanner(System.in);
                                                                                        void printArea()
                                                                                        {
                                                                                            Scanner s=new Scanner(System.in);
                                                                                            void printArea()
                                                                                            {
                                                                                                Scanner s=new Scanner(System.in);
                                                                                                void printArea()
                                                                                                {
                                                                                                    Scanner s=new Scanner(System.in);
                                                                                                    void printArea()
                                                                                                    {
                                                                                                        Scanner s=new Scanner(System.in);
................................................................

```

The Command Prompt window shows the following interaction:

```

Microsoft Windows [Version 10.0.19045.2251]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bmsce>cd C:\Users\bmsce\Desktop\IBM21CS047\week-4>javac Lab4.java
C:\Users\bmsce\Desktop\IBM21CS047\week-4>java Lab4
Enter height and width of rectangle
10 20
Area of Rectangle is 200
Enter height and base of Triangle
10 20
Area of Triangle is 100.0
Enter radius of Circle
5
Area of Circle is 78.5
C:\Users\bmsce\Desktop\IBM21CS047\week-4>

class Lab4
{
public static
{
    rectangle r;
    triangle t;
    circle c;
    r.printArea();
    t.printArea();
    c.printArea();
}
}

```

The Java code defines three classes: Rectangle, Triangle, and Circle, each with a printArea() method. The Lab4 class contains references to these objects and calls their printArea() methods. The command prompt prompts the user for dimensions and prints the calculated areas.

## class Lab 4

8/30/14

```
{  
    public static void main (String xx [])
```

```
{  
    rectangle r = new rectangle (2, 2);  
    rectangle () ;  
    r. print . Area ();  
}
```

```
[ ]  
triangle t = new triangle (2, 2);  
triangle () ;  
t. print Area ();
```

circle c = new

circle ();

c. printArea ();

)

)

Shape

() { }

int :

n, b; abstract void printArea();

} class rectangle extends shape

{ Scanner s = new Scanner (System.in);  
void print Area ()  
{ System.out.printIn("Enter height and  
width of rectangle"); h = s.nextInt();  
b = s.nextInt();  
System.out.printIn("Area of Rectangle is  
" + b \* h);  
}



Lab 9 develop a Java program to create an abstract class named Shape that contains two integers and an empty method named

Print Area(). 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 print Area() that prints the area of the given shape.

Code:

```
import  
java.util.Scanner  
; abstract class  
Shape
```

rectangle () { }

}

class triangle <sup>extends shape</sup>

<sup>(triangle) with base & height</sup>

{

Scanner s = new Scanner(System.in);

[ ]

void

Print Area (

)

{

System.out.print("Enter height and  
base of triangle");

h = s.nextInt();

b = s.nextInt();

System.out.print("Area of Triangle  
is " + 0.5 \* b \* h);

}



triangle () { }

}

class circle extends shape

{ scanner s = new

scanner (System.in);

void print Area ()

{

System.out.printIn ("Enter radius")

of circle"); h = s.nextInt();

System.out.printIn ("Area of Circle is " + 3.14 \* h \* h);

t();

}

circle () {}

}

### Lab 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.

Complete the observation and execution of both the above programs tomorrow.

Code:

```
import java.util.Scanner;  
import java.lang.Math;  
  
class account  
{  
    String name=new String();  
    int accno;  
    double bal;  
    Scanner s=new Scanner(System.in);
```

```
void set()
{
    System.out.println("Enter customer name");
    name=s.nextLine();
    System.out.println("Enter "+name+"'s account number");
    accno=s.nextInt();
    System.out.println("Enter balance amount ");
    bal=s.nextDouble();
}

void display()
{
    System.out.println("Customer Name:"+name);
    System.out.println("Your account number:"+accno);
    System.out.println("Your Account Balance:"+bal);
}

account(){}  
}
```

```
class savacct extends account
{
    Scanner s=new Scanner(System.in);
    savacct()
    {
        System.out.println("Cheque Facility not available ");
    }
    void deposit()
    {
        int ch;
        double amt;
```

