

9) Java program to find the roots of the quadratic equations.

$$(a \neq 0) (ab -) \text{ type } \text{ stamp } = 6 \text{ lines}$$

• 2 lines x 3 lines = 6 lines

import java.util.Scanner; class "Two" {

public class quadratic equation {

public static void main(String [] args) {

Scanner

int a, b, c;

double det, root1, root2;

Scanner myobj = new Scanner (System.in);

System.out.println ("Enter the value of a, b, c : ");

a = myobj.nextInt();

b = myobj.nextInt();

c = myobj.nextInt();

det = a * b * c;

if (det > 0)

{

root1 = (-b + Math.sqrt(det)) / (2 * a));

root2 = (-b - Math.sqrt(det)) / (2 * a));

System.out.println ("The equation has two real roots : " + root1 + " and " + root2);

}

else if (det == 0)

{

one real root = -b / (2 * a);

System.out.println ("The quadratic equation has one real root : " + root);

}

else

{

square root will be $b/2 + a$ (here we will neglect a)

$$\text{root } \alpha = \text{cmath.sqrt}(-\text{det})/(2 * a);$$

System.out.println("The equation has complex roots :
" + "root1" + " and " + "root2");

}

} {
} { (2pm []) prints two state setting $\sqrt{-1}$ and $\sqrt{2}$
YANES

1. d, 0 first

Output: (from 1.0 root, root, both values)

$$((a, b, c) \rightarrow \text{root} = \text{sqrt}(b * b - a * a))$$

; ("Enter the value of a":) \rightarrow 1.0, 0.0, 0.0

1 ((b) first, if $b^2 = 0$)

Enter the value of b: ((b) first, if $b^2 = 0$)

0 ((b) first, if $b^2 = 0$)

Enter the value of c: ((c) first, if $c^2 = a * b$)

0 ((c) first, if $c^2 = a * b$)

root are: -2.0, -6.0

; ((c) Enter (first) value of root1 + root2) = 1.0, 0.0

; ((a + b) (first) root1 + root2) = 0.0, 0.0

Input: Enter value of a: 1.0, 0.0, 0.0

1.0, 0.0, 0.0 "b, 0" (first + "c": 0.0, 0.0)

Enter the value of c:

-9 ((c) = 0.0) if $c^2 = a * b$

roots are equal and real: 1.0, 1.0, 0.0

((b + c)(1.0 - 0.0))

root with Enter the value of a: 1.0, 0.0, 0.0

1 ((b) first + "c": 0.0, 0.0, 0.0)

Enter the value of b:

-3 ((b) = 0.0) if $b^2 = 0$

C:\Windows\System32\cmd.exe x C:\Users\Divya\OneDrive\Desktop\java practice>java Q

C:\Users\Divya\OneDrive\Desktop\java practice>java Q
Divyashree K Raju

1BM23CS096

Enter the value of a: 1

Enter the value of b: 8

Enter the value of c: 12

Roots are real and distinct.

Root 1: -2.0

Root 2: -6.0

C:\Users\Divya\OneDrive\Desktop\java practice>java Q

Divyashree K Raju

1BM23CS096

Enter the value of a: -4

Enter the value of b: 12

Enter the value of c: -9

Root is real and repeated.

Root: 1.5

C:\Users\Divya\OneDrive\Desktop\java practice>java Q

Divyashree K Raju

1BM23CS096

Enter the value of a: 1

Enter the value of b: -3

Enter the value of c: 10

Roots are complex and imaginary.

Root 1: 1.5 + 2.7838821814150108i

Root 2: 1.5 - 2.7838821814150108i

Enter the value of $\sqrt{-1}$: $\{++i; \text{System.out.println}(\text{root})\}$
 1) Roots are imaginary and real (roots + aren't) possible.

