

LAB - 1QUADRATIC

```
import java.io.*;
import java.util.*;
import java.lang.*;
public class quadratic
```

```
{  
    private static double a;  
    private static double b;  
    private static double c;  
    public static void read()
```

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

```
    System.out.println("Enter the Co-Efficient a");  
    a = sc.nextDouble();
```

```
    System.out.println("Enter the Co-Efficient b");  
    b = sc.nextDouble();
```

```
    System.out.println("Enter the Co-Efficient c");  
    c = sc.nextDouble();
```

```
    System.out.println("THANK YOU FOR ENTERING  
THE CO-EFFICIENTS");
```

```
}
```

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

```
{  
    read();
```

```
    double d = b * b - 4 * a * c;
```

```
    if (d > 0)
```

```
{
```

System.out.println ("ROOTS ARE REAL AND DISTINCT");

System.out.println ("FIRST ROOT IS " + (-b + Math.sqrt(d)) / (2 * a));

System.out.println ("SECOND ROOT IS " + (-b - Math.sqrt(d)) / (2 * a));

else if (d == 0)

{

System.out.println ("Roots are equal");

System.out.println ("Roots are " + (-b) / (2 * a));

else

{

System.out.println ("Roots are Imaginary");

System.out.println ("Roots are " + -b / (2 * a) + " + " + "i + (Math.sqrt(-d)) / (2 * a));

System.out.println ("Roots are " + -b / (2 * a) + " - " + "i + (Math.sqrt(-d)) / (2 * a));

}

public static void main (String [] args)

{

Class1;

}

}

LAB-2STUDENT

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

```
class Student
```

```
{
```

```
    private String USN;
```

```
    private String name;
```

```
    private int n;
```

```
    private double SPPA = 0;
```

```
    private int totalCredits = 0;
```

```
    private int credits[];
```

```
    private double marks[];
```

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

```
    void Details()
```

```
{
```

```
    System.out.println ("Enter USN of the student");
```

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

```
    System.out.println ("Enter name of the student");
```

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

```
    System.out.println ("Enter no. of subjects");
```

```
    n = ss.nextInt();
```

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

```
    marks = new double[n];
```

```
    System.out.println ("Enter details of subject : ");
```

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

```
{
```

```
    System.out.println ("Enter credits allotted to the subject
```

```
        "+(i+1));
```

credits[i] = ss.nextInt();

System.out.println ("Enter marks in the subject " + (i+1));

marks[i] = ss.nextInt();

calculate (credits[i], marks[i], i);

}

void calculate (int credit, double mark, int j)

{

totalCredits = totalCredits + credit;

if (mark >= 90 && mark <= 100)

SGPA = SGPA + (10 * credit);

else if (mark >= 80 && mark <= 89)

SGPA = SGPA + (9 * credit);

else if (mark >= 70 && mark <= 69)

SGPA = SGPA + (8 * credit);

else if (mark >= 60 && mark <= 59)

SGPA = SGPA + (7 * credit);

else if (mark >= 50 && mark <= 49)

SGPA = SGPA + (6 * credit);

else if (mark >= 40 && mark <= 39)

SGPA = SGPA + (5 * credit);

else

System.out.println ("Failed in subject " + (j+1));

}

void Display ()

{

```
System.out.println ("Details of the student");
```

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

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

```
System.out.println ("CGPA of Student " + (CGPA / total credit))
```

}

}

```
Class Main
```

{

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

{

```
Student s1 = new Student ();
```

```
s1.Details ();
```

```
s1.Display ();
```

}

}

LAB-3 Books

```
import java.util.*;
```

```
Class book
```

```
{
```

```
String booktitle;
```

```
String author;
```

```
int no_of_pages;
```

```
double price;
```

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

```
book ()
```

```
{
```

```
System.out.print ("Enter book title:");
```

```
booktitle = sc.nextLine();
```

```
System.out.print ("Enter the author name:");
```

```
Author = sc.nextLine();
```

```
System.out.print ("Enter the price:");
```

```
price = sc.nextDouble();
```

```
System.out.print ("Enter the pages:");
```

```
no_of_pages = sc.nextInt();
```

```
}
```

```
public String toString ()
```

```
{
```

```
return ("Book name = " + booktitle + " Author = " + author + "
```

```
Price = " + price + " Pages = " + no_of_pages);
```

```
}
```

```
}.
```

class Books

{

public static void main (String [] args)

