

Harsik Doshi  
IBN19CS057

1) Quadratic Equation:

```

import java.util.Scanner;
public class Quadratic {
    public static void main(String[] args) {
        Scanner out.print("Input a: ");
        double a = input.nextDouble();
        System.out.print("Input b: ");
        double b = input.nextDouble();
        System.out.print("Input c: ");
        double c = input.nextDouble();
        double result = b*b - 4.0*a*c;
        if (result > 0.0) {
            double r1 = (-b + Math.pow(result, 0.5)) / (2.0*a);
            double r2 = (-b - Math.pow(result, 0.5)) / (2.0*a);
            System.out.println("The roots are " + r1 + " and " + r2);
        } else if (result == 0.0) {
            double r1 = -b / (2.0*a);
            System.out.println("The roots are equal and real and is " + r1);
        } else {
            System.out.println("The equation has no real roots.");
        }
    }
}

```

## Lab-2

```
2) import java.util.*;  
class Student  
{  
    private String USN;  
    private String name;  
    private int n;  
    private double sgpa=0;  
    private int total credits = 0;  
    Scanner ss = new Scanner( System.in );  
  
    void details()  
    {  
        System.out.printIn("Enter the USN of the student");  
        USN = ss.nextLine();  
        System.out.printIn("Enter the name of the student");  
        name = ss.nextLine();  
        System.out.printIn("Enter the no. of subjects");  
        n = ss.nextInt();  
        int credits[] = new int[n];  
        double marks = new double[n];  
        System.out.printIn("Enter the details of subjects");  
        for(i=0; i<n; i++)  
        {  
            System.out.printIn("Enter the credits allotted to the subject" + (i+1));  
            credits[i] = ss.nextInt();  
            System.out.printIn("Enter the marks of the subject" + (i+1));  
            marks[i] = ss.nextInt();  
            calculate(credits[i], marks[i]);  
        }  
    }  
}
```

Harshil Doshi  
IBM19CS057

```

void calculate ( int credit, double mark, int j )
{
    total credits = total credits + credit;
    if (mark >= 90 && mark <= 100)
        SGPA = SGPA + (10 * credits);
    else if (mark >= 80 && mark <= 89)
        SGPA = SGPA + (9 * credits);
    else if (marks > 70 && mark <= 79)
        SGPA = SGPA + (8 * credits);
    else if (mark >= 60 && mark <= 59)
        SGPA = SGPA + (7 * credits);
    else if (mark >= 50 && mark <= 49)
        SGPA = SGPA + (6 * credits);
    else if (mark >= 40 && mark <= 39)
        SGPA = SGPA + (5 * credits);
    else
        System.out.println ("Failed in subject " + (j+1));
}

```

### void Display()

```

{
    System.out.println ("Details of the Student");
    System.out.println ("Name: " + Name);
    System.out.println ("USN: " + usn);
    System.out.println ("SGPA of student " + (SGPA/Total credits));
}

```

```

}
class Main
{
    public static void main (String args[])
    {
        Student s1 = new Student ();
        s1.Details();
        s1.Display();
    }
}

```

Lab-3  
Harshil Doshi  
IBN19CS057

```

import java.util.Scanner;
class book{
    String author;
    String title;
    String numPages;
    String price;
    Scanner sc = new Scanner(System.in);
    void getdetails(){
        System.out.print("author");
        author = sc.next();
        System.out.print("booktitle");
        title = sc.next();
        System.out.print("Num of pages");
        numPages = sc.next();
        System.out.print("Price");
        price = sc.next();
    }
    public String toString(){
        return ("Author:" + author + "\nBooktitle:" + title + "\nPages:" + numPages + "\nprice:" + price);
    }
    book(){
        author = "Sanju";
        Title = "Myself";
        numPages = "2875";
        price = "1900";
    }
    void display(){
        System.out.print("author:" + author);
        System.out.print("Title:" + title);
        System.out.print("No.of Pgs:" + numPages);
        System.out.print("Price:" + price);
    }
}