8/10/24

- 10) 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.

import java.util.Scanner; $\{() \text{ full code is in } 29.00\}$

```
class Student {
    String usn, name;
    int[] credits, marks;
    void AcceptDetails() {
        Scanner Hello = new Scanner(System.in);
        System.out.print("Enter the usn : ");
        usn = Hello.nextLine();
        System.out.print("Enter the name : ");
        name = Hello.nextLine();
    }
}
```

```
    System.out.print("Enter the credits and marks in order for 4 subjects : ");
    credits = new int[4];
    marks = new int[4];
}
```

for

```
    credits = new int[4];
    marks = new int[4];
}
```

```
    System.out.println("Enter details of credits and marks in order for 4 subjects : ");
}
```

```

for(int i=0; i<4 ; i++) {
    System.out.println("Enter credits for subject " + 
        subjects[i+1] + " :");
}
    
```

credits[i] = hello.nextInt();

```

System.out.println("Enter the marks for subject " + 
    subjects[i+1] + " :");
marks[i] = hello.nextInt();
}
    
```

marks[i] = hello.nextInt();

```

}
    
```

void display()

```

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

```

for(int i=0; i<4 ; i++) {
    System.out.println("Student De Subject " + (i+1) + credits[i] +
        " " + marks[i]);
}
    
```

```

}
    
```

if (m1 < m2) { m1 = m2 }

if (m1 < m3) { m1 = m3 }

m1 = m1 + m2 + m3 / 3

double} (d) () } private) nicht hier viele viele

int total credits = 0; // Initialize total credits

double totalGradePoints = 0; // initial value

(" : for (int i = 0 ; i < 4 ; i ++) {
 () void fib . backtrace

Total Credits + = Credits [P-].

```
; ((p[i].grade * p[i].credits) / totalCredits);  
} return totalGradePoints / totalCredits;
```

9

~~deputy~~

int ~~f~~ gradePoints (int main)

if (marks >= 90) return 10;

else if (marks >= 80) return "Passed"; else return

else if (marks >= 70) return 8; priv: user rights

else if (marks >= 60) return 7; else return 4;

else if (marks >= 50) return b;

else if(marks >= 40) return 5;

else return 0; Section 1 of Miller 1993

18 is triploid rot strong & bold

Erected up rod across road

PP e tigridia 200 35.000 x 8000

```
System.out.println("Enter a positive int number");
```

100

public class SGPA () {

Flight tracks

19035 small '02A

SPVIST: ŠMOL

of streams exhibiting a tendency

08:24 am, e: 6.0 m & 10.0 m

```
public static void main (String args [ ]) {  
    student student = new student ();  
    student . accept details ();  
    System.out.println ("student details : ");  
    student . display ();  
    [ Styg Hiba] = + Hibix ] Info.
```

08.10.24

output :

enter usn : 1bm23cl096 R: 121400 (08 session) li 226

enter name : divya 8 nature (of <15 sec) : alto

enter details for credits and marks in order for 40 subjects

Enter marks for subject in second (Q2) = (exam) in table

enter marks for subject MENG 0 (012 = <P+R>1) if 4018

enter credits for subject 2 : 2

Enter marks for subject 2: 80
credit

~~entry marks~~ ^{useless} for subject 3: 3

Enter marks for Subject 3 : 90

enter credits for subject 4: A

entry marks for subject 4-89

~~40% marks for sub,ect 4-89~~

Student details :

ISSN: 1134-2343 096

name : Divya

Subject 1 Credits:2, marks :90

Subject 2 credit: 2, marks: 80

36A9

DATE

DATE:

PAGE:

Subject 3 credits: 3, marks : 99 from biov sitab aliya.

Subject 4 credits: 4 , marks : 89.

SGPA : 9.0

Yesterdays marks biov sitab aliya

11

11 overall avg 9.0

File

C:\Windows\System32\cmd.exe

Microsoft Windows [Version 10.0.22000.2538]

(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin\Desktop>javac Student.java

C:\Users\Admin\Desktop>java Student

enter USN: 1bm23cs096

enter name: Divya

enter details of credits and marks in order for 4 subjects:

enter credits for subjects1: 2

enter marks for subjects1: 90

enter credits for subjects2: 2

enter marks for subjects2: 80

enter credits for subjects3: 3

enter marks for subjects3: 99

enter credits for subjects4: 4

enter marks for subjects4: 89

studentDetails:

student details

usn:Divya

name:null

subject1credits:2,marks:90

subject2credits:2,marks:80

subject3credits:3,marks:99

subject4credits:4,marks:89

SGPA:9.02727272727273

Subject 3 credits: 3, marks : 99

Subject 4 credits: 4, marks : 89.

SGPA : 9.0

15/10/24.

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

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

