

# 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.

```
import java.util.Scanner;
class quad
{
    public static void main (String args[])
    {
        Scanner s = new Scanner (System.in);
        System.out.println ("The quadratic equation  
is  $a x^2 + b x + c = 0$ ");
        System.out.println ("Enter the value a");
        int a = s.nextInt();
        System.out.println ("Enter the value b");
        int b = s.nextInt();
        System.out.println ("Enter the value c");
        int c = s.nextInt();
        double d = Math.pow(b, 2) - (4 * a * c);
        int deno = 2 * a;
        double r1, r2;
        if (d >= 0)
        {
            System.out.println ("The roots are real  
and unequal");
            r1 = (-b + Math.sqrt(d)) / (deno);
            r2 = (-b - Math.sqrt(d)) / (deno);
            System.out.println ("The roots are  
" + r1 + " and " + r2);
        }
    }
}
```

else if ( $d == 0$ )  
{

System.out.println("The roots are  
real and equal");

$$r1 = (-b) / (deno);$$

$$r2 = (-b) / (deno);$$

System.out.println("The roots are"  
+ r1 + "and" +  
r2);

y

else

{

System.out.println("There are no  
real solutions for  
the equation");

f

f

A screenshot of a Java code editor window titled "quadratic.java". The code implements a quadratic equation solver. It uses a Scanner to read input values for coefficients a, b, and c. It calculates the discriminant d = b^2 - 4ac. Based on the value of d, it prints different messages and calculates the roots r1 and r2 using the quadratic formula. The code is written in Java syntax with standard imports.

```
import java.util.Scanner;
class quad
{
    public static void main(String args[])
    {
        Scanner s=new Scanner(System.in);
        System.out.println("THE QUADRATIC EXPRESSION IS ax^2+bx+c");
        System.out.println("ENTER THE VALUE OF a");
        int a=s.nextInt();
        System.out.println("ENTER THE VALUE OF b");
        int b=s.nextInt();
        System.out.println("ENTER THE VALUE OF c");
        int c=s.nextInt();
        double d=Math.pow(b,2)-(4*a*c);
        int deno=2*a;
        double r1,r2;
        if(d>0)
        {
            System.out.println("THE ROOTS ARE REAL AND UNEQUAL");
            r1=(-b+Math.sqrt(d))/(deno);
            r2=(-b-Math.sqrt(d))/(deno);
            System.out.println("the roots are "+r1+"and"+r2);
        }
        else if(d==0)
        {
            System.out.println("THE ROOTS ARE REAL AND EQUAL");
            r1=(-b)/(deno);
        }
        else
        {
            System.out.println("THERE ARE NO REAL SOLUTIONS FOR THE EQUATION");
        }
    }
}
```

A screenshot of a Java code editor window titled "quadratic.java". The code is identical to the one in the first screenshot, implementing a quadratic equation solver. It reads coefficients a, b, and c, calculates the discriminant d, and prints the roots r1 and r2 based on the value of d. The code uses standard Java syntax and imports.

```
System.out.println("ENTER THE VALUE OF c");
int c=s.nextInt();
double d=Math.pow(b,2)-(4*a*c);
int deno=2*a;
double r1,r2;
if(d>0)
{
    System.out.println("THE ROOTS ARE REAL AND UNEQUAL");
    r1=(-b+Math.sqrt(d))/(deno);
    r2=(-b-Math.sqrt(d))/(deno);
    System.out.println("the roots are "+r1+"and"+r2);
}
else if(d==0)
{
    System.out.println("THE ROOTS ARE REAL AND EQUAL");
    r1=(-b)/(deno);
    r2=(-b)/(deno);
    System.out.println("the roots are "+r1+"and"+r2);
}
else
{
    System.out.println("THERE ARE NO REAL SOLUTIONS FOR THE EQUATION");
}
```

```
c:\JAVA\bin>javac quad.java
javac: file not found: quad.java
Usage: javac <options> <source files>
use -help for a list of possible options

c:\JAVA\bin>javac quadratic.java

c:\JAVA\bin>java quad
THE QUADRATIC EXPRESSION IS ax^2+bx+c
ENTER THE VALUE OF a
1
ENTER THE VALUE OF b
1
ENTER THE VALUE OF c
1
THERE ARE NO REAL SOLUTIONS FOR THE EQUATION

c:\JAVA\bin>java quad
THE QUADRATIC EXPRESSION IS ax^2+bx+c
ENTER THE VALUE OF a
1
ENTER THE VALUE OF b
4
ENTER THE VALUE OF c
4
THE ROOTS ARE REAL AND EQUAL
the roots are -2.0and-2.0
```

