DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

B M S COLLEGE OF ENGINEERING (AUTONOMOUS COLLEGE UNDER VTU, BELGAUM)

BANGALORE – 560019 2023-24

LAB REPORT OF OBJECT-ORIENTED JAVA PROGRAMMING (23CS3PCOOJ)

LAB REPORT

BY

NAME	USN
Divakar Babu M P	1BM22CS093

Course Instructor

Shravya. A. R., Assistant Professor,

Dept. of CSE, BMSCE

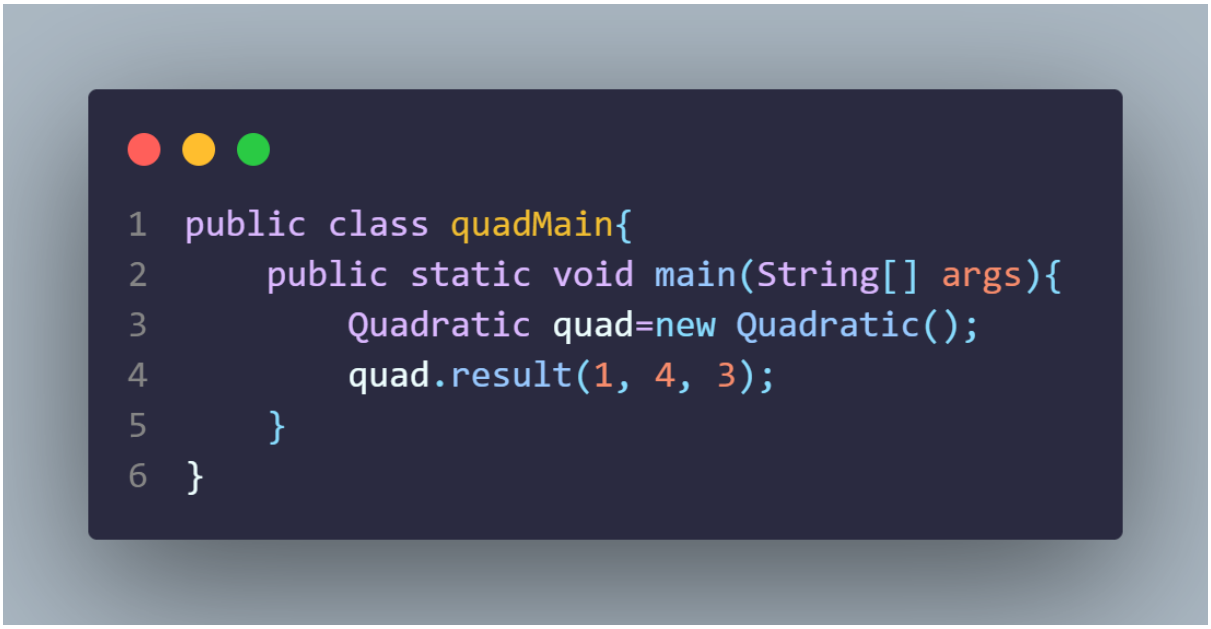
LAB PROGRAM 1:

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 discriminate b^2 -4ac is negative, display a message stating that there are no real solutions.

