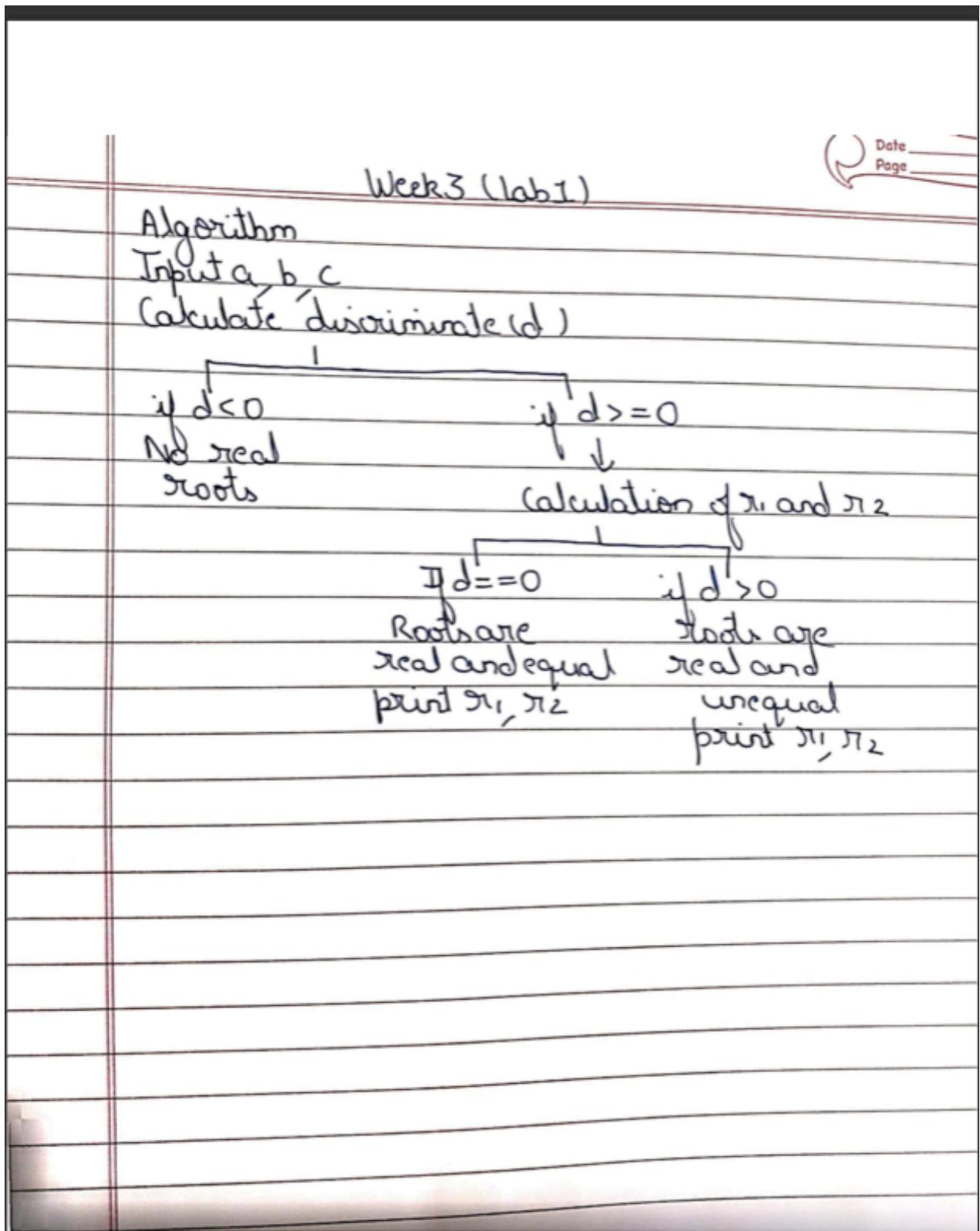


## Lab Program 1

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



```
import java.util.Scanner;
class quadratic
{
    public static void main(String [] args)
    {
        Scanner ss = new Scanner(System.in);
        System.out.println("Enter the value of a:");
        double a = ss.nextDouble();
        System.out.println("Enter the value of b:");
        double b = ss.nextDouble();
        System.out.println("Enter the value of c:");
        double c = ss.nextDouble();

        double d = Math.sqrt(b*b) - (4*a*c);
        double r1, r2;
        if(d > 0)
        {
            r1 = (-b + d) / (2 * a);
            r2 = (-b - d) / (2 * a);
            System.out.printf("Roots are real and distinct  
%f and %f", r1, r2);
            System.out.println();
        }
        else if(d == 0)
        {
            r1 = r2 = (-b) / (2 * a);
            System.out.printf("Roots are real and equal:  
%f and %f", r1, r2);
        }
    }
}
```

else

{

System.out.println("Roots are complex and not real");

System.out.println(;

}

}

Desktop — bash — 80x27

```
Ishas-MacBook-Air:Desktop isha$ javac Lab1.java
Ishas-MacBook-Air:Desktop isha$ java quadratic
Enter the value of a:
1
Enter the value of b:
-6
Enter the value of c:
5
Roots are real and distinct : 5.00 and 1.00
Ishas-MacBook-Air:Desktop isha$ java quadratic
Enter the value of a:
1
Enter the value of b:
4
Enter the value of c:
5
Roots are complex and not real
Ishas-MacBook-Air:Desktop isha$ java quadratic
Enter the value of a:
9
Enter the value of b:
-6
Enter the value of c:
1
Roots are real and equal :0.33 and 0.33
Ishas-MacBook-Air:Desktop isha$
```

## Lab Program 2:

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.

LAB-2

PAGE NO.	11
DATE	

```
import java.util.Scanner;
class Student
{
    String usn, name;
    double cgpa;
    int credits[][];
    double marks[][];
    void getData()
    {
        Scanner ss = new Scanner(System.in);
        System.out.println("Enter the Name and USN of
                           student respectively");
        name = ss.nextLine();
        usn = ss.nextLine();
        System.out.println("Enter the number of subjects :-");