```

class bookmain

```
{ public static void main (String ss [ ]) {  
    Scanner sc = new Scanner (System . in );  
    int n;  
    book s1 = new book ();  
    System . out . printIn (" To show default constructor ");  
    s1 . display ();  
    System . out . printIn (" Enter no . of books ");  
    n = sc . nextInt ();  
    book b [] = new book [n];  
    for (i = 0; i < n; i++) {  
        system . out . printIn (" Enter details of book " + (i + 1));  
        b [i] = new book ();  
        b [i] = getdetails ();  
    }  
    System . out . printIn (" \n Details of all books ");  
    for (i = 0; i < n; i++) {  
        system . out . printIn (" \n Book : " + (i + 1));  
        System . out . printIn ( b [i] . tostring ());  
    }  
}
```

```
import java.util.Scanner;  
abstract class shape  
{  
    private int a, b;  
    void setshape (int x, int y)  
    {  
        a=x;  
        b=y;  
    }  
    int geta()  
    {  
        return a;  
    }  
    int getb()  
    {  
        return b;  
    }  
    abstract public void print_area();  
}  
class rectangle extends shape  
{  
    private int area_rect; // calculate  
    rectangle (int x, int y) // constructor  
    {  
        setshape (x, y);  
        area_rect = geta() * getb();  
    }  
    public void print_area()  
    {  
        area_rect = geta() * getb();  
        System.out.println ("Area of rectangle is: " +  
                           area_rect);  
    }  
}  
class triangle extends shape  
{  
    private double area_tri; // calculate  
    triangle (int x, int y) // constructor  
    {  
        setshape (x, y);  
        area_tri = 0.5 * geta() * getb();  
    }  
    public void print_area()  
    {  
        System.out.println ("Area of triangle is: " +  
                           area_tri);  
    }  
}
```

```

public void printArea()
{
    areaFor = (geta() + getb()) / 2;
    System.out.println ("The area of triangle is: " + areaFor);
}

class Circle extends Shape
{
    private double areaCircle;
    Circle (int y)
    {
        setShape (0, y);
    }

    public void printArea()
    {
        areaCircle = ((3.14 * getb()) * getb());
        System.out.println ("Area of circle is: " + areaCircle);
    }
}

```

```

public class week 8
{
    public static void main (String [] args)
    {
        Scanner xx = new Scanner (System.in);

        int a, b;
        System.out.print ("Enter the length of rectangle: ");
        a = xx.nextInt();
        System.out.print ("Enter the breadth of rectangle: ");
        b = xx.nextInt();

        Rectangle r = new Rectangle (a, b);
        r.printArea();

        System.out.print ("Enter the length of triangle: ");
        a = xx.nextInt();
        System.out.print ("Enter the base of triangle: ");
        b = xx.nextInt();
    }
}

```

```
triangle t = new triangle(a,b);  
t = print-area();  
System.out.printIn("Enter the radius of the circle: ");  
a = nextInt()  
circle c = new circle(a);  
c.print-area();  
}  
}
```

## Lab 5:

Haresh Doshi  
IBM i95057

```
import java.util.Scanner;
import java.lang.Math;
class Account
{
    private String name;
    private double account_no;
    private char account_type;
    private double balance;
    void getData(char ch)
    {
        Scanner xx = new Scanner(System.in);
        System.out.print("Enter the name of the customer: ");
        name = xx.next();
        xx.nextLine();
        System.out.print("Enter the account no. of customer: ");
        account_no = xx.nextDouble();
        System.out.print("Enter the balance of customer: ");
        balance = xx.nextDouble();
        account_type = ch;
    }
    void updatebalance(double x)
    {
        balance = balance + x;
    }
}
```

```

void updatebalance1 (double x)
{
    balance = balance - x;
}

void updatebalance2 (double x)
{
    balance = x;
}

double getbalance()
{
    return balance;
}

void displaybalance()
{
    System.out.println ("The balance is " + balance);
}

class SavingAccount extends Account
{
    private double interestRate;

    saving account()
    {
        Scanner xx = new Scanner (System.in);
        getData ('s');

        System.out.print ("Enter the interest rate");
        interestRate = xx.nextDouble();

        System.out.print ("Enter the amount to be deposited");
        double x = xx.nextDouble();
        updatebalance (x);
    }

    void gotdeposit()
    {
        Scanner xx = new Scanner (System.in);
        System.out.print ("Enter the amount to be deposited");
        double x = xx.nextDouble();
        updatebalance (x);
    }
}

```

```

void computeInterest()
{
    Scanner xx = new Scanner (System.in);
    System.out.print("Enter the no. of yrs : ");
    double time = xx.nextDouble();
    double x = (getBalance() * Math.pow ((1 + (interestRate)
                                             / 100)), time));
    updateBalance(x);
    System.out.println ("The computed interest is : " + x);
    displayBalance();
}

void withdrawl()
{
    Scanner xx = new Scanner (System.in);
    System.out.print("Enter the amount to be withdrawn : ");
    double f = xx.nextDouble();
    while (f > getBalance());
    System.out.println ("The amount withdrawn is more
                        than the balance after again : ");
    xx = nextDouble();
    updateBalance(f);
    displayBalance();
}