```

        System.out.println("Press 1 to deposit ");
        ch=s.nextInt();
        if(ch==1)
        {
            System.out.println("Enter amount to be deposited ");
            amt=s.nextDouble();
            bal=bal+amt;
        }
        else
            System.out.println("Invalid Input");
    }

    void in()
    {
        System.out.println("Enter rate of interest ");
        double r=s.nextDouble();
        r=r/100;
        System.out.println("Enter number of times interest applied per time period");
        int n=s.nextInt();
        System.out.println("Enter number of time periods");
        int t=s.nextInt();
        double x=(1+(r/n));
        double ci=bal*Math.pow(x,(n*t));
        System.out.println("Interest amount="+"(ci-bal)+" \nBalance amount without interest
is"+bal);
        bal=ci;
        System.out.println("Available balance after updating is"+bal);
    }

    void wd()
    {

```

```
System.out.println("Press 1 to withdraw amount");
int ch=s.nextInt();
if(ch==1)
{
    System.out.println("Enter the amount to be withdrawn ");
    double wdraw=s.nextDouble();
    if(wdraw<=bal)
    {
        bal=bal-wdraw;
        System.out.println("Available Balance:"+bal);
    }
}
else System.out.println("Invalid input");
}
```

```
class curacct extends account
{
    Scanner s=new Scanner(System.in);
    curacct()
    {
        System.out.println("Cheque Facility available ");
    }
    void deposit()
    {
        int ch;
        double amt;
        System.out.println("Press 1 to deposit ");
        ch=s.nextInt();
```

```

        if(ch==1)
        {
            System.out.println("Enter amount to be deposited ");
            amt=s.nextDouble();
            bal=bal+amt;
        }
        else
            System.out.println("Invalid Input");
    }

void wd()
{
    double wdraw;
    System.out.println("Press 1 to withdraw ammount");
    int ch=s.nextInt();
    if(ch==1)
    {
        System.out.println("Enter the amount to be withdrawn ");
        wdraw=s.nextDouble();
        bal=bal-wdraw;
        if(bal<1000)
        {
            System.out.println("You are running out of minimum balance \nAmount of rs 50
will be deducted as service charge for having low balance ");
            System.out.println("Do you want to continue with your transaction with
fine?\nPress 1 if yes ");
            int op=s.nextInt();
            if(op==1)
            {
                bal=bal-50;
            }
        }
    }
}

```

```
        System.out.println("Your Available Balance:"+bal);
    }
else
{
    System.out.println("your transaction is cancelled ");
    bal=bal+wdraw;
}
}
else System.out.println("Invalid input");
}

}
```

```
class Lab5
{
public static void main(String xx[])
{
    Scanner s=new Scanner(System.in);
    int ch;
    System.out.println("\n\nPress\n1. if your account is savings account \n2. if your account is current account");
    ch=s.nextInt();
    switch(ch)
    {
        case 1:
            savacct s1=new savacct();
            s1.set();
            s1.display();
    }
}
```

```

        s1.deposit();
        s1.in();
        s1.wd();
        break;