        n = ss.nextInt();
        credits = new int[n][];
        marks = new double[n];
        for(int i=0; i<n; i++)
        {
            System.out.println("Enter the number "+(i+1)
                               + " subject marks and credits respectively");
            marks[i] = ss.nextDouble();
            credits[i] = ss.nextInt();
        }
    }
}
```

```

void cal_cgpa()
{
    double sum=0.0;
    for(int i=0; i<marks.length; i++)
    {
        if(marks[i] >= 90 && marks[i] <= 100)
            sum += 10 * credits[i];
        else if(marks[i] >= 80 && marks[i] < 90)
            sum += 9 * credits[i];
        else if(marks[i] >= 70 && marks[i] < 80)
            sum += 8 * credits[i];
        else if(marks[i] >= 60 && marks[i] < 50)
            sum += 7 * credits[i];
        else if(marks[i] >= 50 && marks[i] < 40)
            sum += 6 * credits[i];
        else if(marks[i] >= 40 && marks[i] < 50)
            sum += 5 * credits[i];
        else
            sum += 0;
    }
}

```

```

int creditSum = 0;
for (int i = 0; i < credits.length; i++) {
    creditSum += credits[i];
}
cgpa = (double) sum / creditSum;
}

void printData()
{
    System.out.println("Student Details:");
    System.out.println("Student name: " + name);
    System.out.println("Student wno: " + wno);
    System.out.println("Student CGPA: " + cgpa);
}

class StudentMain
{
    public static void main(String args[])
    {
        Student s1 = new Student();
        s1.getData();
        s1.cal_cgpa();
        s1.printData();
    }
}

```

[Ishas-MacBook-Air:Desktop isha\$ Desktop — -bash — 80x24

```
Ishas-MacBook-Air:Desktop isha$ javac Lab2.java
Ishas-MacBook-Air:Desktop isha$ java StudentMain
Enter the Name and USN of student respectively
Isha 1BM19EC057
Enter the number of Subjects:
5
Enter the number 1 subject marks and credit respectively
95 5
Enter the number 2 subject marks and credit respectively
94 4
Enter the number 3 subject marks and credit respectively
93 4
Enter the number 4 subject marks and credit respectively
87 4
Enter the number 5 subject marks and credit respectively
90 3
Student Details:
Student name: Isha
Student USN: 1BM19EC057
Student CGPA: 9.8
Ishas-MacBook-Air:Desktop isha$
```