## 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.

## Lab program-2

```
import java.util.Scanner;
class student
{
    private String usn;
    private String name;
    private int[] credit;
    private int[] marks;
    void getdetails()
    {
        System.out.println("Enter details of student");
        Scanner s = new Scanner(System.in);
        System.out.print("Enter name");
        name = s.nextLine();
        System.out.print("Enter usn");
        usn = s.nextLine();
        credit = new int[4];
        marks = new int[4];
        for (int i = 0; i < 4; i++)
        {
            System.out.print("Enter credit of subject " + (i + 1));
            credit[i] = s.nextInt();
        }
        for (int i = 0; i < 4; i++)
        {
            System.out.print("Enter mark of subject " + (i + 1));
            marks[i] = s.nextInt();
        }
    }
    void printdetails()
    {
    }
```

```
SOP("Usn of student is = " + usn);
```

```
SOP("Name of student is " + name);
```

```
for (int i = 0; i < 4; i++)
```

```
{  
    SOP("Mark of sub " + i + " is " + marks[i]);  
}
```

```
for (int i = 0; i < 4; i++)
```

```
{  
    SOP("Credit of subject " + i + " is " + credit[i]);  
}
```

```
Void calculate()
```

```
int [] total = new int [4];
```

```
double tc = 0.0;
```

```
double t = 0.0;
```

```
double sgpa = 0.0;
```

```
for (int i = 0; i < 4; i++)
```

```
{  
    If (marks[i] >= 85 && marks[i] <= 100)  
        total[i] = credit[i] + 10;
```

```
else if (marks[i] >= 70 && marks[i] < 85)  
    total[i] = credit[i] + 9;
```

```
else if (marks[i] >= 60 && marks[i] < 70)  
    total[i] = credit[i] + 8;
```

else if (Mark[i] >= 50 && Mark[i] < 60)

total[i] = credit[i] + 7;

else

total[i] = credit[i] + 6;

}

{

tc = tc + credit[i];

}

for (int i = 0; i < 4; i++)

{

t = t + total[i];

}

sgpa = (t / (tc + 0));

System.out.println("The sgpa of student is " + sgpa);

}

}

class student main

{ public static void main (String args[])

{

student st = new student();

st.getdetails();

st.printdetails();

st.calculate();

}

}

```

VA > bin > sgpajava > ...
import java.util.Scanner;
class Student
{
    private String usn;
    private String name;
    private int[] credit;
    private int[] marks;
    void getdetails()
    {
        System.out.println("ENTER THE DETAILS OF THE STUDENTS");
        Scanner s=new Scanner(System.in);
        System.out.println("ENTER NAME OF STUDENT");
        name=s.nextLine();
        System.out.println("ENTER THE USN OF STUDENT");
        usn=s.nextLine();
        credit=new int[4];
        marks=new int[4];
        for(int i =0;i<4;i++)
        {
            System.out.println("ENTER THE CREDIT OF SUBJECT"+(i+1));
            credit[i]=s.nextInt();
        }
        for(int i =0;i<4;i++)
        {
            System.out.println("ENTER THE MARK OF SUBJECT"+(i+1));
            marks[i]=s.nextInt();
        }
    }
    void printdetails()
    {
        System.out.println("THE USN OF STUDENT IS="+usn);
    }
}