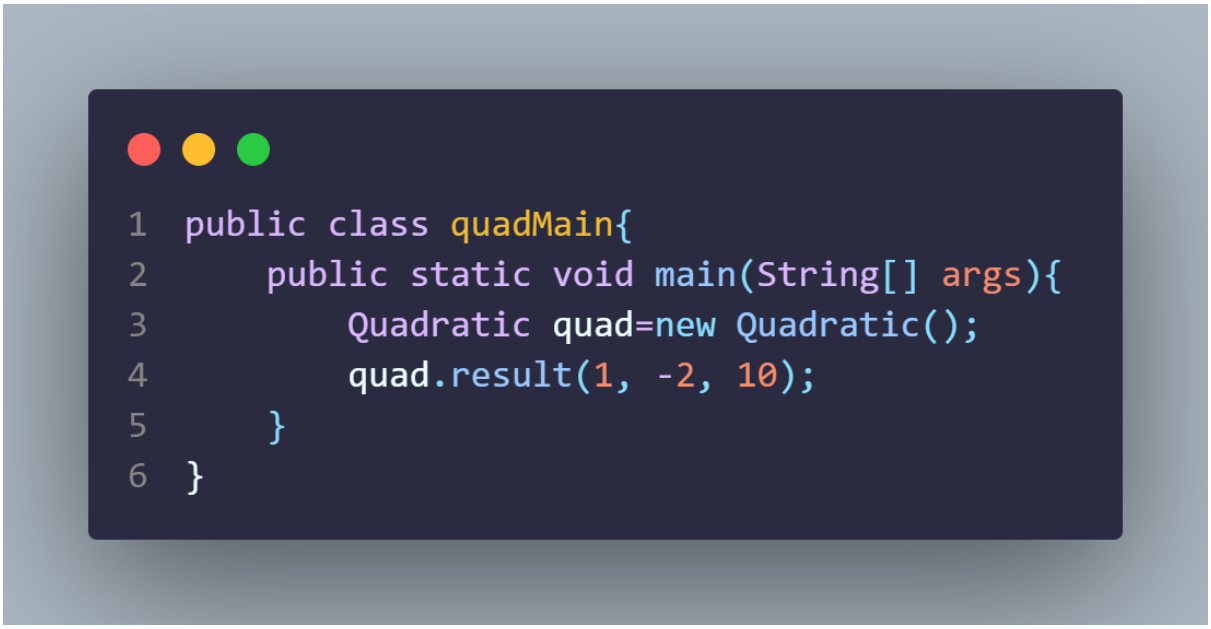
CODE:

```java
public class Quadratic {
    double a,b,c;
    double D;
    double res,res1;
    void result(double a,double b,double c){
        D=Math.pow(b, 2)-(4*a*c);
        try{
        if(D>0){
            res=(-b+Math.sqrt(D))/2*a;
            res1=(-b-Math.sqrt(D))/2*a;
            System.out.println("Result1:"+res);
            System.out.println("Result2:"+res1);
        }
        else if(D==0){
            res=-b/(2*a);
            System.out.println("Result:"+res);
        }
        else{
            System.out.println("No real solution Exist");
        }
    }catch(ArithmeticException e){
        System.out.println(e);
    }
    }
}

```



```java
public class quadMain{
    public static void main(String[] args){
        Quadratic quad=new Quadratic();
        quad.result(1, 4, 3);
    }
}
```



```java
public class quadMain{
    public static void main(String[] args){
        Quadratic quad=new Quadratic();
        quad.result(1, -2, 10);
    }
}
```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\newJava> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\newJava\" ; if ($?) { java
c quadMain.java } ; if ($?) { java quadMain }
Name:DivakarBabu M P
USN:1BM22CS093
Result1:-1.0
Result2:-3.0
```

OUTPUT2:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\newJava> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\newJava\" ; if ($?) { java
c quadMain.java } ; if ($?) { java quadMain }
Name:DivakarBabu M P
USN:1BM22CS093
No real solution Exist
```

LAB PROGRAM 2:

Develop a Java program to create a class Student with members usn, name, an array of credits and an array of marks. Include methods to accept and display details and a method to calculate SGPA of a student.