```
class Book;
```

```
{
```

```
private String name;
```

```
private String Author;
```

```
private double price;
```

```
private int num_pages;
```

```
public Book (String name, String Author, double price,  
num_pages);
```

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

```
this.Author = Author;
```

```
this.price = price;
```

```
this.num_pages = num_pages;
```

(marked points)

~~public static void main()~~

P8: 29 cm, P: 21 cm, P: 13 cm

~~public void main(String[] args)~~

O.P.: A902

public String getName();

return name; // returning null means object don't have a name (if

set it's value will return string + Name which is a shared ref of max. one

object will be copied on top hand but it's shallow because first mem

reference will be copied with it's value. So if we change first mem

reference will be copied with it's value. So if we change first mem

return Author;

}

public double getPrice();

{

return price;

}

public int getNumPages();

{

return numPages;

}

public void setName(String name);

{

this.name = name;

}

public void setAuthor(String author)

{

```

    this.author = author; } // name = m tri
}
} // nullity error
: [ ] stock value = 2000 [ ] book
public void setPrice(double price) {
{
    this.price = price; } // type two mistake
}
: (" " num book ") // type two mistake
    // nullity error - val
public void setNumPages(int numPages) {
{
    // nullity error - val
    this.numPages = numPages; } // type two mistake
}
: (" " stock & num pages ") // type two mistake
public String toString() { // exist many file
{
}
}

```

```

Input file name, author, name) stock = [ ] book
return "Book Details : \n " + "Name : " + name + "\n " +
"Author : " + "\n " + "Price : " + num Pages : " + "
(" stock + num Pages + "\n " // type two mistake
}
    (int i; i < i; i++) {
}
}

```

```

}
    (stock = good book ) val
}
public class Main
{
}

```

```

(( )) public static void main(String[] args) {
}

```

```

Scanner scanner = new Scanner (System.in);
System.out.println("Enter the number of Books : ");
}

```

```

int n = scanner.nextInt(); scanner.nextLine();
scanner.nextLine();
Book[] books = new Book[n];
for (int i = 0; i < n; i++) {
    System.out.println("Book details :");
    System.out.print("Book Name :");
    String name = scanner.nextLine();
    System.out.print("Author's name :");
    String author = scanner.nextLine();
    System.out.print("Enter the Price :");
    double price = scanner.nextDouble();
    System.out.print("Enter the no. of pages :");
    int numPages = scanner.nextInt();
}

```

`Book[n] = newBook(name, author, price, no. of pages)`

`+ " " + name + " " + author + " " + price + " " + numPages`

`+ " " + name + " " + author + " " + price + " " + numPages`

`System.out.println("Details of all Books :");`

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

~~`{}`~~

`for (Book book : books)`

`{}`

`every value printed`

`(System.out.println(book.toString()));`

`}`

`new Book [] constructor used = parameter values`

`" " + name + " " + author + " " + price + " " + numPages`

~~`process`~~

Output:

Enter the number of books : 2

Enter the details for book 1.

Enter book name : Harry Potter

Enter author name : jR Rowling

Enter price : 400

Enter number of pages : 400

Enter the details for book 2

Enter the book name : Grandma stories

Enter the author name : Sudha Murthy

Enter price : 400

Enter number of pages : 300

Details of all books :

Divyashree - 1BM23CS096

Book Name : Harry Potter

Author : jR Rowling

Price : 400

Number of pages : 400

Book name : Grandma stories

Author : Sudha Murthy

Price : 400

Number of pages : 300

Output
seen

C:\Windows\System32\cmd.e X +

C:\Users\STUDENT\Desktop>java Main
Enter the number of books: 2

Enter details for book 1:

Enter book name: harry potter
Enter author name: jk rowling
Enter price: 400
Enter number of pages: 400

Enter details for book 2:

Enter book name: grandma stories
Enter author name: sudha murthy
Enter price: 500
Enter number of pages: 300

Details of all books:

Divyashree- 1BM23CS096
Book Name: harry potter
Author: jk rowling
Price: 400.0
Number of Pages: 400

Divyashree- 1BM23CS096
Book Name: grandma stories
Author: sudha murthy
Price: 500.0
Number of Pages: 300

C:\Users\STUDENT\Desktop>