```

File Run Terminal Help

sgpajava - Visual Studio Code

```

... login.html # login.css adminlogin.html JS login.js JS admin.js sgpajava X player.java ●
C: > JAVA > bin > sgpajava > ...
38     {
39         System.out.println(" THE CREDITS OF SUBJECT"+(i+1)+"="+credit[i]);
40     }
41     void calculate()
42     {
43         int[] total=new int[4];
44         double tc=0.0;
45         double t=0.0;
46         double sgpa=0.0;
47         for(int i=0;i<4;i++)
48         {
49             if(marks[i]>=85 && marks[i]<=100)
50                 total[i]=credit[i]*10;
51             else if(marks[i]>=70 && marks[i]<85)
52                 total[i]=credit[i]*9;
53             else if(marks[i]>=60 && marks[i]<70)
54                 total[i]=credit[i]*8;
55             else if(marks[i]>=50 && marks[i]<60)
56                 total[i]=credit[i]*7;
57             else
58                 total[i] =credit[i]*6;
59         }
60         for(int i=0;i<4;i++)
61         {
62             tc=tc+credit[i];
63         }
64         for(int i=0;i<4;i++)
65         {
66             t=t+total[i];
67         }
68         sgpa=(t/(tc+0.0));
69         System.out.println("THE SGPA OF STUDENT IS="+sgpa);
70     }
71 }
72 class studentmain

```

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder named "aved" containing files like login.html, login.css, adminlogin.html, login.js, admin.js, sgpa.java, and player.java.
- Code Editor:** Displays Java code for calculating SGPA. The code defines a class `studentmain` with a `main` method. It creates a `Student` object `s1`, gets details, prints them, and calculates the SGPA by summing four subjects and dividing by 4.0.

```
C:\>JAVA>bin> sgpa.java > ...
65     for(int i=0;i<4;i++)
66     {
67         t=t+total[i];
68     }
69     sgpa=(t/(tc*4.0));
70     System.out.println("THE SGPA OF STUDENT IS="+sgpa);
71 }
72 }
73 class studentmain
74 {
75     public static void main(String args[])
76     {
77         Student s1=new Student();
78         s1.getdetails();
79         s1.printdetails();
80         s1.calculate();
81     }
82 }
83
84
85
86
87
88
89
90
```

The screenshot shows a terminal window displaying the execution of the Java program and its output:

```
\$0
ENTER THE MARK OF SUBJECT3
90
ENTER THE MARK OF SUBJECT4
74
THE USN OF STUDENT IS=45
THE NAME OF STUDENT IS f
THE MARK OF SUBJECT1=54
THE MARK OF SUBJECT2=50
THE MARK OF SUBJECT3=90
THE MARK OF SUBJECT4=74
THE CREDITS OF SUBJECT1=5
THE CREDITS OF SUBJECT2=5
THE CREDITS OF SUBJECT3=5
THE CREDITS OF SUBJECT4=5
THE SGPA OF STUDENT IS=8.25

c:\JAVA\bin>
```

## 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.

Info about `toString()` method:

```
class Test
{
    int a=10;

    public String toString() // when an object is printed this method is automatically called
    { return("a=" + a); } //return statement with String is required
}
```

```
class TestMain
{
    public static void main (String ss[])
    {
        Test t1=new Test ();
        System.out.println (t1);
        //t1 - First example where an object is printed directly
        //calls toString method with the object given as the invoking object
    }
}
```

LAB 3 program

```
import java.util.Scanner;
class Book
{
    private String name;
    private String auth;
    private double p;
    private int np;

    Book()
    {
        name=null;
        auth=null;
        p=0.0;
        np=0;
    }

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

        System.out.println("Enter title of the book");
        name=s.nextLine();
        System.out.println("Enter author of the book");
        auth=s.nextLine();
        System.out.println("Enter price of book");
        p=s.nextDouble();
        System.out.println("Enter no of pages of book");
        np=s.nextInt();
    }
}
```

```
public String toString()
```

{

```
return ("The title of book: " + name + " if " + "Author of  
book: " + author + " if " + "Price of book: " + price  
+ " " + "No. of pages: " + np);
```

}

```
class Bookmain
```

{

```
public static void main (String args[])
```

{

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

```
SOP("Enter no. of books");
```

```
int n = sr.nextInt();
```

```
Book b[] = new Book[n];
```

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

{

```
SOP("Enter details of book " + (i + 1));
```

```
b[i] = new Book();
```

```
b[i].getdetails();
```

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

```
{ SOP("details of book " + (i + 1));
```

```
    SOP(b[i]);
```