```

```

class CurrentAccount extends Account
{
    private double minBalance;
    private int chequeBook;
    CurrentAccount()
    {
        Scanner xx = new Scanner (System.in);
        getData('C');
    }
}

```

System.out.print("Enter the min. balance");

minBalance = sc.nextDouble();

void getDeposit()

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

System.out.print("Enter the amount to be deposited:");

double x = sc.nextDouble();

updateBalance(x);

void issueCheck()

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

System.out.print("Enter the amount of check");

double n = sc.nextDouble();

if (n > (getBalance() - minBalance))

{ System.out.println("You have issued a cheque of  
more than min. bal & have charged the penalty  
of 100 rupees.");

updateBalance(100);

{

else

x = updateBalance(x);

{

displayBalance();

{

void withdraw()

{ System.out.print("Enter the amount to be withdrawn:");

Scanner sc = new Scanner(System.in);

double x = sc.nextDouble();

while (x > (getBalance() - minBalance))

{ System.out.print("Amt. withdrawn more than  
balance enter again");

x = sc.nextDouble();

{

updateBalance();  
displayBalance();

}  
}

## Lab 6:

CODE for CIE:

Hanshil Doshi

1B91ACSO57

### Internals.java:

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

public class Internals extends Student {
    Scanner in = new Scanner(System.in);
    public double cie_marks[] = new double[5];
    public Internals() {
        super();
        System.out.print("Enter CIE marks of 5 subjects:-");
        for (int i=0; i<5; i++) {
            do {
                System.out.print("Marks in subject " + (i+1) + ":");
                cie_marks[i] = in.nextDouble();
                if (cie_marks[i] > 50)
                    System.out.println("ERROR: MARKS CANNOT BE OVER
50. PLEASE INPUT AGAIN!");
            } while (cie_marks[i] > 50);
        }
    }
}
```

### Student.java:

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

public class Student {
    Scanner in = new Scanner (System.in);
```

```
int Usn;  
String name;  
int Sem;  
public Student(){  
    System.out.println("Enter name:");  
    name = in.nextLine();  
    System.out.println("Enter USN:");  
    USN = in.nextInt();  
    System.out.println("Enter Sem:");  
    Sem = in.nextInt();  
}  
}
```

CODE for SEE:

### External.java

```
package SEE;  
import java.*;  
import java.util.*;  
import java.lang.*;  
public class External extends Internals  
{  
    Scanner in = new Scanner(system-in);  
    public double see_marks[] = new double[5];  
    public External() {  
        super();  
        System.out.println("Enter SEE marks of 5 subjects:-");  
        for(int i=0; i<5; i++) {  
            do {  
                System.out.println("Marks in subject " + (i+1) + ": ");  
                see_marks[i] = in.nextDouble();  
            } while (see_marks[i] > 100);  
        }  
    }  
}
```

System.out.println("ERROR MARKS CANNOT BE OVER 100; PLEASE INPUT AGAIN");

```
while(see_marks[i]>100);  
{  
}  
{  
}  
}
```

### Main.java:

```
import CIE.*;  
import SEE.*;  
import java.util.*;  
import java.lang.*;  
class Main{  
    public static void Tot_marks(External obj, double[ ] arr){  
        for(int i=0; i<5; i++)  
        { arr[i] = (obj.CIE_marks[i]+(obj.SEE_marks[i]))/2; }  
    }  
}  
public static void main(String[ ] args){  
    Scanner in = new Scanner(System.in);  
    int n;  
    double Total[ ] = new double[5];  
    System.out.print("Enter the no. of students:- ");  
    n = in.nextInt();  
    External e[ ] = new External[n];  
    for(int i=0; i<n; i++){  
        e[i] = new External();  
        Tot_marks(e[i], Total);  
        System.out.print("Total marks: ");  
        for(int j=0; j<5; j++){  
            System.out.print(Total[j]);  
        }  
    }  
}
```