{

int n, i;

Scanner in = new Scanner (System.in);

System.out.println ("Enter number of books:");

n = in.nextInt();

book [] b = new book [n];

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

{

System.out.println ("Enter details of Books: " + (i+1));

b [i] = new book ();

y

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

{

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

}

}

}

LAB - 4SHAPE

```
import java.util.*;
import java.lang.Math.*;
```

abstract class Shape {

```
public int a;
```

```
public int b;
```

```
abstract public void printArea();
```

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

}

Class rectangle extends shape {

```
public int void printArea () {
```

System.out.print ("Please enter length
and breadth of rectangle : ");

```
float a = s.nextFloat();
```

```
float b = s.nextFloat();
```

```
float area = a * b;
```

```
System.out.println ("Area = " + area + "sq. units");
```

y

)

Class triangle extends shape {

```
public void printArea () {
```

System.out.print ("Please enter three sides of
triangle : ");

```
float a = s.nextFloat();
```

```
float b = s.nextFloat();
```

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

```
float d = (a+b+c)/2;
```

```
double area = Math.sqrt((d*(d-a)*(d-b)*(d-c))));
```

```
System.out.println("Area = " + area + " sq. units");
```

{}

}

class circle extends shape {

```
public void printArea() {
```

```
System.out.println("Please enter the radius  
of circle : ");
```

```
float a = s.nextInt();
```

```
float area = 22/7 * a * a;
```

```
System.out.println("Area = " + area + " sq. units");
```

{}

{}

class ShapeDemo {

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

```
Shape r = new rectangle();
```

```
Shape t = new triangle();
```

```
Shape c = new circle();
```

```
for (int i=0; i<100; ++i)
```

```
System.out.println("In Triangle")
```

```
Triangle In")
```

```
Circle In");
```

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

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

Switch (c++)

Case 1: t.printArea();
break;

Case 2: r.printArea();
break; (for next case)

Case 3: c.printArea();
break; (for next case)

default: System.out.println ("invalid
choice");

{Case 1, 2, 3} Main.java

y y (if you want to use switch)

if you want to use if else if then else,

if you want to use for loop,

> (Main.java)

else if

else if you want to use for loop and for

if and else if and

switch (if needed)

if you want to use if else if then else or

> (Main.java)

intelliJ

using the new file intelliJ idea (ctrl + shift + i)

Ctrl + Alt + I

LAB-5BANK

```
import java.util.*;
```

```
class Account {
```

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

```
String cusName, accType;
```

```
long accNumber;
```

```
double balance = 9876.5
```

```
void Accept () {
```

```
System.out.println ("Enter name");
```

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

```
System.out.println ("Enter account number");
```

```
accNumber = in.nextLong ();
```

```
}
```

```
void deposit () {
```

```
int dep;
```

```
System.out.println ("Enter the amount to be deposited");
```

```
dep = in.nextInt();
```

```
balance += dep;
```

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

```
}
```

```
void withdrawal () {
```

```
int withdraw;
```

```
System.out.println ("Enter the amount you want  
to withdraw");
```

```
    withdraw = in.nextInt();
```

```
    balance -= withdraw;
```

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

{}

}

```
class CurrentAccount extends Account {
```

```
    void penalty () {
```

```
        if (balance < 2000) {
```

```
            balance -= 400;
```

```
            System.out.println("400 penalty for maintained less  
than minimum balance");
```

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

{}

}

}

```
class Bank {
```

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

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

```
        System.out.println ("Enter your choice \n1. Savings  
Account \n2. Current Account");
```

```
        int choice = sc.nextInt();
```

```
        CurrentAccount c = new CurrentAccount();
```

```
        SavingsAccount s = new SavingsAccount();
```

```
        if (choice == 2) {
```

```
            c.Accept();
```

System.out.println ("Enter your choice\n1. Deposit\n2. Withdraw");

int n = sc.nextInt();

switch () {

case 1: {

S.deposit();

break;