}

Go Run Terminal Help

book.java - bin - Visual Studio Code

```
player.java book1.java bookJava employee.java age.java distance.java sgp.java
```

```
book.java > Book > getdetails()
1 import java.util.Scanner;
2 class Book
3 {
4     private String name;
5     private String auth;
6     private double p;
7     private int np;
8     Book()
9     {
10         name=null;
11         auth=null;
12         p=0.0;
13         np=0;
14     }
15     void getdetails()
16     {
17         Scanner s=new Scanner(System.in);
18         System.out.println("ENTER THE TITLE OF BOOK");
19         name=s.next();
20         System.out.println("ENTER AUTHOR OF BOOK");
21         auth=s.next();
22         System.out.println("ENTER PRICE OF BOOK");
23         p=s.nextDouble();
24         System.out.println("ENTER NUMBER OF PAGES OF BOOK");
25         np=s.nextInt();
26     }
27     public String toString()
28     {
29         return("THE TITLE OF BOOK:"+name+ " "+ "AUTHOR OF BOOK:"+auth+ " "+ " PRICE OF BOOK: "+p+ " "+ " NUMBER OF PAGES OF BOOK:"+np);
30     }
31 }
32 class Bookmain
33 {
    Run | Debug
    public static void main(String args[])
    {
        Scanner ss=new Scanner(System.in);
        System.out.println("ENTER THE NUMBER OF BOOKS");
    }
}
```

Ln 26, Col 6 Spaces: 4 UTF-8 CRLF Java JavaSE-14

INSAVED

```
book.java > Book > getdetails()
27     | public String toString()
28     {
29         return("THE TITLE OF BOOK:"+name+ " "+ "AUTHOR OF BOOK:"+auth+ " "+ " PRICE OF BOOK: "+p+ " "+ " NUMBER OF PAGES OF BOOK:"+np);
30     }
31 }
32 class Bookmain
33 {
    Run | Debug
    public static void main(String args[])
    {
        Scanner ss=new Scanner(System.in);
        System.out.println("ENTER THE NUMBER OF BOOKS");
        int n=ss.nextInt();
        Book b[]=new Book[n];
        for(int i=0;i<n;i++)
        {
            System.out.println("ENTER THE DETAILS OF BOOK"+(i+1));
            b[i]=new Book();
            b[i].getdetails();
        }
        for(int i=0;i<n;i++)
        {
            System.out.println("THE DETAILS OF BOOK"+(i+1));
            System.out.println(b[i]); /*SOP(b[i].toString()) will also work*/
        }
    }
}
```

```
c:\JAVA\bin>java Bookmain
ENTER THE NUMBER OF BOOKS
2
ENTER THE DETAILS OF BOOK1
ENTER THE TITLE OF BOOK
ad
ENTER AUTHOR OF BOOK
ed
ENTER PRICE OF BOOK
234
ENTER NUMBER OF PAGES OF BOOK
56
ENTER THE DETAILS OF BOOK2
ENTER THE TITLE OF BOOK
sdf
ENTER AUTHOR OF BOOK
ew
ENTER PRICE OF BOOK
234
ENTER NUMBER OF PAGES OF BOOK
45
THE DETAILS OF BOOK1
THE TITLE OF BOOK:ad AUTHOR OF BOOK:ed PRICE OF BOOK: 234.0 NUMBER OF PAGES OF BOOK:56
THE DETAILS OF BOOK2
THE TITLE OF BOOK:sdf AUTHOR OF BOOK:ew PRICE OF BOOK: 234.0 NUMBER OF PAGES OF BOOK:45
c:\JAVA\bin>
```

## LAB PROGRAM-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.

LAB 4 - program

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

```
abstract class Shape {
```

```
    double a;
```

```
    double b;
```

```
    Shape(double a, double b)
```

```
{
```

```
    this.a = a;
```

```
    this.b = b;
```

```
}
```

```
    abstract double area();
```

```
}
```

```
class Rectangle extends Shape
```

```
{
```

```
    Rectangle(double a, double b)
```

```
{
```

```
    super(a, b);
```

```
}
```

```
    double area()
```

```
{
```

```
    System.out.println("Area of Rectangle - ");
```

```
    return a * b;
```

```
}
```

```
}
```

EDGE

Class Triangle extends Shape

{  
Triangle (double a, double b)

{  
super (a, b);

double area()

{  
System.out.println ("Area of triangle");

return (a+b)/2;

}

Class Circle extends Shape {

final double pi = 3.14;

Circle (double a, double b) {

super (a, b); }

double area()

{  
System.out.println ("Area of Circle");

return pi \* Math.pow (a, 2);

}

}

class shapein

{

public static void main (String args[])