### 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.

Lab-3

PAGE NO.	
DATE	/ /

```
import java.util.Scanner; // for input output  
class Book  
{  
    String name; // string variable  
    String author; // string variable  
    double price; // double variable  
    int pages; // integer variable
```

Book()  
{

```
    name = " ";  
    author = " ";  
    price = 0.0;  
    pages = 0;
```

```
    void getDetails()  
{
```

```
        Scanner ss = new Scanner(System.in);  
        System.out.println("Enter the book's name:");  
        name = ss.nextLine();
```

```
        System.out.println("Enter the author's name:");  
        author = ss.nextLine();
```

```
        System.out.println("Enter the price of book:");  
        price = ss.nextDouble();
```

```
        System.out.println("Enter the number of pages in  
        the book");
```

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

PAGE NO. / /  
DATE / /

```
public String toString()
{
    return "Name of book :" + name + "\n"
           + "Author of book" + author + "\n" + "Price :" + price
           + "\n" + "Number of pages :" + pages;
}
```

```
class BookMain
```

```
{
```

```
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number
                           of Books");
        int n = sc.nextInt();
        Book b[] = new Book[n];
        for(int i=0; i<n; i++)
        {
            System.out.print("Book " + (i+1) + " details");
            b[i] = new Book();
            b[i].getDetails();
        }
        for(int i=0; i<n; i++)
        {
            System.out.println(b[i]);
            System.out.println();
        }
    }
}
```

```
}
```

```
3
```

Desktop -- bash -- 80x34

```
Ishas-MacBook-Air:Desktop isha$ javac Lab3.java
Ishas-MacBook-Air:Desktop isha$ java BookMain
Enter the number of Books :
2
Book 1 details
Enter the book's name:
MOV
Enter the author's name:
Shakespeare
Enter the price of book:
200
Enter the number of pages in the book:
100
Book 2 details
Enter the book's name:
Alice
Enter the author's name:
Mark
Enter the price of book:
300
Enter the number of pages in the book:
200
Name of book: MOV
Author of book :Shakespeare
Price :200.0
Number of pages: 100

Name of book: Alice
Author of book :Mark
Price :300.0
Number of pages: 200
```

Ishas-MacBook-Air:Desktop isha\$

## 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.

Lab4

RAISE NO.	
DATE	/ /

```
import java.util.Scanner;
abstract class Shape {
    double d1;
    double d2;
    abstract void printArea();
}

class Rectangle extends Shape {
    Rectangle() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the dimensions of rectangle:");
        d1 = sc.nextDouble();
        d2 = sc.nextDouble();
    }

    void printArea() {
        System.out.println("Area of rectangle :" + d1 * d2);
    }
}

class Triangle extends Shape {
    Triangle() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter dimensions of triangle:");
    }
}
```

```

d1 = sc.nextDouble();
d2 = sc.nextDouble();
}
void printArea()
{
    System.out.println("Area of triangle : " + (d1*d2)/2);
}
class Circle extends Shape
{
    double radius;
    Circle()
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter radius of circle : ");
        radius = sc.nextDouble();
    }
    void printArea()
    {
        System.out.println("Area of circle : " +
                           (3.14 * radius * radius));
    }
}
class ShapeMain
{
    public static void main(String args[])
    {
        Rectangle r = new Rectangle();
        r.printArea();
        Triangle t = new Triangle();
        t.printArea();
        Circle c = new Circle();
        c.printArea();
    }
}

```

java — bash — 80x24

```
Ishas-MacBook-Air:~ isha$ cd Documents/java/  
Ishas-MacBook-Air:java isha$ javac lab4.java  
Ishas-MacBook-Air:java isha$ java ShapeMain  
Enter the dimensions of rectangle:  
4 5  
Area of rectangle: 20.0  
Enter the dimensions of triangle:  
4 6  
Area of triangle: 12.0  
Enter the radius of circle:  
5  
Area of circle: 78.5  
Ishas-MacBook-Air:java isha$
```