CODE:

```java
public class Student {
    String usn;
    String Name;
    int[] grade;
    int[] credit;

    public Student(String usn, String Name, int[] grade, int[] credit) {
        this.usn = usn;
        this.Name = Name;
        this.grade = grade;
        this.credit = credit;
    }

    public int marksConverter(int grade) {
        if (grade >= 90) {
            return 10;
        } else if (grade >= 80) {
            return 9;
        } else if (grade >= 70) {
            return 8;
        } else if (grade >= 60) {
            return 7;
        } else if (grade >= 50) {
            return 6;
        } else if (grade >= 40) {
            return 5;
        } else {
            return 0;
        }
    }

    public void displayInformation() {
        System.out.println("Student usn:" + usn);
        System.out.println("Student Name:" + Name);
        for (int i = 0; i < credit.length; i++) {
            System.out.printf("Grade in subject %d is %d%n", i + 1, grade[i]);
        }
    }

    public double calculateSgpa() {
        double Sgpa = 0.0;
        double totalCredits = 0.0;

        for (int i = 0; i < grade.length; i++) {
            totalCredits += credit[i];
            Sgpa += marksConverter(grade[i]) * credit[i];
        }

        if (totalCredits == 0) {
            return 0.0;
        }

        return Sgpa / totalCredits;
    }
}
```

```java
import java.util.Scanner;
public class Main{
    public static void main(String[] args){
        int[] grade=new int[5];
        int[] marks=new int[5];
        Scanner scanner=new Scanner(System.in);
        System.out.println("Enter the Usn of the student:");
        String Usn=scanner.nextLine();
        System.out.println("Enter the Name of the student:");
        String Name=scanner.nextLine();
        System.out.println("Enter the marks of each subject:");
        for(int i=0;i<grade.length;i++){
            grade[i]=scanner.nextInt();
        }
        System.out.println("Enter the credit of each subject:");
        for(int i=0;i<grade.length;i++){
            marks[i]=scanner.nextInt();
        }
        Student stud=new Student(Usn,Name,grade,marks);
        stud.displayInformation();
        System.out.println("Sgpa="+stud.calculateSgpa());
    }
}
```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\java23022024> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\java23022024\" ; if (
$?) { javac Main.java } ; if ($?) { java Main }
Name:DivakarBabu M P
USN:1BM22CS093
Enter the Usn of the student:
1CBU82PHY098
Enter the Name of the student:
Issac Newton
Enter the marks of each subject:
99
100
98
89
95
Enter the credit of each subject:
4
4
3
2
1
Student usn:1CBU82PHY098
Student Name:Issac Newton
Grade in subject 1 is 99
Grade in subject 2 is 100
Grade in subject 3 is 98
Grade in subject 4 is 89
Grade in subject 5 is 95
Sgpa=9.857142857142858
```

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.

CODE:

```java
public class Book {
    String Bookname;
    String author;
    int price;
    int num_pages;
    public Book(String Bookname,String author,double price2,int num_pages){
        this.Bookname=Bookname;
        this.author=author;
        this.price=price;
        this.num_pages=num_pages;
    }
    @Override
    public String toString(){
        return "Book Name="+Bookname+"\n"+
               "Author="+author+"\n"+
               "price="+price+"\n"+
               "Number of pages="+num_pages;
    }
}

```

```java
import java.util.Scanner;
public class Main{
    public static void main(String[] args){
        int[] grade=new int[5];
        int[] marks=new int[5];
        Scanner scanner=new Scanner(System.in);
        System.out.println("Enter the Usn of the student:");
        String Usn=scanner.nextLine();
        System.out.println("Enter the Name of the student:");
        String Name=scanner.nextLine();
        System.out.println("Enter the marks of each subject:");
        for(int i=0;i<grade.length;i++){
            grade[i]=scanner.nextInt();
        }
        System.out.println("Enter the credit of each subject:");
        for(int i=0;i<grade.length;i++){
            marks[i]=scanner.nextInt();
        }
        Student stud=new Student(Usn,Name,grade,marks);
        stud.displayInformation();
        System.out.println("Sgpa="+stud.calculateSgpa());
    }
}
```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\java23022024> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\java23022024\" ; if ( $?) { javac MainBook.java } ; if ($?) { java MainBook }
Name:DivakarBabu M P
USN:1BM22CS093
Enter the name of the book:
abc
Enter the name of the Author:
bca
Enter the price of the book:
100
Enter the pages of the book:
200
Enter the name of the book:
xyz
Enter the name of the Author:
zyx
Enter the price of the book:
150
Enter the pages of the book:
250
Enter the name of the book:
pqr
Enter the name of the Author:
rqp
Enter the price of the book:
230
Enter the pages of the book:
500
```

```
 Book 1 details:
 Book Name=abc
 Author=bca
 price=0
 Number of pages=200

 Book 2 details:
 Book Name=xyz
 Author=zyx
 price=0
 Number of pages=250

 Book 3 details:
 Book Name=pqr
 Author=rqp
 price=0
 Number of pages=500
```

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.