{

Scanner s = new Scanner (System.in);

sop ("Enter length & breadth of  
rectangle");

double l = s.nextDouble();

double b = s.nextDouble();

rectangle rec = new Rectangle (l, b);

sop ("Enter height and base of  
triangle");

double h = s.nextDouble();

double ba = s.nextDouble();

Triangle tri = new Triangle (h, ba);

sop ("Enter radius of circle");

double r = s.nextDouble();

Circle c = new Circle (r, 2 \* pi);

Shape sh;

sh = rec;

*alternate*

SOP (sh.area());  
sh = tri;  
SOP (sh.area());  
sh = c;  
SOP (sh.area());  
}

```
shapejava > ⚡ shapemain > ⏱ main(String[])
1 import java.util.Scanner;
2 abstract class Shape {
3     double a;
4     double b;
5     Shape(double a, double b)
6     {
7         this.a = a;
8         this.b = b;
9     }
10    abstract double area();
11 }
12
13 class Rectangle extends Shape
14 {
15     Rectangle(double a, double b)
16     {
17         super(a, b);
18     }
19     double area()
20     {
21         System.out.println("Area of Rectangle.");
22         return a*b;
23     }
24 }
```

```
# submitdetails.css      JS submitdetails.js      student1.html      Bank.java      shape.java      player.java
shape.java > shapemain > main(String[])
System.out.println("Area of Rectangle.");
return a*b;
}
}
class Triangle extends Shape {
Triangle(double a, double b) {
super(a, b);
}
double area() {
System.out.println(" Area of Triangle.");
return (a*b)/ 2;
}
}

class Circle extends Shape {
final double pi=3.14;
Circle(double a,double b) {
super(a, b);
}
double area() {
System.out.println(" Area of Circle.");
return pi*Math.pow(a,2);
}
}
```

Action View Go Run Terminal Help

shape.java - bin - Visual Studio Code

1 UNSAVED

```
... # submitdetails.css JS submitdetails.js student1.html Bank.java shape.java player.java
shape.java > shapemain > main(String[])
44 }
45 class shapemain
46 {
47     Run | Debug
48     public static void main [String args[]]
49     {
50         Scanner s=new Scanner(System.in);
51         System.out.println("ENTER LENGTH AND BREADTH OF RECTANGLE");
52         double l=s.nextDouble();
53         double b=s.nextDouble();
54         Rectangle rec = new Rectangle(l,b);
55         System.out.println("ENTER HEIGHT AND BASE LENGTH OF TRIANGLE");
56         double h=s.nextDouble();
57         double ba=s.nextDouble();
58         Triangle tri = new Triangle(h,ba);
59         System.out.println("ENTER RADIUS OF CIRCLE");
60         double r=s.nextDouble();
61         Circle c=new Circle(r,2*r);
62         Shape sh;
63         sh=rec;
64         System.out.println( sh.area());
65         sh=tri;
66         System.out.println( sh.area());
67         sh=c;
68         System.out.println( sh.area());
69     }
70 }
```

PROBLEMS 73 OUTPUT DEBUG CONSOLE TERMINAL

Filter (e.g. text, )

- shape.java 1
  - Resource leak: 's' is never closed Java[536871799] [49, 15]
- actor.java 2
  - Resource leak: 's' is never closed Java[536871799] [10, 17]
  - Resource leak: 'ss' is never closed Java[536871799] [35, 17]

```
shape.java > shapemain > main(String[])
60   Circle c=new Circle(r,2*r);
61   Shape sh;
62   sh=rec;
63   System.out.println( sh.area());
64   sh=tri;
65   System.out.println( sh.area());
66   sh=c;
67   System.out.println( sh.area());
68 }
69 }
```

```
4-exit
ENTER THE CHOICE
4
c:\JAVA\bin>java shapemain
ENTER LENGTH AND BREADTH OF RECTANGLE
12
12
ENTER HEIGHT AND BASE LENGTH OF TRIANGLE
2
4
ENTER RADIUS OF CIRCLE
5
Inside of Rectangle.
144.0
Area for Triangle.
4.0
Area for Circle.
78.5
c:\JAVA\bin>javac Bank.java
c:\JAVA\bin>java bank
1-CURRENT ACCOUNT
2-SAVINGS ACCOUNT
```