## **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 Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- Accept deposit from customer and update the balance.
- Display the balance.
- Compute and deposit interest
- Permit withdrawal and update the balance
- Check for the minimum balance, impose penalty if necessary and update the balance

## Lab5.

Acc No	/ /
Date	/ /

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

```
class Account
```

```
{
```

```
    String acc_name; // account name  
    int acc_num; // account number
```

```
    String acc_type; // type of the account  
    double balance; // initial deposit for
```

```
}
```

```
class SavAccount extends Account
```

```
{
```

```
    Scanner sc = new Scanner(System.in);  
    double cmp; // interest rate
```

```
    SavAccount()  
        cmp = 0.05; // 5%
```

```
{
```

```
    System.out.println("Enter the account name:");  
    acc_name = sc.next();
```

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

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

```
    System.out.println("Enter the balance:");
```

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

```
}
```

```
void deposit()
```

```
{
```

```
    double amt; // amount to deposit
```

```
    System.out.println("Enter the amount to deposit:");
```

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

```
    balance += amt;
```

```
}
```

void cmp(int)

{

    int time;

    System.out.println("Enter the time period elapsed");

    time = sc.nextInt();

    System.out.println("Enter the rate of C/I");

    int rate = sc.nextInt();

    cmp = balance;

    balance = balance \* (Math.pow(1 + (rate \* 0.01)), time);

    cmp = balance - cmp;

    System.out.println("Compound Interest : " + cmp);

    System.out.println("Balance after depositing interest : " + balance);

}

void withdraw()

{

    double amt;

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

    amt = sc.nextDouble();

    if(balance >= amt)

    {

        balance -= amt;

        System.out.println("Your amount after withdrawal is : " + balance);

    }

    else

    {

        System.out.println("The given amount cannot be withdrawn");

    }

}

void displ()

{

System.out.println("balance amount :" + balance);

}

Class curr\_acc extends account

{

Scanner sc = new Scanner(System.in);

double min\_bal;

double penalty;

curr\_acc();

{

System.out.println("Enter the account name :");

acc\_name = sc.nextLine();

System.out.println("Enter the account number :");

acc\_num = sc.nextInt();

System.out.println("Enter the balance :");

balance = sc.nextDouble();

}

void deposit()

{

double amt;

System.out.println("Enter the amount to deposit :");

amt = sc.nextDouble();

balance += amt;

}

void minimum()

{

System.out.println("Enter the minimum balance allowed and service charge percentage");  
 min\_bal = sc.nextDouble();  
 penalty = sc.nextDouble();

if(balance < min\_bal)

{

System.out.println("Your balance is less than minimum amount");

balance = balance - ((min\_bal - balance) \* penalty \* 0.01);

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

}

double amt;

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

amt = sc.nextDouble();

if(balance >= amt)

{

balance -= amt;

System.out.println("Your amount after withdrawal is " + balance);

}

else{

System.out.println("The given amount cannot be withdrawn");

}

void disp()

{

}

System.out.println("balance amount :" + balance);

}

class accMain

{

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter the account type  
write 1 for savings and 2 for current");

int acc\_type = sc.nextInt();

if (acc\_type == 1)

{

Sav-Acc o2 = new Sav-Acc();

while (true)

{

System.out.println("1. Deposit \n 2. Display  
Balance \n 3. Withdraw Amount \n 4. Check  
for Compound Interest");

int choice = sc.nextInt();

switch (choice)

{

case 1: o2.deposit();

break;

case 2: o2.disp();

break;

case 3: o2.withdraw();

break;

case 4: o2.comp\_int();

break;

```

        default: System.out.println("Enter choice");
        }

    }

    close if (acc_type == 2)

    curr_acc = new curr_acc();
    while (true)
    {
        System.out.println("1. Deposit \n 2. Display
balance \n 3. Withdraw Amount \n 4. Check
for minimum balance and penalty \n 5.
Exit choice = sc.nextInt();
        switch (choice)
        {
            case 1: acc.deposit();
            break;
            case 2: acc.display();
            break;
            case 3: acc.withdraw();
            break;
            case 4: acc.minimum();
            break;
            default: System.exit(0);
        }
    }
}
}