}

case 2: {

S.withdraw();

S.penalty();

break;

}

~~default:~~

System.out.println ("Wrong choice");

}

if (choice == 1) {

S.Accept();

System.out.println ("Enter your choice\n1. Deposit\n2. Withdraw");

int n = sc.nextInt();

switch (n) {

case 1: {

S.deposit();

S.withdraw();

break; final for splitting

{

transient

Case 2: {

s.withdrawal();

breaks;

}

default: System.out.println ("wrong choice");

}

{

public class ATM {

y

{ atm. balance = 0; }

{ atm. max. limit = 1000; }

{ atm. min. limit = 100; }

{ atm. withdrawal = 0; atm. i = 0; }

{ atm. withdraw (int amount) { atm. i = 0; atm. balance = atm. balance - amount; atm. withdrawal = amount; atm. printBalance (); } }

{ atm. printBalance () {

System.out.println ("Available balance is " + atm. balance);

{

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

LAB 6PACKAGE PROGRAMCIE→ INTERNALS

```
package CIE;
import java.util.*;
```

```
public class Internals extends Student {
```

```
    public double cie() {
```

```
        public void display () {
```

```
            item = new double [5];
```

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

```
System.out.println ("Enter cie marks out of 50:");
```

```
for (int i=0; i<5; i++) {
```

```
    item [i] = E.nextDouble();
```

```
}
```

```
y
```

```
y
```

→ STUDENT

```
package CIE;
import java.util.*;
```

```
public class Student {
```

```
    public String Name;
```

```
    public String USN;
```

```
    public int sem;
```

```
    public void display() {
```

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

```
        System.out.println ("Name :");
```

```
        Name = sc.nextLine();
```

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

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

```
        System.out.println ("Sem : ");
```

```
        Sem = sc.nextInt();
```

3

g

SEE

→ EXTERNAIS

```
package SEE;
```

```
import CIE.*;
```

```
import java.util.*;
```

```
public class External extends CIE.Student {
```

```
    public double sem[];
```

```
    public void display() {
```

```
Scen = new Scanner double [5];
```

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

```
System.out.println ("SEE marks for subject  
out of 100:");
```

```
for (int i=0; i<5; i++) {
```

```
Scen [i] = s.nextInt();
```

```
}
```

```
}
```

MAIN

```
import CIE.*;
```

```
import SEE.*;
```

```
import java.util.*;
```

```
public class Main {
```

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

```
int n;
```

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

```
System.out.println ("Enter .no. of students: ");
```

```
n = sc.nextInt();
```

```
CIE.Student st [] = new CIE.Student [n];
```

~~SEE.Externals ex [] = new SEE.Externals [n];~~

~~CIE.Internal st [] = new CIE.Internal [n];~~

~~SEE.Externals ex [] = new SEE.Externals [n];~~

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

st[i] = new CIE.Student();

in[i] = new CIE.Externals();

ex[i] = new SEE.Externals();

st[i].display();

in[i].display();

ex[i].display();

System.out.println("Total Marks of " + st[i].name + "\n");

for (int j=0; j<5; j++) {

System.out.println(in[i].item[j] + ex[i].item[j]/2);

}

}

}

}

LAB 7GENERIC

import java.io.*;
import java.lang.*;
import java.util.*;

class gen<T>
{

T ob;

gen(T o)
{

ob=o;

}

T getob()
{

return ob;

}

void showtype()
{

System.out.println ("Type of T is "+ ob.getClass().

getName());

}

}

class generic

public static void main (String [] args)

String n;

Scanner sc = new Scanner(System.in);

System.out.println("Enter the Integer Number");

to be Displayed using the generic style");

n = sc.nextInt();

gen<Integer> ob1 = new gen<Integer>(Integer.
valueOf(n));

ob1.showtype();

int val = ob1.getOb();

System.out.println("Value is: " + val);

System.out.println();

System.out.println("Enter the String to be Displayed
using the generic style:");

n = sc.nextLine();

gen<String> ob2 = new gen<String>(n);

ob2.showtype();

String x = ob2.getOb();

System.out.println("Value : " + x);

System.out.println();