## LAB PROGRAM-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 Curr-acct and Sav-acct to

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

## Lab Program - 5

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

```
abstract class Account
```

```
{
```

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

```
String name;
```

```
long num;
```

```
String type;
```

```
double bal;
```

```
Account() { }
```

```
Account (String name, long num, String type,  
double bal)
```

```
{
```

```
this.name = name;
```

```
this.num = num;
```

```
this.type = type;
```

```
↳ this.bal = bal;
```

double Min = 2000.00;

abstract void deposit();

abstract void withdrawal();

abstract void display();

{

class Coract extends Account

{

Coract (String name, long num, String type,  
double bal)

{

super (name, num, type, bal);

{

Void withdrawal()

{

System.out.print ("Enter amount");

int amt = System.in.nextInt();

If (bal == 0 || amt > bal)

System.out.println ("Withdrawal not  
possible");

else

{

bal = bal - amt;

SOP ("Amount of " + amt + " is withdrawn");

SOP ("Rem balance = " + bal);

{

void deposit()

{

SOP ("Enter amount");

int amt = cin.get();

bal = bal + amt;

SOP ("Remaining balance of

account = " + bal);

{

void display()

{

if (bal < min)

{

SOP ("Amt of 145/- is  
deducted");

bal = bal - 145;

SOP ("Balance = " + bal);

{

else

SOP ("Balance = " + bal);

{

class Savact extends Account

{

Savact (String name, long num, String type,  
double bal)

{

super (name, num, type, bal);

}

void withdrawal()

{

SOP ("Enter amount");

int amt = sc.nextInt();

if (bal == 0 || amt > bal)

{

SOP ("Withdrawal not  
possible");

}

else

{

bal = bal - amt;

SOP ("Amount of " + amt + " is with-  
drawn from ac")

SOP ("Rem balance is " + bal);

}

```
void deposit()
```

{

```
SOP ("Enter amt to be deposited")
```

```
int amt = s.nextInt();
```

```
SOP ("The rate of interest is 5%");
```

```
double ci = amt * (100.05);
```

```
bal = bal + ci;
```

```
SOP ("The balance = " + bal);
```

}

```
void display()
```

{

```
SOP ("Balance = " + bal);
```

}

}

```
class bank
```

{

```
public static void main (String args [] )
```

{

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

```
SOP ("1 - Current account
```

```
2 - Savings account");
```

```
int c = s.nextInt();
```

```
String nam;
```

```
long n;
```

```
double openbal;
```

```
if (c == 1)  
{
```

```
SOP ("Enter name, acc no and  
opening balance");
```

```
nam = s.next();
```

```
n = s.nextLong();
```

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

```
Curact cur = new Curact(nam, n, openbal);
```

```
int i = 0;
```

```
while (i <= 4)
```

```
SOP (i: Deposit In 2: Display
```

```
In 3: Withdrawal 4: Exit);
```

```
SOP ("Enter choice");
```

```
int ch = s.nextInt();
```

```
switch(ch){
```

Case 1:

```
cu.deposit();
```

```
break;
```

Case 2:

```
cu.display();
```

```
break;
```

Case 3:

```
cu.withdrawal();
```

```
break;
```

Case 4:

```
System.exit(0);
```

```
break;
```

default:

```
System.out.println("Invalid choice");
```

g

t

p

```
else if (c == 2)
```

```
System.out.println("Enter name, ac no and opening  
balance");
```

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

```
ac = s.nextLong();
```

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

Savacct sa

= new Savacct (nam, n, bal, serivce,

openbal);

int j = 0;

while (j != 4)

{

SOP ("1 - Deposit In 2 - Display\n3 - Withdrawal 4 - Exit");

SOP ("Enter choice");

int ch = s.nextInt();

switch(ch){

case 1:

sa.deposit();

break;

case 2:

sa.display();

break;

case 3:

sa.withdrawal();

break;

case 4:

System.exit(0);

default: break;

SOP ("Invalid choice");

URBAN  
EDGE

```
    }  
    }  
    }  
else  
{  
    System.out.println("Invalid choice");  
}  
}  
}
```

```
Bank.java > 📁 bank > ⚒ main(String[])  
1 import java.util.Scanner;  
2 abstract class Account  
3 {  
4     Scanner s=new Scanner(System.in);  
5     String name;  
6     long num;  
7     String type;  
8     double bal;  
9     Account(){  
10    Account(String name,long num,String type,double bal)  
11    {  
12        this.name=name;  
13        this.num=num;  
14        this.type=type;  
15        this.bal=bal;  
16    }  
17    String acc()  
18    {  
19        return type;  
20    }  
21    double min=2000.00;  
22    abstract void deposit();  
23    abstract void withdrawal();  
24    abstract void display();
```

```

24    abstract void display();
25 }
26 class Curact extends Account
27 {
28     Curact(String name,long num,String type,double bal)
29     {
30         super( name ,num, type, bal);
31     }
32     void withdrawal()
33     {
34         System.out.println("ENTER THE AMOUNT TO BE WITHDRAWED");
35         int amt=s.nextInt();
36         if(bal==0 || amt>bal )
37         {
38             System.out.println("WITHDRAWAL NOT POSSIBLE");
39         }
40         else
41         {
42             bal=bal-amt;
43             System.out.println("AMOUNT OF"+amt+"IS WITHDRAWN FROM THE ACCOUNT");
44             System.out.println("REMAINING BALANCE IS="+bal);
45         }
46     }
47     void deposit()
48     {

```

```

submitdetails.css      JS submitdetails.js      student1.html      Bank.java X      shape.java ●      player.java      playerinheritance.java
● Bank.java > ⚡ bank > ⚡ main(String[])
46 }
47     void deposit()
48     {
49         System.out.println("ENTER THE AMOUNT TO BE DEPOSITED");
50         int amt1=s.nextInt();
51         bal=bal+amt1;
52         System.out.println("THE REMAINING BALANCE OF THE ACOOUNT= "+bal);
53     }
54     void display()
55     {
56         if (bal<min)
57         {
58             System.out.println("AMOUNT OF 145/- IS DEDUCTED FROM UR ACCOUNT DUE TO LESS BALANCE");
59             bal=bal-145;
60             System.out.println("BALANCE="+bal);
61         }
62         else
63             System.out.println("BALANCE="+bal);
64     }
65 }
66 class Savact extends Account
67 {
68     Savact (String name,long num,String type,double bal)
69     {