```

```
Ishas-MacBook-Air:~ isha$ cd Documents/java/
Ishas-MacBook-Air:java isha$ javac lab5.java
Ishas-MacBook-Air:java isha$ java accMain
Enter the account type, write 1 for savings and 2 for current:
1
Enter the account name:
Isha
Enter the account number:
123
Enter the balance:
2000

1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for compound interest 5.Exit
1
Enter the amount to deposit:
200

1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for compound interest 5.Exit
2
balance amount:2200.0

1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for compound interest 5.Exit
3
Enter the amount to be withdrawn:
500
Your amount after withdrawl is: 1700.0

1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for compound interest 5.Exit
4
Enter the time period elapsed:
```



java — java accMain — 62x42

Enter the time period elapsed:

4

Enter the rate of CI

8

Compound Interest:612.8312320000005

Balance after depositing interest:2312.8312320000005

1.Deposit

2.Display Balance

3.Withdraw Amount

4.Check for compound interest 5.Exit

5

Ishas-MacBook-Air:java isha\$ java accMain

Enter the account type, write 1 for savings and 2 for current:

2

Enter the account name:

Isha

Enter the account number:

456

Enter the balance:

2000

1.Deposit

2.Display Balance

3.Withdraw Amount

4.Check for minimum balance and penalty

1

Enter the amount to deposit:

600

1.Deposit

2.Display Balance

3.Withdraw Amount

4.Check for minimum balance and penalty

2

balance amount:2600.0

1.Deposit

2.Display Balance

3.Withdraw Amount

4.Check for minimum balance and penalty

3



java — java accMain — 62x42

2000

```
1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for minimum balance and penalty
```

1

Enter the amount to deposit:

600

```
1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for minimum balance and penalty
```

2

balance amount:2600.0

```
1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for minimum balance and penalty
```

3

Enter the amount to be withdrawn:

1700

Your amount after withdrawl is: 900.0

```
1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for minimum balance and penalty
```

4

Enter the minimum balance allowed and the service charge percentage

1200 5

Your balance is less than minimum amount

Your new total balance is885.0

```
1.Deposit
2.Display Balance
3.Withdraw Amount
4.Check for minimum balance and penalty
```

LAB-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.

Lab-6

ISHA SINGH

IBM 1975-057  
Section - A

In the package folder - Student.java and Internal.java  
In the package folder - External.java  
In default package - TotalMarks.java

File → Student.java

package size

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

public class Student

8 b

string name  
string type

int sem;

public Student();

Scanner sc = new Scanner(System.in);

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

Name = S.C. Pendleton

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

wn = sc.next()

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

$$\text{Sum} = \text{Sum} + \text{nextInt}$$

3

3

Chap. 11 - 11.2.1 - 11.2.2 - 11.2.3

卷之三

Copyright © 2014 Pearson Education, Inc.

—

*Experiments on the effect of temperature on the rate of absorption of oxygen by the blood.*

*1000*

Copyright © The McGraw-Hill Companies, Inc.

600 601 602 603

14th & 15th October 1908

ISHA SINGH  
13MTEC057

File → Student.java Internals.java

```
package iie;
import java.util.Scanner;
public class Student Internals extends Student
```

```
{ public double iem[] = new double[5];
    public Internals()
```

```
Scanner sc = new Scanner(System.in);
System.out.println("Enter marks:");
for(int i=0; i<5; i++)
```

```
System.out.println("Enter the subject "
    +(i+1)+" marks::");
    iem[i] = sc.nextDouble();
}
```

File → Externals.java

```
package iie;
import java.util.Scanner;
public class Externals /
```

```
{ public double seem[] = new double[5];
    public Externals()
```

```
Scanner sc = new Scanner(System.in);
for(int i=0; i<5; i++)
```

```
System.out.println("Enter the subject "+
```

*bring*

```

        (i+1) + " see marks::");
        scemli] = sc.nextDouble();
    }
}
}

```

File → TotalMarks.java

```

import i.e.*;
import see.*;
import java.util.Scanner;
public class TotalMarks
{