CODE:

```java
public class MainShape {
    public static void main(String args[]){
        Rectangle rect=new Rectangle();
        Circle cir=new Circle();
        Triangle tri=new Triangle();
        rect.printArea(10,20);
        cir.printArea(2, 0);
        tri.printArea(2,8);
    }
}

```

```java
public class Rectangle extends Shape{
    @Override
    void printArea(double x,double y){
        double area=x*y;
        System.out.println("area of reactangle:"+area);
    }
}

```

```java
public class Triangle extends Shape {
    @Override
    void printArea(double x,double y){
        double area=(x*y)/2;
        System.out.println("area of Triangle:"+area);
    }
}

```

```java
public class Circle extends Shape{
    @Override
    void printArea(double x,double y){
        double area=Math.PI*Math.pow(x, 2);
        System.out.println("area of circle:"+area);
    }
}

```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\java23022024> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\java23022024\" ; if (
$?) { javac MainShape.java } ; if ($?) { java MainShape }
Name:DivakarBabu M P
USN:1BM22CS093
area of reactangle:200.0
area of circle:12.566370614359172
area of Triangle:8.0
```

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 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.

CODE:

```java
class Account {
protected String customerName;
protected String accountNumber;
protected double balance;
public Account(String customerName, String accountNumber) {
this.customerName = customerName;
this.accountNumber = accountNumber;
this.balance = 0;
}
public void deposit(double amount) {
balance += amount;
System.out.println("Deposit of $" + amount + " successful");
}
public void displayBalance() {
System.out.println("Account Number: " + accountNumber + "\nBalance: " + balance);
}
}
class SavingsAccount extends Account {
public SavingsAccount(String customerName, String accountNumber) {
super(customerName, accountNumber);
}
public void computeInterest() {
double interestRate = 0.05;
double interest = balance * interestRate;
balance += interest;
System.out.println("Interest of $" + interest + " computed and added to the balance.");
}
public void withdraw(double amount) {
if (balance >= amount) {
balance -= amount;
System.out.println("Withdrawal of " + amount + " successful");
} else {
System.out.println("Insufficient funds for withdrawal.");

}
}
}
class CurrentAccount extends Account {
private double minimumBalance = 1000;
public CurrentAccount(String customerName, String accountNumber) {
super(customerName, accountNumber);
}
public void withdraw(double amount) {
if (balance - amount >= minimumBalance) {
balance -= amount;
System.out.println("Withdrawal of " + amount + " successful.");
} else {
System.out.println("Insufficient funds. Service charge applied.");
imposeServiceCharge();
}
}
private void imposeServiceCharge() {
double serviceCharge = 20;
balance -= serviceCharge;
System.out.println("Service charge of $" + serviceCharge + " imposed.");
}
}
class Bank {
public static void main(String XX[]) {
SavingsAccount savingsAccount = new SavingsAccount("John Doe", "SA1001");
CurrentAccount currentAccount = new CurrentAccount("Jane Smith","CA2002");
savingsAccount.deposit(5000);
savingsAccount.displayBalance();
savingsAccount.computeInterest();
savingsAccount.displayBalance();
savingsAccount.withdraw(2000);
savingsAccount.displayBalance();
currentAccount.deposit(8000);
currentAccount.displayBalance();

currentAccount.withdraw(5000);
currentAccount.displayBalance();
}
}

```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\class java\java1\java> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\class java\j
ava1\java\" ; if ($?) { javac Bank.java } ; if ($?) { java Bank }
Name:Divakar Babu M P
USN:1BM22CS093
Deposit of $5000.0 successful
Account Number: SA1001
Balance: 5000.0
Interest of $250.0 computed and added to the balance.
Account Number: SA1001
Balance: 5250.0
Withdrawal of 2000.0 successful
Account Number: SA1001
Balance: 3250.0
Deposit of $8000.0 successful
Account Number: CA2002
Balance: 8000.0
Withdrawal of 5000.0 successful.
Account Number: CA2002
Balance: 3000.0
```

LAB PROGRAM 6:

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.