Lab 7:

```

class Generic<T,U> {
    T obj1;
    U obj2;

    Generic(T x, U y) {
        obj1 = x;
        obj2 = y;
    }

    T getObj1() {
        return obj1;
    }

    U getObj2() {
        return obj2;
    }

    void display() {
        System.out.println("Obj1:" + getObj1());
        System.out.println("Obj2:" + getObj2());
    }

    U join() {
        if (obj1 instanceof Integer && obj2 instanceof Integer)
            int i1 = (Integer) getObj1();
            int i2 = (Integer) getObj2();
            return (U) new Integer(i1 + i2);

        else if (obj1 instanceof Double && obj2 instanceof Double)
            double d1 = (Double) getObj1();
            double d2 = (Double) getObj2();
            return (U) new Double(d1 + d2);

        else if (obj1 instanceof String && obj2 instanceof String)
            String s1 = (String) getObj1();
            String s2 = (String) getObj2();
            return (U) new String(s1 + s2);
    }
}

```

```

else {
    return (U) new String ("ERROR: obj1 & obj2 Type Mismatch..."),
}
}

public class Lab7 {
    public static void main (String [] args) {
        Generics < Integer, Integer > iObj = new Generics < Integer, Integer >
            (5, 4),
        iObj . display ();
        System. out. printIn ("Sum: " + iObj. join ());
        System. out. printIn ();
        Generics < Double, Double > dObj = new Generics < Double, Double >
            (3.05, 4.02),
        dObj. display ();
        System. out. printIn ("Sum: " + dObj. join ());
        System. out. printIn ();
        Generics < String, String > sObj = new Generics < String, String >
            ("Hello.", "How are you?"),
        sObj. display ();
        System. out. printIn ("Concatenation: " + sObj. join ());
        System. out. printIn ();
    }
}

```

Lab 8:

```
import java.util.Scanner;
class WrongAge extends Exception
{ public WrongAge(String s)
```

```
{ Super(s);
}
```

```
class Father{
```

```
    int fatherAge;
```

```
    Father()
```

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

```
    System.out.print("Enter Father's Age: ");
```

```
    fatherAge = inp.nextInt();
```

```
}
```

```
class Son extends Father{
```

```
    int sonAge;
```

```
    Son()
```

```
{ super();
```

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

```
try { if (fatherAge <= 0)
```

```
{ throw new WrongAge ("Age cannot be less than or equal  
to zero ");
```

```
}
```

```
else
```

```
{ System.out.print("Enter Son's Age: ");
```

```
SonAge = inp.nextInt();
```

```
try { if (sonAge >= fatherAge)
```

```
{
```

throw new WrongAge ("Age of son cannot be greater than or equal to age of father");

}  
else { System.out.println("Entered ages are " + a + ", " + b);

}  
}  
catch (WrongAge e)

{ System.out.println(e);

}  
}  
catch (WrongAge e)

{ System.out.println(e);

}  
}  
}

class ageException{

public static void main (String [] args)

{ Son s = new Son();

}  
}

MultithreadDemo.java:

Harshil Doshi

```

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

class NewThread implements Runnable {
    Thread t;
    String s;
    int x;

    NewThread(String threadName, int x) {
        s = threadName;
        this.x = x;
        t = new Thread(this, s);
        t.start();
    }

    public void run() {
        try {
            for (int i = 0; i < 10; i++) {
                System.out.println(s);
                Thread.sleep(x);
            }
        } catch (InterruptedException e) {
            System.out.println("Thread Interrupted");
        }
    }
}

public class MultithreadDemo {
    public static void main(String[] args) {
        new NewThread("BMS college of Engineering", 10000);
        new NewThread("CSE", 2000);
    }
}

```

```

import java.awt.*;
import java.awt.event.*;
public class Main extends Frame implements ActionListener
{
    TextField tf1, tf2;
    Label l;
    Button b;
    Main()
    {
        tf1 = new TextField(10);
        tf1.setBounds(50, 50, 200, 25);
        tf2 = new TextField();
        tf2.setBounds(50, 100, 200, 25);
        l = new Label();
        l.setBounds(50, 150, 200, 25);
        b = new Button("OK");
        b.setBounds(50, 200, 100, 50);
        b.addActionListener(this);
        add(b);
        add(tf1);
        add(tf2);
        add(l);
        setSize(800, 800);
        setLayout(null);
        setVisible(true);
    }
    public void actionPerformed(ActionEvent e)
    {
        try
        {
            String n1 = tf1.getText();
            String n2 = tf2.getText();
        }
    }
}

```

```
1. setText("Quotient" + (Integer.parseInt(n1) /  
* catch(NumberFormatException ze){  
    l. setText("Cannot divide non-numerical / non-integer values");  
* }  
catch(Exception ex){  
    System.out.println(ex);  
* }  
* }  
* }  
public static void main(String[] args){  
    new Main();  
* }
```