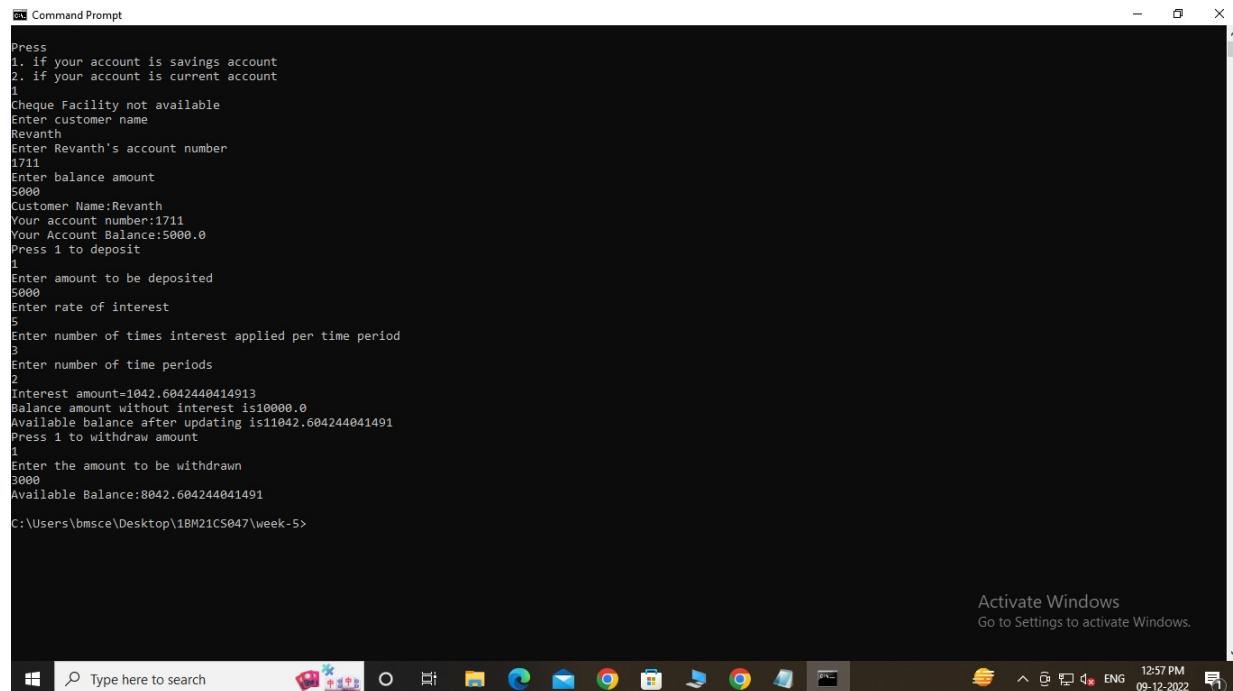
    case 2:
        curacct c1=new curacct();
        c1.set();
        c1.display();
        c1.deposit();
        c1.wd();
        break;

    default : System.exit(0);

}
}

}

```



```

Command Prompt
Press
1. if your account is savings account
2. if your account is current account
1
Cheque Facility not available
Enter customer name
Revanth
Enter Revanth's account number
1711
Enter balance amount
5000
Customer Name:Revanth
Your account number:1711
Your Account Balance:5000.0
Press 1 to deposit
1
Enter amount to be deposited
5000
Enter rate of interest
5
Enter number of times interest applied per time period
3
Enter number of time periods
2
Interest amount=1042.6042440414913
Balance amount without interest is10000.0
Available balance after updating is11042.604244041491
Press 1 to withdraw amount
1
Enter the amount to be withdrawn
3000
Available Balance:8042.604244041491
C:\Users\bmsce\Desktop\1BM21CS047\week-5>

```

Activate Windows  
Go to Settings to activate Windows.

```
 Command Prompt
Enter amount to be deposited
5000
Enter rate of interest
5
Enter number of times interest applied per time period
3
Enter number of time periods
2
Interest amount=1042.604244041491
Balance amount without interest is10000.0
Available balance after updating is11042.604244041491
Press i to withdraw amount
1
Enter the amount to be withdrawn
3000
Available Balance:8042.604244041491
C:\Users\bmsce\Desktop\1BM21CS047\week-5>java Lab5

Press
1. if your account is savings account
2. if your account is current account
2
Cheque Facility available
Enter customer name
Revanth
Enter Revanth's account number
1711
Enter balance amount
20000
Customer Name:Revanth
Your account number:1711
Your Account Balance:20000.0
Press 1 to deposit
1
Enter amount to be deposited
3000
Press 1 to withdraw ammount
1
Enter the amount to be withdrawn
20000
C:\Users\bmsce\Desktop\1BM21CS047\week-5>
```

Activate Windows  
Go to Settings to activate Windows.



Code

import

java . util . Scanner

2; import

`java.lang.Math;`

class account.

{

String

name = new

Strong(); int

accno ;

Double bar;

