

Java program to find roots of quadratic equation

```
import java.util.Scanner;
import static java.lang.Math.*;
class Quadratic
{
    public static void main (String args[])
    {
        Scanner sc = new Scanner (System.in);
        System.out.print ("Enter the value of a:");
        float a = sc.nextFloat();
        System.out.print ("Enter the value of b:");
        float b = sc.nextFloat();
        System.out.print ("Enter the value of c:");
        float c = sc.nextFloat();
        if (a == 0)
        {
            System.out.print ("Invalid! a cannot be zero");
        }
        else if
        float d = b*b - 4*a*c;
        float sqrt_val = (float) Math.sqrt (abs(d));
        float root1 = (-b + sqrt_val) / (2*a);
        float root2 = (-b - sqrt_val) / (2*a);
        if (d == 0)
            System.out.print ("Roots are real and equal: " + root1);
        else if (d > 0)
        {
            System.out.print ("Roots are real and unique: ");
            System.out.print (root1 + "\n" + root2);
        }
        else
        {
            System.out.print ("Roots are imaginary");
            System.out.print (-b / (2*a) + " + i" + sqrt_val);
            System.out.print (-b / (2*a) + " + i" + sqrt_val);
        }
    }
}
```

Batch : 2

USN: IBM19CS137

Name: S Skanda

Algorithm

- 1) Get the input for \mathbb{R} values of a, b, c in expression ax^2+bx+c ;
- 2) Calculate the value of discriminant
 $d = \sqrt{b^2 - 4ac}$
- 3) Calculate the value of the roots
 $\text{root1} = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$ $\text{root2} = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$
- 4) If value of a is zero print invalid statement and exit.
- 5) If d is equal to zero print that the roots are real and equal and also print the roots
- 6) If d is greater than zero print the roots with a message saying the roots are real and unequal/distinct
- 7) If d is less than zero print the roots with a message saying that the roots are imaginary

Output:

Enter value of a : 1

Enter value of b : 2

Enter value of c : 1

The roots are real and equal: -1.0

Enter value of a : 0

Enter value of b : 2

Enter value of c : 5

Invalid Input.

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.

Algorithm :-

- 1) Define student class and declare variables name, usn, array of marks and credits, sgpa.
- 2) Define function/method to read details from the user
- 3) Define function/method display details to display the user details and sgpa.
- 4) Define function/method to calculate grade points for each subject and thus calculate the sgpa using following formula
$$sgpa = (\sum \text{grade points} * \text{credits}) / (\sum \text{credits});$$

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

```
class student {
```

```
    String usn, name;
```

```
    int credits[] = new int[6];
```

```
    int marks[] = new int[6];
```

```
    int i;
```

```
    float sgpa;
```

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

```
    void read-details()
```

```
    { System.out.print("Enter your Name : ");
```

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

```
      System.out.print("Enter your USN : ");
```

```
      usn = in.nextLine();
```

```
      System.out.println("Enter credits for each Sub:");
```

```
      for(i=0; i<6; i++)
```

```
      { System.out.print("Subject" + (i+1) + " : ");
```

```
        credits[i] = in.nextInt();
```

```
      }
```

```
      for(i=0; i<6; i++)
```

```
      { System.out.print("Subject Marks:");
```

```
        marks[i] = in.nextInt();
```

```
      }
```

```
    }
```

```
    void display-details()
```

```
    { System.out.println("Name : " + name);
```

```
      System.out.println("USN : " + usn);
```

```
      System.out.print("Credits ");
```

```
      for(i=0; i<6; i++)
```

```
      { System.out.print("Subject" + (i+1) + " : " + credits[i]);
```

```
      }
```

```
      System.out.println("Marks");
```



```

        for (i=0; i<6; i++)
        {
            System.out.println("Subject "+(i+1) +
                                "marks : " + marks[i]);
        }
        calc-sgpa();
        System.out.println("Sgpa:" + sgpa);
    }