```

I

Filter (e.g. text, \*\*/\*.js, \*\*/node\_modules/)

```
Bank.java > 🐣 bank > ⚒ main(String[])
+
5     }
6
7     class Savact extends Account
8     {
9         Savact (String name,long num,String type,double bal)
10        {
11            super( name ,num, type, bal);
12        }
13        void withdrawal()
14        {
15            System.out.println("ENTER THE AMOUNT TO BE WITHDRAWED");
16            int amt=s.nextInt();
17            if(bal==0 || amt>bal)
18            {
19                System.out.println("WITHDRAWAL NOT POSSIBLE");
20            }
21            else
22            {
23                bal=bal-amt;
24                System.out.println("AMOUNT OF "+amt+" IS WITHDRAWN FROM THE ACCOUNT");
25                System.out.println("REMAINING BALANCE IS="+bal);
26            }
27        }
28        void deposit()
29        {
30
31
32
33
34
35
36
37
38 }
```

```
Bank.java > 🐣 bank > ⚒ main(String[])
+
1     void deposit()
2     {
3         System.out.println("ENTER THE AMOUNT TO BE DEPOSITED");
4         int amti=s.nextInt();
5         System.out.println("THE RATE OF INTEREST IS 5%");
6         double ci=amti*(1+0.05);
7         bal=bal+ci;
8         System.out.println("THE BALANCE OF THE ACCOUNT= "+bal);
9     }
10    void display()
11    {
12        System.out.println("BALANCE="+bal);
13    }
14
15
16 class bank
17 {
18     Run | Debug
19     public static void main(String args[])
20     {
21         Scanner s=new Scanner(System.in);
22         System.out.println("1-CURRENT ACCOUNT \n 2-SAVINGS ACCOUNT");
23         int c=s.nextInt();
24         String nam;
25         long n;
26         double openbal;
27         if(c==1)
28         {
29             System.out.println("ENTER NAME,ACCOUNT NUMBER AND OPENING BALANCE");
30             nam=s.next();
31             n=s.nextLong();
32             openbal=s.nextDouble();
33             System.out.println("ACCOUNT DETAILS \n *****");
34             System.out.println("NAME-"+ nam + " ACC NUMBER-"+n +"OPENING BALANCE-"+openbal+"TYPE-Current");
35             Curact cu=new Curact(nam,n,"current",openbal);
36             int i=0;
37             while( i != 4 )
38         }
39     }
40 }
```

```

Bank.java > ⚡ bank > ⏪ main(String[])
  21     int i=0;
  22     while( i != 4 )
  23     {
  24         System.out.println("1:DEPOSIT \n2:DISPLAY BALANCE\n 3:WITHDRAWAL \n 4-exit");
  25         System.out.println("ENTER THE CHOICE");
  26         int ch = s.nextInt();
  27         switch (ch) {
  28             case 1:
  29                 cu.deposit();
  30                 break;
  31
  32             case 2:
  33                 cu.display();
  34                 break;
  35
  36             case 3:
  37                 cu.withdrawal();
  38                 break;
  39             case 4:
  40                 System.exit(0);
  41                 break;
  42             default:
  43                 System.out.println("INVALID CHOICE");
  44         }
  45     }
  46
  47     else if(c==2){
  48         System.out.println("ENTER NAME,ACCOUNT NUMBER AND OPENING BALANCE");
  49         nam=s.next();
  50         n=s.nextLong();
  51         openbal=s.nextDouble();
  52         System.out.println("ACCOUNT DETAILS \n *****");
  53         System.out.println("NAME- " + nam + " ACC NUMBER-" +n + "OPENING BALANCE-"+openbal+"TYPE-Savings");
  54         Savact sa=new Savact(nam,n,"savings",openbal);
  55         int j=0;
  56         while( j != 4 )
  57         {
  58             System.out.println("1:DEPOSIT \n2:DISPLAY BALANCE\n 3:WITHDRAWAL \n 4-exit");
  59             System.out.println("ENTER THE CHOICE");
  60             int ch1 = s.nextInt();
  61             switch (ch1) {
  62                 case 1:
  63                     sa.deposit();
  64                     break;
  65
  66                 case 2:
  67                     sa.display();
  68                     break;
  69
  70                 case 3:
  71                     sa.withdrawal();
  72                     break;
  73                 case 4:
  74                     System.exit(0);
  75                     break;
  76                 default:
  77                     System.out.println("INVALID CHOICE");
  78             }
  79         }
  80     }
  81     else
  82     {
  83         System.out.println("INVALID CHOICE");
  84     }
  85 }

```

```
c:\JAVA\bin>javac Bank.java  
c:\JAVA\bin>java bank  
1-CURRENT ACCOUNT  
2-SAVINGS ACCOUNT  
1  
ENTER NAME,ACCOUNT NUMBER AND OPENING BALANCE  
nithin  
12344  
3000  
1:DEPOSIT  
2:DISPLAY BALANCE  
3:WITHDRAWAL  
4-exit  
ENTER THE CHOICE  
1  
ENTER THE AMOUNT TO BE DEPOSITED  
2000  
THE REMAINING BALANCE OF THE ACOOUNT= 5000.0  
1:DEPOSIT  
2:DISPLAY BALANCE  
3:WITHDRAWAL  
4-exit  
ENTER THE CHOICE  
3  
ENTER THE AMOUNT TO BE WITHDRAWED  
2000  
AMOUNT OF 2000 IS WITHDRAWN FROM THE ACCOUNT  
REMAINING BALANCE IS=3000.0  
1:DEPOSIT
```

```
> 3000  
1:DEPOSIT  
2:DISPLAY BALANCE  
3:WITHDRAWAL  
4-exit  
ENTER THE CHOICE  
3  
ENTER THE AMOUNT TO BE WITHDRAWED  
2000  
AMOUNT OF 2000 IS WITHDRAWN FROM THE ACCOUNT  
REMAINING BALANCE IS=1000.0  
1:DEPOSIT  
2:DISPLAY BALANCE  
3:WITHDRAWAL  
4-exit  
ENTER THE CHOICE  
2  
AMOUNT OF 145/- IS DEDUCTED FROM UR ACCOUNT DUE TO LESS BALANCE  
BALANCE=855.0  
1:DEPOSIT  
2:DISPLAY BALANCE  
3:WITHDRAWAL  
4-exit  
ENTER THE CHOICE  
4
```