```
Scanner s = new Scanner(System.in);
```

[ ]

void set()

{

System. out. print. In C. Enter

*customer name "));*

```
    system.out.println("Press  
1 to deposit ");  
ch = s.nextInt();  
if (ch == 1)  
{  
    System.out.print("Enter amount  
to be deposited ");  
    amt = s.nextDouble();  
    float bal = bal + amt;  
    System.out.println("New balance is : " + bal);  
}  
else  
    System.out.println("Invalid Input");  
}
```



account() {}

class savacct extends account

{ Scanner s = new

scanner (System.in);

Savacct()

{

System. met. present In ('Cheque Facility  
not available "'),

3

void deposit()

{

int ch;

Double

amt';

System.out.println("Enter  
rate of interest "); double

r = s.nextDouble();

r = r / 100;

System.out.println("Enter number of  
times interest applied per time period "); int n = s.  
nextInt();

System.out.println("Enter number of time periods "); int t = s.  
nextInt();

double x = (1 + (r / n));

double ci = bal \* Math.pow(x, (n \* t));

System.out.println("Interest amount = "  
ci - bal) + "

" in Balance amount without interest

is " + bal);



## Lab 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

```
name = s.nextLine();  
System.out.printIn("Enter " + name + " account  
number"); accno = s.nextInt();
```

```
System.out.printIn("Enter  
balance amount ");
```

```
bal = s.nextDouble();
```

```
}
```

```
void display()
```

```
{
```

```
System.out.printIn("Customer  
Name: " + name);
```

```
System.out.printIn("Your Account  
number: " + accno);
```

```
System.out.printIn("Your  
Account Balance: " + bal);
```



to achieve the following tasks

- ① Accept deposit from customer and update the balance.
- ② Display the balance.
- ③ Compute and deposit interest.
- ④ Permit withdrawal and update the balance. Check for the minimum balance, impose penalty if necessary and update the balance.

complete the observation and see.

ition of both the above

programs tomorrow.

for first of office. we must  
abstain from precesses and schedule

class Lab5 {

{

public static void main(String xx[])

{

Scanner s = new

Scanner (System.in);

int ch;

System.out.print("1) Impresn1.

If your account is savings

account In 2 if your account is current  
account");

ch = s.next

Int();

switch (

begin

b1ch)

{

if (b1ch == '1')

Case 1 :

Savacct s1 = new



savacct();

s.l. set();

s.l. display();

s.l. deposit();

s.l. in();

s.l. w

d();

break

};

case 2:

curacct c.1 = new

curacct();

c1. set();

c1. display();

c1. deposit();

c.1. w

d();

break; system . exit(0);

default : system . exit(0);

System.out.println("Your Available Balance = " + bal);  
else  
{  
 withdraw = 2 \* amount;  
 bal = bal + withdraw;

}

System.out.println("Your transaction is cancelled");

bal = bal + withdraw;

}

tax = 2 \* amount;

}

amount = 0;

}

else amount = 0;

else System.out.println("Invalid input");

}

amount = 0;

}

amount = 0;

### Lab 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.

Program:

```
import java.util.Scanner;
class WrongAgeException extends Exception{
    public String toString(){
        return ("WrongAge!!! Age cannot be negative");
    }
}

class AgeException extends Exception{
    public String toString(){
        return("NotPossible!! Son's Age cannot be greater than Father's Age");
    }
}

class Father{
    int father_age;
    Father(int x) throws WrongAgeException{
        father_age=x;
        if(father_age<0){
            throw new WrongAgeException();
        }
    }
}

class Son extends Father{
    int son_age;
    Son(int x,int y) throws AgeException, WrongAgeException{
        super(x);
        son_age=y;
        if(son_age<0){
            throw new WrongAgeException();
        }
        if(son_age>=father_age){
            throw new AgeException();
        }
    }
}
```