System.out.println("Enter the Double Number.
to be Displayed Using the generic style:");

n = sc.nextDouble();

```

gen <Double> ob3 = new gen <Double> (Double.parseDouble("1"));
ob3.showType();
double ans = ob3.getOb();
System.out.println ("Value : " + ans);
    
```

y

(Object)

(Object)

• return object - return value (Object) type

(Object)

(Object) type

• return value (Object) type

(Object) type

• return value (Object) type

(Object) type is returned

int function

• return value (Object) type

(Object) type

Object type

• return value (Object) type

(Object) type

• return value (Object) type

Object type

Object type

LAB 8AGE

```
import java.util.*;
```

```
class fatherAgeException extends Exception  
{
```

```
    public String toString()  
    {
```

```
        return ("Father's age is less than 0");  
    }
```

```
class sonAgeException extends Exception  
{
```

```
int a1;
```

```
sonAgeException (int age)  
{
```

```
    a1 = age;  
}
```

```
public String toString()  
{
```

```
    if (a1 < 0)
```

```
        return ("Son's age is less than 0");
```

```
    else
```

```
        return ("Son's age is more than father's age");
```

```
}
```

```
}
```

```
class father
```

{

public int age1;

Scanner sc = new Scanner (System.in);

father () {

System.out.print ("Enter father's age: ");

age1 = sc.nextInt ();

}

void ex1 () throws FatherAgeException {

if (age1 < 0)

throws new FatherAgeException ();

}

}

class son extends father

{

public int age2;

son () {

System.out.print ("Enter son's age: ");

age2 = sc.nextInt ();

void ex2 () throws SonAgeException {

{

if (age2 < 0 || age2 > super.age1)

throws new SonAgeException (age2);

}

}

class Main

{

public static void main (String args [])

{

Scanner s1 = new Scanner (System.in);

try

s1. nextInt();

{

catch (InputMismatchException e)

{

System.out.println (e);

}

try

{ s1.nextInt();

s1.nextInt();

}

catch (InputMismatchException e)

{

System.out.println (e);

}

}

LAB 9THREAD

Class bus implements Runnable {

Thread t1;

bus() {

t1 = new Thread (this, 'bus');

}

public void run () {

try {

for (int i = 5; i > 0; i--) {

System.out.println ("BMS College of Engineering");

Thread.sleep (10000);

}

}

Catch (InterruptedException e) {

System.out.println ("BMS interrupted");

}

System.out.println ("Exiting: " + t1);

}

Class Cse implements Runnable {

Thread t2;

cse() {

t2 = new Thread (this, "cse");

}

public void run () {

```
try {
```

```
    for (int i = 5; i > 0; i--) {
```

```
        System.out.println ("BMS (SE)");
```

```
        Thread.sleep (2000);
```

```
}
```

```
}
```

```
Catch (InterruptedException e) {
```

```
    System.out.println ("Exiting : "+t1);
```

```
    "CSE interrupted\n");
```

```
t1.interrupt();
```

```
    System.out.println ("Exiting : "+t2);
```

```
}
```

```
}
```

```
class threadprg {
```

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

```
    BMS obj1 = new BMS ();
```

```
    CR obj2 = new CR ();
```

```
    obj1.t1.start ();
```

```
    obj2.t2.start ();
```

```
y
```

```
y
```

MULTIPLICATION
TABLE OF 5LAB10

Class Table

{

void printTable (int n)

{

synchronized (this)

{

for (int i = 1; i < 5; i++)

{

System.out.println (" " + n + " * " + i + " = " + (n * i));

try

{

Thread.sleep (400);

}

catch (Exception e)

{

System.out.println (e)

}

}

}

}

}

Class Mythread extends Thread

{

Table t;

My Thread1 (Table 1)

{

this.t=t;

}

public void run()

{

t.printTable(5);

}

}

Class MyThread2 extends Thread

{

Table t;

MyThread2(Table t)

{

this.t=t;

}

public void run()

{

t.printTable(100);

}

}

Class Use

{

public static void main (String args[])

{

Table obj = new Table();

Mythread1.th1 = new Mythread1(obj);

Mythread2.th2 = new Mythread2(obj);

th1.start();

th2.start();

}

}