CODES:

```java
package CIE;

public class Students {
    public String usn;
    public String name;
    protected int sem;

    public Students(String usn, String name, int sem) {
        this.usn = usn;
        this.usn = usn;
        this.usn = usn;
    }
}
```

```java
package CIE;

public class Internals extends Students {
    public int[] internalMarks = new int[5];

    public Internals(String usn, String name, int sem, int[] internalMarks) {
        super(usn, name, sem);
        this.internalMarks = internalMarks;
    }
}

```

```java
package SEE;

import CIE.Students;

public class External extends Students {
    public int[] seeMarks;

    public External(String usn, String name, int sem, int[] seeMarks) {
        super(usn, name, sem);
        this.seeMarks = seeMarks;
    }
}
```

```java
import CIE.*;
import SEE.*;

public class Main {
    public static void main(String[] args) {
        int[] internalMarks1 = {80, 75, 90, 85, 88};
        Internals student1 = new Internals("1ABC123", "John Doe", 3, internalMarks1);

        int[] seeMarks1 = {70, 85, 78, 92, 88};
        External student1SEE = new External("1ABC123", "John Doe", 3, seeMarks1);

        int[] finalMarks1 = new int[5];
        for (int i = 0; i < 5; i++) {
            finalMarks1[i] = student1.internalMarks[i] + student1SEE.seeMarks[i];
        }

        System.out.println("Final Marks for " + student1.name + " (USN: " + student1.usn + ")");
        for (int i = 0; i < 5; i++) {
            System.out.println("Course " + (i + 1) + ": " + finalMarks1[i]);
        }
    }
}
```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\New folder (4)> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\New folder (4)\" ;
if ($?) { javac Main.java } ; if ($?) { java Main }
Name:Divakar Babu MP
USN:1BM22CS093
Final Marks for null (USN: 1ABC123)
Course 1: 75
Course 2: 80
Course 3: 84
Course 4: 88
Course 5: 88
```

LAB PROGRAM 7:

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.

CODE:

```
public class Son extends Father{
Son(int Age){
super(Age);
}
@Override
void checkAge(){
if(Age>=super.Age){
throw new WrongAge("Son's age cannot be greater than father's age");
}
}
}
```

```java
public class Father {
    int Age;

    Father(int Age) {
        this.Age = Age;
    }

    void checkAge() throws WrongAge {
        if (Age < 0) {
            throw new WrongAge("Age can't be less than zero");
        } else {
            System.out.println(Age);
        }
    }
}
```

```java
public class WrongAge extends RuntimeException {
    WrongAge(String msg) {
        super(msg);
    }
}
```

```java
public class Main {
    public static void main(String[] args) {
        Father father = new Father(-1);
        Son son = new Son(20);
        try {
            father.checkAge();
            son.checkAge();
        } catch (WrongAge e) {
            System.out.println(e);
        }
    }
}
```

```java
public class Main {
    public static void main(String[] args) {
        Father father = new Father(20);
        Son son = new Son(30);
        try {
            father.checkAge();
            son.checkAge();
        } catch (WrongAge e) {
            System.out.println(e);
        }
    }
}
```

OUTPUT 1:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\class java> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\class java\" ; if ($?)
{ javac Main.java } ; if ($?) { java Main }
Name:DivakarBabu M P
USN:1BM22CS093
WrongAge: Age can't be less than zero
```

OUTPUT 2:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\class java> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\class java\" ; if ($?)
{ javac Main.java } ; if ($?) { java Main }
Name:DivakarBabu M P
USN:1BM22CS093
10
WrongAge: Son's age cannot be greater than father's age
```

LAB PROGRAM 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.

CODE:

```java
public class Main{
    public static void main(String[] args){
        BMSCE bmsce=new BMSCE();
        CSE cse=new CSE();
        Thread thread1=new Thread(bmsce);
        Thread thread2=new Thread(cse);
        thread1.start();
        thread2.start();
    }
}
```

```java
public class BMSCE implements Runnable {
    public void run() {
        try {
            for (int i = 0; i < 10; i++) {
                System.out.println("Bms college of Engineering");
                Thread.sleep(10000);
            }
        } catch (InterruptedException e) {
            System.err.println(e);
        }
    }
}

```

```java
public class CSE implements Runnable {
    public void run() {
        try {
            for (int i = 0; i < 10; i++) {
                System.out.println("Cse");
                Thread.sleep(2000);
            }
        } catch (InterruptedException e) {
            System.err.println(e);
        }
    }
}

```

OUTPUT:

```
PS C:\Users\Divakarbabu MP\OneDrive\Desktop\class java> cd "c:\Users\Divakarbabu MP\OneDrive\Desktop\class java\java1\java\"
 ; if ($?) { javac Main_1.java } ; if ($?) { java Main_1 }
Name:DivakarBabu M P
USN:1BM22CS093
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
```