```

```

    public static void main(String args[])
    {

```

```

        Scanner sc = new Scanner(System.in);

```

```

        System.out.println("Enter the number of students::");
        int n = sc.nextInt();

```

```

        i.e. Internals[i] = new i.e. Internals[n];

```

```

        see.Externals sc[i] = new see.Externals[n];

```

```

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

```

        System.out.println("Student " + (i+1) + " details:");

```

```

        iel[i] = new i.e. Internals();

```

```

        sc[i] = new see.Externals();

```

```

        System.out.println("The total marks of
        student " + (i+1) + " in all 5 subjects");

```

```

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

```

        System.out.println("Total marks in subject"
            + (i+1) + ":" + (iel[i].iuelj] + sc[i].scemj));
    }
}
}

```

```
[Ishas-MacBook-Air:~ isha$ cd Desktop/lab6
[Ishas-MacBook-Air:lab6 isha$ javac TotalMarks.java
[Ishas-MacBook-Air:lab6 isha$ java TotalMarks
Enter the number of students::
2
Student 1 details::
Enter the name of student::
isha
Enter the usn of student::

1bm
Enter the sem of student::
3
Cie marks::
Enter the subject 1 marks::
45
Enter the subject 2 marks::
45
Enter the subject 3 marks::
45
Enter the subject 4 marks::
45
Enter the subject 5 marks::
45
```

```
Cie marks::
Enter the subject 1 marks::
34
Enter the subject 2 marks::
34
Enter the subject 3 marks::
34
Enter the subject 4 marks::
34
Enter the subject 5 marks::
34
Enter the subject 1 see marks::
34
Enter the subject 2 see marks::
34
Enter the subject 3 see marks::
34
Enter the subject 4 see marks::
34
Enter the subject 5 see marks::
34
The total marks of student 2 in all 5 subjects::
Total marks in subject 1 ::68.0
Total marks in subject 2 ::68.0
```

```
lab6 — -bash — 80x24
Cie marks::
Enter the subject 1 marks::
34
Enter the subject 2 marks::
34
Enter the subject 3 marks::
34
Enter the subject 4 marks::
34
Enter the subject 5 marks::
34
Enter the subject 1 see marks::
34
Enter the subject 2 see marks::
34
Enter the subject 3 see marks::
34
Enter the subject 4 see marks::
34
Enter the subject 5 see marks::
34
The total marks of student 2 in all 5 subjects::
Total marks in subject 1 ::68.0
Total marks in subject 2 ::68.0
```

```
lab6 — -bash — 80x24
34
Enter the subject 3 marks::
34
Enter the subject 4 marks::
34
Enter the subject 5 marks::
34
Enter the subject 1 see marks::
34
Enter the subject 2 see marks::
34
Enter the subject 3 see marks::
34
Enter the subject 4 see marks::
34
Enter the subject 5 see marks::
34
The total marks of student 2 in all 5 subjects::
Total marks in subject 1 ::68.0
Total marks in subject 2 ::68.0
Total marks in subject 3 ::68.0
Total marks in subject 4 ::68.0
Total marks in subject 5 ::68.0
Ishas-MacBook-Air:lab6 isha$
```

## LAB-7

Write a program to demonstrate generics with multiple object parameters.

ISHA SINGH

IBN/9E6057

DATE / /

lab-7

```
import java.util.Scanner;  
class TwoGen<T, V>
```

{

T ob1;

V ob2;

TwoGen(T ob1, V ob2)

{

ob1 = ob1;

ob2 = ob2;

}

void showTypes()

{

System.out.println("Type of Generic

T which contains the Integer is " +

ob1.getClass().getName());

System.out.println("Type of Generic V

which contains the String is " +

ob2.getClass().getName());

{

T getob1()

{ return ob1;

}

V getob2()

{ return ob2;

}

{

char Simp[10]

public static void main(String args[])

Scanner sc = new Scanner(System.in);

System.out.println("Enter integer:");

int n = sc.nextInt();

System.out.println("Enter String:");

String s = sc.nextLine();