```
}

class Lab6{
    public static void main(String xx[]) {
        try {
            Scanner s=new Scanner(System.in);
            System.out.println("Enter Son's age and Fathers age ");
            int y=s.nextInt();
            int x=s.nextInt();
            Son so=new Son(x,y);
            System.out.println("Father is " + so.father_age + " years old and son is
"+so.son_age + " years old");
        }
        catch (WrongAgeException wa) {
            System.out.println(wa);
        }
        catch (AgeException a){
            System.out.println(a);
        }
        catch (Exception e){
            System.out.println("Age is Interger value");
        }
    }
}
```

The screenshot shows a Windows desktop environment. In the foreground, a Notepad window titled "Lab6 - Notepad" displays Java code. Below it, a Command Prompt window shows the execution of the code and its output. The Taskbar at the bottom includes icons for File Explorer, Edge, Mail, and others, along with system status indicators.

```
Lab6 - Notepad
File Edit Format View Help
class Son extends Father{
    int son_age;
    Son(int x,int y) throws AgeException, WrongAgeException{
        super(x);
        son_age=y;
    }
}
class Lab6{
    public static void main(String[] args) {
        int father_age = 43;
        int son_age = 10;
        if (son_age > father_age) {
            System.out.println("NotPossible!! Son's Age cannot be greater than Father's Age");
        } else {
            System.out.println("Father is " + father_age + " years old and son is " + son_age + " years old");
        }
    }
}

Command Prompt
Microsoft Windows [Version 10.0.19045.2364]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bmsce>cd C:\Users\bmsce\Desktop\IBM21CS047\Week-6
C:\Users\bmsce\Desktop\IBM21CS047\Week-6>javac Lab6.java
C:\Users\bmsce\Desktop\IBM21CS047\Week-6>java Lab6
Enter Son's age and Father's age
43
10
NotPossible!! Son's Age cannot be greater than Father's Age
C:\Users\bmsce\Desktop\IBM21CS047\Week-6>
```

Lab G :

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 Wrong Age() 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  $\geq$  father's age.

Program :

```
import java.util.Scanner;  
class WrongAge extends Exception  
    extends Exception { public  
        String toString() {
```



return ("Wrong Age !!"); Age  
cannot be negative);  
but "positive" tells, only need a char)  
throws } no "balls only behavior  
Class Age Exception extends  
Exception { public String  
to String() {  
return ("Not possible !! Son's Age  
cannot be greater than  
Father's Age);  
};  
};  
class Father {



Wrong Age Exception

father\_age = x;

If (father\_age < 0) {

throw new WrongAgeException();

}

}

class Son extends

Father { int

Son\_age; } // son's age

Son (int x, int y) throws AgeException,

WrongAgeException { Super(x); }

Son\_age =

y;

If (Son\_age

```
    throw new WrongAgeException();
```

```
}
```

```
if (son_age >= father_age) {
```

```
    throw new AgeException();
```

```
}
```

```
}
```

```
□
```

```
}
```

```
class Lab 6 {
```

```
    public static void
```

```
    main (String xx []) {
```

```
        try {
```

```
            Scanner s = new Scanner (System.in);
```

```
            System.out.println ("Enter Son's age and
```

```
            Father's age");
```



```
    int x = s.nextInt();  
    int y = s.nextInt();  
    Son so = new Son(x, y);  
    System.out.printIn("Father is " + so.  
                      fatherAge + "  
Years old and son is " + so.sonAge + " years old");  
}  
catch (WrongAgeException  
      wa) {  
    System.out.printIn(wa)  
}  
}  
catch (AgeException a) {  
  System.out.printIn(a);  
}  
}  
catch (Exception e) {  
  System.out.printIn("Age is not Integer  
value");  
}
```

## LAB-8

Write a

program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once everytwo seconds.

Program:

```
class Call implements Runnable
{
    String a; int
    x,time;
    Thread t;
    Call(String tn,int ti,int ex)
    {
        a=tn;
        x=ex;
        time=ti;
        t=new Thread(this,a);
        t.start();
    }
    public void run()
    {
        try{
            for(int i=0;i<x      ;i++)
            {
                System.out.println(a);
                Thread.sleep(time);
            }
        }
        catch(InterruptedException ie)
        {
        }
    }
}
```

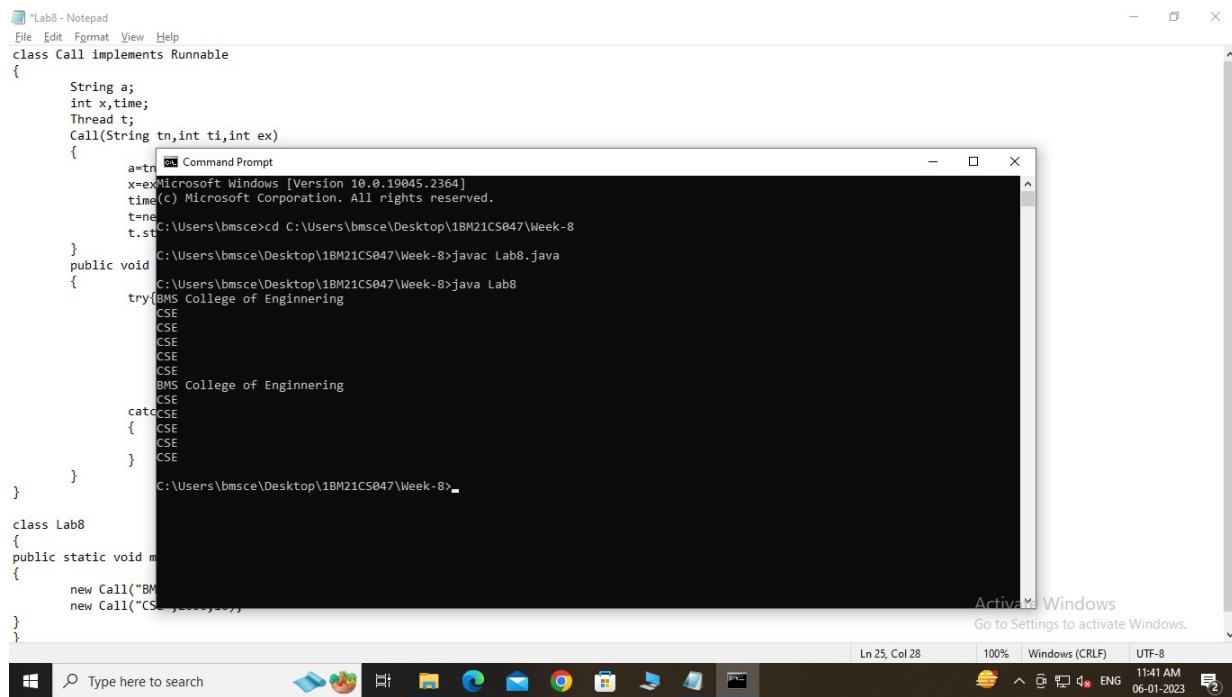
```

        System.out.println("Interrupted ");
    }
}

}

class Lab8
{
public static void main(String xx[])
{
    new Call("BMS College of Engineering",10000,2);
    new Call("CSE",2000,10);
}
}

```



The screenshot shows a Windows Command Prompt window titled "Lab8 - Notepad". The window displays the following Java application output:

```

a@n ~ Command Prompt
File Edit Format View Help
class Call implements Runnable
{
    String a;
    int x,time;
    Thread t;
    Call(String tn,int ti,int ex)
    {
        a=tn
        x=ex
        Microsoft Windows [Version 10.0.19045.2364]
        Copyright (c) Microsoft Corporation. All rights reserved.
        time=c
        t=new Thread(this);
        t.start();
    }
    public void run()
    {
        C:\Users\bmsce\Desktop\1BM21CS047\Week-8>javac Lab8.java
        C:\Users\bmsce\Desktop\1BM21CS047\Week-8>java Lab8
        try{BMS College of Enginnering
        CSE
        CSE
        CSE
        CSE
        CSE
        BMS College of Enginnering
        CSE
        CSE
        {
            CSE
            CSE
        }
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
}

class Lab8
{
public static void main(String xx[])
{
    new Call("BMS College of Engineering",10000,2);
    new Call("CSE",2000,10);
}
}

```

The application prints the names of two departments: "BMS College of Engineering" and "CSE", each repeated five times. The command prompt shows the path "C:\Users\bmsce\Desktop\1BM21CS047\Week-8>" and the file names "Lab8.java" and "Lab8". The status bar at the bottom indicates "Ln 25, Col 28", "100%", "Windows (CRLF)", "UTF-8", and the date/time "06-01-2023 11:41 AM".

Lab - 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.

Program:

```
class call implements Runnable
```

```
{
```

```
}
```

String

a; int time; do, make;

x, time;

Thread

t;

call (String tn, int li, int xc)

a = tn;

x = xc;

tim =



ti ;

t = new

Thread (fns, a);

t. start();

} public void reen()

{

□

for (int i = 0; i < x; i++)

{

□

System.out.println(a);

Thread.sleep(time);

{}

Catch (Interrupted Exception ie)

System.out.println("Interrupted");

↳ handle - events out and return

↳ interrupt events out. handled

↳ class Lab 8. now run it

↳ public static void main (String xx[])

↳ new call ("BMS College of

Engineering", 1000, 2); new method

call ("CSE", 2000, 10);

↳ where brief and careless first of

↳ narrow and no or (blank).

↳ where brief and careless first of

↳ narrow and no or (blank).

## Lab 9:

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:

```
package cie;
import java.util.Scanner;

public class student
{
    Scanner s= new Scanner(System.in);
    public String usn=new String(); public
    String name=new String(); public int
    sem;
    public student()
    {
        System.out.println("Enter USN, Name and Semester ");
        usn=s.nextLine();
        name=s.nextLine();
        sem=s.nextInt();
    }
}

package cie;
import java.util.Scanner;

public class internals extends cie.student
{
    Scanner s= new Scanner(System.in);public
    int marks[] =new int[5];
    public internals()
    {
        System.out.println("Enter marks of all 5 courses ");for(int
        i=0;i<5;i++)
        marks[i]=s.nextInt();
    }
}
```

package see;

```
import java.util.Scanner;
import cie.*;

public class externals extends cie.internals
{
    Scanner s= new Scanner(System.in);public
    int emarks[] =new int[5];
    public externals()
    {
        System.out.println("Enter marks of all 5 courses ");for(int
        i=0;i<5;i++)
        emarks[i]=s.nextInt();
    }
    public void display()
    {
        for(int i=0;i<5;i++)
        System.out.println("Course "+i+": "+(marks[i]+(0.5*emarks[i])));
    }
}

import java.util.Scanner;
import cie.*;
import see.*;

class main
{
    public static void main(String xx[])
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter number of students"); int n;
        n=s.nextInt();
        externals es[] =new externals[n];
        for(int i=0;i<n;i++)
        es[i]=new externals();
        for(int j=0;j<n;j++)
        {
            System.out.println("Student "+(j+1));
            for(int i=0;i<n;i++)
            es[i].display();
        }
    }
}
```

## Output:

```
Command Prompt
Enter number of students
2
Enter USN, Name and Semester
CS0475
Revanth
2
Enter marks of all 5 courses
40
46
42
48
37
Enter marks of all 5 courses
76
78
88
88
86
Enter USN, Name and Semester
CS0155
XYZ
3
Enter marks of all 5 courses
46
45
44
43
48
Enter marks of all 5 courses
88
86
74
66
37
Student 0
Course 0: 78.0
Course 1: 85.0
Course 2: 91.0
Course 3: 92.0
Course 4: 80.0
Course 0: 90.0
Course 1: 88.0
Course 2: 81.0
Course 3: 76.0
Course 4: 91.5
Student 1
Course 0: 78.0
Course 1: 85.0
Course 2: 91.0
Enter marks of all 5 courses
76
78
88
88
86
Enter USN, Name and Semester
CS0155
XYZ
3
Enter marks of all 5 courses
46
45
44
43
48
Enter marks of all 5 courses
88
86
74
66
37
Student 0
Course 0: 78.0
Course 1: 85.0
Course 2: 91.0
Course 3: 92.0
Course 4: 80.0
Course 0: 90.0
Course 1: 88.0
Course 2: 81.0
Course 3: 76.0
Course 4: 91.5
Student 1
Course 0: 78.0
Course 1: 85.0
Course 2: 91.0
Course 3: 92.0
Course 4: 80.0
Course 0: 90.0
Course 1: 88.0
Course 2: 81.0
Course 3: 76.0
Course 4: 91.5
```

## Lab - 9

Code:

```
package cie; // class name
```

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

```
public class student
```

```
{
```

```
Scanner s = new
```

```
Scanner (System.in);
```

```
public String
```

```
USN = new String();
```

```
public String
```

```
name = new String();
```

```
public int sem;
```

```
public student()
```

```
{
```

```
the student
```

```
System.out.print("Enter USN,
```

```
Name and Semester.");
```

```
USN = s.nextLine();
```



name = s. ne (did spelling  
at Line(3); this was not required)

sem = s. next

int(?)

}

ans = 2 sem(?)

}

(ans, ans(?) resonses)

package cie;

import java.util.Scanner;

public class internals extends cie

student

{

Scanner s = new Scanner(

Scanner(System.in));

public int

marks[] = new int[5];

public internals( ) {

{

```
System.out.println("Enter  
marks of all 5 courses");  
for (int i = 0; i < 5; i++)  
    marks[i] = s.nextInt();  
} // (Courses & No problem)  
++(i); } i.(0-5 times) of  
□ (i) for loop; s = [i] marks  
package see;  
import  
java.util.Scanner  
ner; import  
cie*;  
public class extends cie, implements  
{  
    Scanner s = new Scanner.in;  
    Scanner( System.in);  
    public int
```



```
emarks[] = new int [5];  
public external ()  
{  
    System.out.print("Enter  
marks of all 5 courses");  
    for (int i=0; i<5; i++)  
        emarks[i] = s.nextInt();  
}  
public void display()  
{  
    for (int i=0, i<5, i++)  
        System.out.print("Course  
" + i + " marks: " + emarks[i]);  
    System.out.println("Total marks: " +  
        0.5 * emarks[i]));
```

3

3

```
import java.util.Scanner;
import cie.*;
import see.*;

class main
{
    public static void main(String xx[])
    {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter
number of students");
        int n;
        n = s.nextInt();
        for (int i = 1; i <= n; i++)
        {
            System.out.print("Enter
marks of student " + i);
            int marks;
            marks = s.nextInt();
            if (marks > 100)
            {
                System.out.println("Marks
should not be greater than 100");
            }
            else
            {
                System.out.println("Your
marks are " + marks);
            }
        }
    }
}
```



externals

es [] = new

Externals [n];

for (int

i = 0; i < n; i++)

es[i] = new

externals ();

for (int

j = 0; j < n; j++)

j = 0; j < n; j++)

)

new = 2 numbers

{

System.out.println("

"Student" + (j+1));

for (int i = 0; i < n; i++)

es[i].display();

}