```

```

void calc-sgpa()
{
    int sumgp=0, sumcred=0, gp;
    for (i=0; i<6; i++)
    {
        if (marks[i] >= 90) gp = 10;
        else if (marks[i] >= 80) gp = 9;
        else if (marks[i] >= 70) gp = 8;
        else if (marks[i] >= 60) gp = 7;
        else if (marks[i] >= 50) gp = 6;
        else gp = 5;
        sumgp = sumgp + (gp * credits[i]);
        sumcred = sumcred + credits[i];
    }
    sgpa = (sumgp / sumcred);
}

```

```

}

class Student-main {
    public static void main (String args[])
    {
        student s = new student();
        s.read-details();
        s.display-details();
    }
}

```

```

}

```

Expected Output:

Enter your name : QWERTY

Enter your USN : 123

Credits for each subject:

Subject 1 : 2

Subject 2 : 4

Subject 3 : 4

Subject 4 : 4

Subject 5 : 4

Subject 6 : 5

Marks for each subject:

Subject 1 : 45

Subject 2 : 56

Subject 3 : 67

Subject 4 : 78

Subject 5 : 89

Subject 6 : 90

Name : QWERTY

USN : 123

~~Subject~~

Credits:

Subject 1 : 2

Subject 2 : 4

Subject 3 : 4

Subject 4 : 4

Subject 5 : 4

Subject 6 : 5

Marks : Subject 1 : 45

Subject 2 : 56

Subject 3 : 67

Subject 4 : 78

Subject 5 : 89

Subject 6 : 90

SgPA : 7.0

Lab - 3

Name: S Skanda

USN: 1BM19CS137

Batch 2

Create a class book which contains four members

: name, author, price and num pages.

Include a constructor to set values for the members.

Include a toString() method that could display the complete details of the book.

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

```
class book {
```

```
    String title, author;
```

```
    int no. of pages;
```

```
    float price;
```

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

```
    book ()
```

```
    { System.out.print("Title:");
```

```
      title = in.nextLine();
```

```
      System.out.print("Author:");
```

```
      author = in.nextLine();
```

```
      System.out.print("Number of pages:");
```

```
      no. of pages = in.nextInt();
```

```
      System.out.print("Price:");
```

```
      price = in.nextFloat();
```

```
    }
```

```
    public String toString()
```

```
    { return ("In Book title: " + title + " \t Author' +
```

```
      author + " \t Number of pages: " + no. of pages
```

```
      + " \t Price: " + price);
```

```
    }
```

```
}
```

class books {

public static void main (String[] args)

{ book[] b = new book[3];

int i;

for(i=0; i<3; i++)

{ System.out.println("Enter book" + (i+1) + " details:");

b[i] = new book();

}

for(i=0; i<3; i++)

{ System.out.println("The book" + (i+1) + " details:");

System.out.println(b[i]);

}

}

}

Output

Enter book 1 details

Title: q

Author: a

Number of pages: 45

Price: 56

Enter book 2 details

Title: w

Author: s

Number of pages: 23

Price: 12

The book 1 details:

Title: q Author: a Number of pages: 45 Price: 56

The book 2 details:

Title: w Author: s Number of pages: 23 Price: 12

Lab - 4

Name S Skande

USN : IBM19CS137

Batch - 2

To create abstract class shape with two integer members. and create three class namely circle, rectangle and triangle to demonstrate ~~the~~ function overriding

```
abstract class shape {
```

```
    int dim1;
```

```
    int dim2;
```

```
    shape (int a, int b) {
```

```
        dim1 = a;
```

```
        dim2 = b;
```

```
    }
```

```
class Rectangle extends shape {
```

```
    Rectangle (int a, int b) {
```

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

```
    }
```

```
    void area() {
```

```
        double area = dim1 * dim2;
```

```
        System.out.println ("Area of Rectangle is " + area);
```

```
    }
```

```
}
```

```
class Triangle extends shape {
```

```
    Triangle (int a, int b) {
```