TwoGen<Integer, String> tgObj

= new TwoGen<Integer, String>(n, s);

tgObj.showTypes();

int v = tgObj.getObj1();

System.out.println("Value of integer: " + v);

String str = tgObj.getObj2();

System.out.println("Value of String: " + str);

J

*Janah*



## Desktop — bash — 80x24

```
[Ishas-MacBook-Air:~ isha$ cd Desktop/  
[Ishas-MacBook-Air:Desktop isha$ javac Gen.java  
[Ishas-MacBook-Air:Desktop isha$ java SimpGen  
Enter an integer:  
45  
Enter a String:  
isha  
Type of Generic T which contains the integer is java.lang.Integer  
Type of Generic V which contains the String is java.lang.String  
value of integer: 45  
value of String: isha  
Ishas-MacBook-Air:Desktop isha$ ]
```

## LAB-8

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 Wrong Age( ) when the input age=father's age.

ISHA SINGH  
16MIS1057

### Lab-8

import java.util.Scanner;  
class Father

```
int fage;
Father(int age)
{
    if (age < 0)
        throw new Age();
    catch (Exception e)
```

System.out.println("Age cannot be  
negative.");

```
}  
Static void check(int a) throws Excp
```

```
if (a < 0)
    throw new Excp();
```

System.out.println("No exception in father's  
age");

```
}  
class Son extends Exception
```

```
public String toString()
```

return "Wrong Age.";

```
}
```

Jyoti

JAVA Son extends Father

? int sonAge  
Son extends Father

? Super() /  
buy

? checkIfS()

? catch(Excp e)

? System.out.println("Son's age  
cannot be greater than equal to  
father's age.");

? }

static void checkIfS(int fa, int son) throws Excp

? if (son >= fa)  
throw new Excp();

? System.out.println("No exception  
in Son's age.");

? }

JAVA lab8

? public static void main(String args[])

? Scanner sc = new Scanner(System.in);  
int fa;

? System.out.println("Enter the age of  
father and son.");

1. ~~new York~~ / ~~new York~~ / ~~new York~~ / ~~new York~~

{ = sc.nextLine();  
bs = sc.nextLine();

son db = new StringTokenizer(s);

}

Desktop — bash — 80x24

```
Ishas-MacBook-Air:~ isha$ cd Desktop/
Ishas-MacBook-Air:Desktop isha$ javac lab8.java
Ishas-MacBook-Air:Desktop isha$ java lab8
Enter the age of father and son
-45 50
Wrong age. Age cannot be negative.
Wrong age. Son's age cannot be greater than or equal to father's age.
Ishas-MacBook-Air:Desktop isha$ java lab8
Enter the age of father and son
45 50
No exception in father's age
Wrong age. Son's age cannot be greater than or equal to father's age.
Ishas-MacBook-Air:Desktop isha$ java lab8
Enter the age of father and son
45 15
No exception in father's age
No exception in son's age
Ishas-MacBook-Air:Desktop isha$
```

## LAB-9

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.

Lab-7

ISHA SINGH  
18BT15EC057

```
class NewThread implements Runnable
{
    String name;
    Thread t;
    int n;

    NewThread(String threadname, int num)
    {
        name = threadname;
        n = num;
        t = new Thread(this, name);
        System.out.println("New thread :" + t);
        t.start();
    }

    public void run()
    {
        try
        {
            for(int i=5; i>0; i--)
            {
                System.out.println(name + ":" + i);
                if(n==1)
                    Thread.sleep(10000);
                else if(n==2)
                    Thread.sleep(2000);
            }
        } catch(InterruptedException e)
        {
            e.printStackTrace();
        }
    }
}
```

4/10

System.out.println(name + " Interrupted");

System.out.println("name + " exiting.");

}

class lab9

{

public static void main(String args[])

new NewThread("BMS College of Engineering", 1);

new NewThread("CSE", 2);

}

}

*Arpit*

Desktop — bash — 80x24

```
[Ishas-MacBook-Air:~ isha$ cd Desktop/  
[Ishas-MacBook-Air:Desktop isha$ javac lab9.java  
[Ishas-MacBook-Air:Desktop isha$ java lab9  
New thread: Thread[BMS College of Engineering,5,main]  
New thread: Thread[CSE,5,main]  
BMS College of Engineering: 5  
CSE: 5  
CSE: 4  
CSE: 3  
CSE: 2  
CSE: 1  
BMS College of Engineering: 4  
CSE exiting.  
BMS College of Engineering: 3  
BMS College of Engineering: 2  
BMS College of Engineering: 1  
BMS College of Engineering exiting.  
Ishas-MacBook-Air:Desktop isha$ ]
```

## LAB-10

Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.