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

```
    }
```

```
    void area() {
```

```
        double area = dim1 * dim2 / 2;
```

```
        System.out.println ("Area of Triangle is " + area);
```

```
    }
```

```
}
```

```
class Circle extends Shape {  
    Circle(int a, int b) {  
        super(a, b);  
    }  
    void area() {  
        double area = 3.14 * dim1 * dim1;  
        System.out.println("Area of circle is " + area);  
    }  
}
```

```
}  
class Shapes {  
    public static void main(String args[]) {  
        Rectangle r = new Rectangle(7, 3);  
        Triangle t = new Triangle(6, 8);  
        Circle c = new Circle(5, 5);  
        Shape s;  
        s = r;  
        s.area();  
        s = t;  
        s.area();  
        s = c;  
        s.area();  
    }  
}
```

Output

Area of rectangle is 21

Area of triangle is 24

Area of circle is 78.5

Lab-5

06-11-2022

Name S Skanda

USN: 18M19CS137

Batch 2

To create account class and two subclass
savings account and current account

```
class account {
```

```
    String name, acc-no, acc-type;
```

```
    double balance;
```

```
    account (String name, String acc-no, String acc-type) {
```

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

```
        this.acc-no = acc-no;
```

```
        this.acc-type = acc-type;
```

```
        balance = 0;
```

```
    }
```

```
    void deposit (double amt) {
```

```
        System.out.println ("Balance : " + balance);
```

```
        balance += amount;
```

```
        System.out.println ("Updated balance : " + balance);
```

```
    }
```

```
    void withdraw (double amt) {
```

```
        System.out.println ("Balance : " + balance);
```

```
        balance -= amount;
```

```
        System.out.println ("Updated balance : " + balance);
```

```
    }
```

```
}
```

```
class curr-acct extends account {
```

```
    curr-acct (String name, String acc-no, String acc-type) {
```

```
        super (name, acc-no, acc-type);
```

```
    }
```

```
    double service = 100;
```

```
    double min-bal = 3000;
```

```
    int charged = 0;
```

```
void check {
```

```
if (balance < min_bal && charged == 0) {
```

```
    balance -= service;
```

```
    charged = 1;
```

```
    System.out.println("Service + " deducted  
                        due to low balance");
```

```
if (charged == 1)
```

```
{ System.out.println("your balance is low to  
    avoid being fined again increase your balance");
```

```
}
```

```
}  
void disp bal() {
```

```
    check();
```

```
    System.out.println("your balance is " + balance);
```

```
}
```

```
}
```

```
class sav_acct extends account {
```

```
sav_acct (String name, String acc_no, String acc_type) {
```

```
    super (name, acc_no, acc_type);
```

```
}
```

```
int given = 0;
```

```
void interest() {
```

```
if (balance > 10000 && given == 0) {
```

```
    balance += 0.007 * balance;
```

```
    System.out.println("0.7% interest credited");
```

```
    given += 1;
```

```
}
```

```
if (balance > 100000 && given == 1) {
```

```
    balance += 0.005 * balance; given += 1;
```

```
    System.out.println("0.5% interest credited"); }
```

```
if (balance > 1000000 && given == 2) {
```

```
    balance += 0.002 * balance;
```

```
    System.out.println("0.2% interest credited");
```

```
    given += 1;
```

```
}
```

```
}
```



```
void disp-bal(){
```

```
interest();
```

```
System.out.println("Your balance is "+balance);
```

```
}
```

```
}
```

```
class bank{
```

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

```
{ sav-acct sav = new sav-acct('A', '1b', "Savings");
```

```
System.out.println("Savings account function:");
```

```
sav.deposit(11000);
```

```
sav.disp-bal();
```

```
sav.withdraw(5000);
```

```
curr-acct cur = new curr-acct('B', '2b', "Current");
```

```
System.out.println("Current account functions:");
```

```
cur.deposit(5000);
```

```
cur.withdraw(2500);
```

```
cur.disp-bal();
```

```
}
```

```
}
```

Output:

Savings account functions:

balance: 0.0

updated balance: 11000.0

~~your account has been cr.~~

0.7% interest credited

your balance is 11077.0

balance: 11077.0

updated balance: 6077.0

Current account functions:

balance: 0.0

updated balance: 5000.0

balance: 5000.0

updated balance: 2500

100.0 deducted due to low balance

your balance is too low to avoid being fined again
increase your balance