TSHA SINGH  
1EMTEC057

Lab-10

```
import java.awt.*;
import java.awt.event.*;
public class ooplo extends Frame implements
ActionListener
{
    Button b;
    TextField n1, n2, r;
    String s;
    public ooplo()
    {
        setLayout(new FlowLayout());
        b = new Button("Divide");
        Label num1 = new Label("Num1 : ", Label.RIGHT);
        n1 = new TextField(10);
        add(num1);
        add(n1);
        Label num2 = new Label("Num2 : ", Label.RIGHT);
        n2 = new TextField(10);
        add(num2);
        add(n2);
        Label res = new Label("Result : ", Label.RIGHT);
        r = new TextField(20);
        add(res);
        add(r);
        add(b);
        b.addActionListener(this);
        addWindowListener(new MyWindowAdapter());
    }
}
```

public void actionPerformed(ActionEvent e)

int i = 0, j = 0;  
try

i = Integer.parseInt(tn1.getText());  
j = Integer.parseInt(tn2.getText());

catch(NumberFormatException e)

}

s = "Number Format Exception";

rr.setText(" "));

DialogBox db = new DialogBox

(this, "Dialog", s);

db.setVisible(true);

return;

)

buy

if(j == 0)

throw new ArithmeticException();

double q = (double)i/j;

s = Double.toString(q);

rr.setText(s);

)

catch(ArithmeticException)

catch(ArithmaticException e)

{

s = "Arithmatic Exception";

rr.setText(" "));

```

DialogBox db = new DialogBox(this, "Dialog", s)
db.setVisible(true)

}

public static void main(String args[])
{
    Applet appwin = new Applet()
    appwin.setSize(new Dimension(380, 180))
    appwin.setTitle("Dissable")
    appwin.setVisible(true)
}

class MyWindowAdapter extends WindowAdapter
{
    public void windowClosing(WindowEvent we)
    {
        System.exit(0)
    }
}

class DialogBox extends Dialog
{
    DialogBox(Frame parent, String title, String s)
    {
        super(parent, title, false)
        setLayout(new FlowLayout())
        setSize(300, 200)
        add(new Label(s))
        Button b
        b = new Button("OK")
        b.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e)
            {
                dispose()
            }
        })
    }
}

```

Page No.	
Date	/ /

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent we)

{

dispers();

}

}.

}/

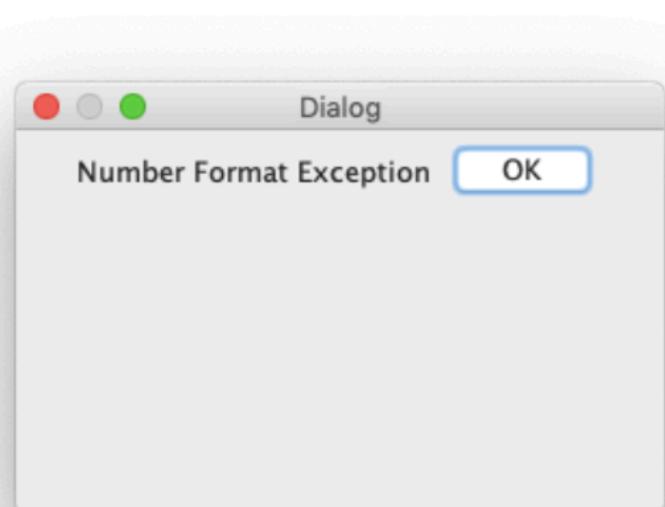
}

Division

Num1:  Num2:  Result:

Division

Num1:  Num2:  Result:



Division

Num1:  Num2:  Result:

