

JAVA LAB PROGRAMS

- 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 quad

{
    public static void main(String xx[])
    {
        double a,b,c,r1,r2,d;
        Scanner s1=new Scanner(System.in);
        System.out.println("Enter the values of a,b,c");
        a=s1.nextDouble();
        b=s1.nextDouble();
        c=s1.nextDouble();
        d=b*b-4*a*c;
        System.out.println("The value of d is:"+d);
        if(d>0)
        {
            System.out.println("The roots are real");
            r1=(-b+Math.sqrt(d))/(2*a);
            r2=(-b-Math.sqrt(d))/(2*a);
            System.out.println("First root is:"+r1);
            System.out.println("Second root is:"+r2);
        }
        else if(d<0)
        {
            System.out.println("The roots are not real");
            r1=-b/(2*a);
            r2=Math.sqrt(-d)/(2*a);
            System.out.println("First root is:"+r1);
            System.out.println("Second root is:"+r2);
        }
    }
}
```

```

    }
else
{
    System.out.println("The roots are equal");
    r1=r2=-b/(2*a);
    System.out.println("The roots are:"+r1+" "+r2);
}
}
}
}

```

```

C:\Users\Lakshitha.L\Desktop>java lab>javac quad.java
C:\Users\Lakshitha.L\Desktop>java lab>java quad
Enter the values of a,b,c
1
-4
4
The value of d is:0.0
The roots are equal
The roots are:2.0 2.0

C:\Users\Lakshitha.L\Desktop>java lab>java quad
Enter the values of a,b,c
1
-7
12
The value of d is:1.0
The roots are real
First root is:4.0
Second root is:3.0

C:\Users\Lakshitha.L\Desktop>java lab>java quad
Enter the values of a,b,c
1
4
5
The value of d is:-4.0
The roots are not real
First root is:-2.0
Second root is:1.0

```

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.

```
import java.util.Scanner;

class Studentsgpa{

int USN;

String Name;

int Credits[] = new int[5];

int Marks[] = new int[5];

Scanner s1 = new Scanner(System.in);

void accept()

{

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

USN = s1.nextInt();

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

Name = s1.next();

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

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

{

Credits[i] = s1.nextInt();

}

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

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

{

Marks[i] = s1.nextInt();

}

}

void display()

{

System.out.println(" The usn and name is:" + USN + " " + Name);

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

{



}
```

```
System.out.println("Marks is"+Marks[i]);  
}  
  
int Creditscal(int Marks)  
{  
    if(Marks>=90)  
    {  
        return 10;  
    }  
    else if(Marks>=80)  
    {  
        return 9;  
    }  
    else if(Marks>=70)  
    {  
        return 8;  
    }  
    else if(Marks>=60)  
    {  
        return 7;}  
    else if(Marks>=50)  
    {  
        return 6;}  
    else  
    {  
        return 0;}  
}  
  
double sgpacal()  
{  
    double totalcredits=0.0,totalmarks=0.0;  
    for(int i=0;i<5;i++)  
    {
```

```

totalcredits+=Credits[i];
totalmarks+=Creditscal(Marks[i])*Credits[i];
}
return totalmarks/totalcredits;
}}
class Sgpacal
{
public static void main(String xx[])
{
Studentsgpa a1=new Studentsgpa();
a1.accept();
a1.display();
double sgpa=a1.sgpacal();
System.out.println("SGPA is:"+sgpa);
}
}

```

Output:

```

C:\Users\Lakshitha.L\Desktop>java lab>javac Sgpacal.java
C:\Users\Lakshitha.L\Desktop>java lab>java Sgpacal
Enter the usn:
1
Enter the name:
ABC
Enter the credits:
4
4
4
4
4
Enter the marks:
99
98
97
90
99
The usn and name is:1 ABC
Marks is99
Marks is98
Marks is97
Marks is90
Marks is99
SGPA is:10.0

```

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.

```
import java.util.Scanner;

class Book

{
    String name;
    String author;
    double price;
    int num_pages;

    Scanner s1=new Scanner(System.in);

    Book(String name, String author, double price, int num_pages)
    {
        this.name=name;
        this.author=author;
        this.price=price;
        this.num_pages=num_pages;
    }

    void display()
    {
        System.out.println(" the name of the book:"+name);
        System.out.println(" the name of the author:"+author);
        System.out.println(" the price of the book:"+price);
        System.out.println(" the num_pages of the book:"+num_pages);
    }

    void accept()
    {
        System.out.println("Enter the name of the book:");
        name=s1.next();
    }
}
```

```
System.out.println("Enter the name of the author:");
author=s1.next();

System.out.println("Enter the price of the book:");
price=s1.nextDouble();

System.out.println("Enter the num_pages of the book:");
num_pages=s1.nextInt();

}

}

class bookarray
{
public static void main(String xx[])
{
Scanner s1=new Scanner(System.in);

int i;int n;

System.out.println("Enter the number of books:");
n=s1.nextInt();

Book a[]=new Book[n];

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

a[i]= new Book("a","c",12.9,23);
a[i].accept();

System.out.println("The details of the book is");
a[i].display();
}
}
}
```

Output:

```
C:\Users\Lakshitha.L\Desktop>java bookarray
Enter the number of books:
2
Enter the name of the book:
abc
Enter the name of the author:
xyz
Enter the price of the book:
90
Enter the num_pages of the book:
100
The details of the book isBook name: abc author: xyz price: 90.0 numpages: 100
Enter the name of the book:
bbb
Enter the name of the author:
ccc
Enter the price of the book:
99
Enter the num_pages of the book:
100
The details of the book isBook name: bbb author: ccc price: 99.0 numpages: 100
```

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.

```
abstract class Shape
{
    int num1;
    int num2;
    abstract void printarea();
    Shape(int num1,int num2)
    {
        this.num1=num1;
        this.num2=num2;
    }
}

class rectangle extends Shape
{
    rectangle(int x, int y)
    {
        super(x,y);
    }
    void printarea(){
        int result=num1*num2;
        System.out.println("The area is:"+result);
    }
}

class triangle extends Shape
{
    triangle(int x, int y)
    {
        super(x,y);
    }
}
```

```
void printarea(){
    double result=num1*num2*0.5;
    System.out.println("The area is:"+result);
}
}

class circle extends Shape
{
circle(int x)
{
super(x,x);
}

void printarea(){
    double result=num1*num2*Math.PI;
    System.out.println("The area is:"+result);
}
}

class areacalculation
{
public static void main(String xx[])
{
rectangle r1=new rectangle(10,20);
r1.printarea();
triangle t1=new triangle(2,4);
t1.printarea();
circle c1=new circle(2);
c1.printarea();
}
}
```

Output:

```
C:\Users\Lakshitha.L\Desktop\java lab>javac areacalculation.java
C:\Users\Lakshitha.L\Desktop\java lab>java areacalculation
The area is:200
The area is:4.0
The area is:12.566370614359172
```

/*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.*/

```
import java.util.Scanner;

class Bank

{

    float balance;

}

class Account extends Bank

{

    String cus_name,acc_type;int acc_no;

    Account(String name,int acc_no,String acc_type)

    {

        this.cus_name=name;this.acc_no=acc_no;this.acc_type=acc_type;balance=0.0f;

    }

    void setBal(float amt)

    {

        balance+=amt;

    }

    void disBal(){

        System.out.println("Balance:Rs "+this.balance);

    }

}
```

```
    }

}

class Sav_acct extends Account
{
    float comp_int,withdrawal;

    Sav_acct(String name,int acc_no,String acc_type)
    {
        super(name,acc_no,acc_type);
    }

    void inter(float rate)
    {
        System.out.println("Interest:"+ (balance*rate/100));
        balance+=(balance*rate/100);
    }
}

class Cur_acct extends Account
{
    float comp_int,withdrawal;

    Cur_acct(String name,int acc_no,String acc_type)
    {
        super(name,acc_no,acc_type);
        balance=0.0f;
    }

    void with(float amt){
        if(amt<balance)
            balance-=amt;
        else
            System.out.println("Amount exceeds balance!");
        if(balance<500)
```

```

    {
        System.out.println("No minimum balance maintained!Rs.500 fine levied.");
        balance-=500;
    }
}

class LabQ5
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        int k=0,j=0;int ch;boolean t=true,t1=true;
        System.out.println("Enter the number of customers:");
        int n=sc.nextInt();
        Sav_acct a[]=new Sav_acct[n];Cur_acct b[]=new Cur_acct[n];

        for(int i=0;i<n;i++)
        {
            System.out.println("Enter the name, Account type and Account number of user" +(i+1)+":");
            String name=sc.next();
            String acc_type=sc.next();
            int acc_no=sc.nextInt();

            if(acc_type.equalsIgnoreCase("savings"))
            {
                a[k]=new Sav_acct(name,acc_no,acc_type);
                while(t)
                {
                    System.out.println("Enter 1.Update balance.\n2.Display balance.\n3.Compute and
deposit interest.\n4.End");

```

```

ch=sc.nextInt();

switch(ch){

case 1:System.out.println("Enter the deposit amount:");

a[k].setBal(sc.nextFloat());

break;

case 2:

a[k].disBal();

break;

case 3:System.out.println("Enter the rate of interest:");

a[k].inter(sc.nextFloat());

break;

case 4:t=false;break;

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

}

}

k++;

}

if(acc_type.equalsIgnoreCase("current"))

{

b[j]=new Cur_acct(name,acc_no,acc_type);

while(t1)

{

System.out.println("Enter 1.Update balance.\n2.Display balance.\n3.withdraw and

update balance.\n 4.End");

ch=sc.nextInt();

switch(ch){

case 1:System.out.println("Enter the deposit amount:");

b[j].setBal(sc.nextFloat());

break;

case 2:

```

```
b[j].disBal();
break;
case 3:System.out.println("Enter the withdraw amount:");
b[j].with(sc.nextFloat());
break;
case 4:t1=false;break;
default:System.out.println("Invalid choice");
}
}j++;
}}}}
```

OUTPUT:

```
Enter the number of customers:  
2  
Enter the name, Account type and Account number of user1:  
A  
Savings  
1  
Enter 1.Update balance.  
2.Display balance.  
3.Compute and deposit interest.  
4.End  
1  
Enter the deposit amount:  
1000  
Enter 1.Update balance.  
2.Display balance.  
3.Compute and deposit interest.  
4.End  
2  
Balance:Rs 1000.0  
Enter 1.Update balance.  
2.Display balance.  
3.Compute and deposit interest.  
4.End  
3  
Enter the rate of interest:  
2  
Interest:20.0  
Enter 1.Update balance.  
2.Display balance.  
3.Compute and deposit interest.  
4.End  
4.End  
2  
Balance:Rs 1020.0  
Enter 1.Update balance.  
2.Display balance.  
3.Compute and deposit interest.  
4.End  
4  
Enter the name, Account type and Account number of user2:  
B  
Current  
2  
Enter 1.Update balance.  
2.Display balance.  
3.withdraw and update balance.  
4.End  
1  
Enter the deposit amount:  
1500  
Enter 1.Update balance.  
2.Display balance.  
3.withdraw and update balance.  
4.End  
2  
Balance:Rs 1500.0  
Enter 1.Update balance.  
2.Display balance.  
3.withdraw and update balance.  
4.End  
3  
Enter the withdraw amount:  
120  
Enter 1.Update balance.  
2.Display balance.  
3.withdraw and update balance.  
4.End  
2  
Balance:Rs 1380.0  
Enter 1.Update balance.  
2.Display balance.  
3.withdraw and update balance.  
4.End  
4
```

```
6) package CIE;

import java.util.Scanner;

public class Student

{

    int usn;

    String name;

    int sem;

    Scanner s1=new Scanner(System.in);

    public void accept()

    {

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

        usn=s1.nextInt();

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

        name=s1.next();

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

        sem=s1.nextInt();

    }

    public void display()

    {

        System.out.println("Usn:"+ " " + usn);

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

        System.out.println("Sem:"+ " " +sem);

    }

}

package CIE;

import java.util.Scanner;

public class Internals extends Student

{

    Scanner s1=new Scanner(System.in);
```

```
public int lmarks[] = new int[5];
public void accept()
{
    System.out.println("Enter the marks scored in 5 subjects");
    for(int i=0;i<5;i++)
    {
        lmarks[i]=s1.nextInt();
    }
}
public void display()
{
    for(int i=0;i<5;i++)
    {
        System.out.println("The marks scored in "+(i+1)+"subject is"+lmarks[i]);
    }
}
package CIE;
import java.util.Scanner;

public class Internals extends Student
{
    Scanner s1=new Scanner(System.in);
    public int lmarks[] = new int[5];
    public void accept()
    {
        System.out.println("Enter the marks scored in 5 subjects");
        for(int i=0;i<5;i++)
        {
            lmarks[i]=s1.nextInt();
        }
    }
    public void display()
    {
        for(int i=0;i<5;i++)
        {
            System.out.println("The marks scored in "+(i+1)+"subject is"+lmarks[i]);
        }
    }
}
```

```
}}}

import CIE.Student;
import CIE.Internals;
import SEE.Externals;
import java.util.Scanner;

public class Final

{
    public static void main(String xx[])
    {
        Scanner S1=new Scanner(System.in);

        int n;

        System.out.println("Enter the value of n");

        n=S1.nextInt();

        CIE.Student S[] = new CIE.Student[n];
        CIE.Internals I[] = new CIE.Internals[n];
        SEE.Externals E[] = new SEE.Externals[n];

        for (int i=0;i<n;i++)
        {
            S[i]=new Student();
            I[i]=new Internals();
            E[i]=new Externals();

            S[i].accept();
            I[i].accept();
            E[i].accept();

            S[i].display();
            I[i].display();
            E[i].display();

            for (int j=0;j<5;j++)
            {
                finalmarks=I[i].Imarks[j]+E[i].EMarks[j];
            }
            System.out.println("Marks in :"+(i+1)+"subject is"+finalmarks);
        }
    }
}
```

```
C:\Users\Lakshitha.L\Desktop\Hat>java Final
Enter the value of n
2
Enter the usn:
1
Enter the name:
A
Enter the sem:
3
Enter the markes scored in 5 subjects
100
90
90
100
100
Enter the markes scored in 5 subjects
100
90
100
99
100
Usn: 1
Name: A
Sem: 3
The marks scored in 1subject is100
The marks scored in 2subject is90
The marks scored in 3subject is90
The marks scored in 4subject is100
The marks scored in 5subject is100
The marks scored in 1subject is100
The marks scored in 2subject is90
The marks scored in 3subject is100
The marks scored in 4subject is99
The marks scored in 5subject is100
Marks in :1subject is200
Marks in :1subject is180
Marks in :1subject is190
Marks in :1subject is199
Marks in :1subject is200
Enter the usn:
```

```
Enter the usn:
2
Enter the name:
B
Enter the sem:
3
Enter the markes scored in 5 subjects
100
99
100
100
100
Enter the markes scored in 5 subjects
100
90
99
99
100
Usn: 2
Name: B
Sem: 3
The marks scored in 1subject is100
The marks scored in 2subject is99
The marks scored in 3subject is100
The marks scored in 4subject is100
The marks scored in 5subject is100
The marks scored in 1subject is100
The marks scored in 2subject is90
The marks scored in 3subject is99
The marks scored in 4subject is99
The marks scored in 5subject is100
Marks in :2subject is200
Marks in :2subject is189
Marks in :2subject is199
Marks in :2subject is199
Marks in :2subject is200
```

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=father's age.

```
import java.util.Scanner;

class WrongAgeException extends Exception {

    public WrongAgeException(String message) {
        super(message);
        System.out.println("Error");
    }
}

class Father {

    int age;

    Father(int age) throws WrongAgeException {
        this.age = age;
        if (age < 0) {
            throw new WrongAgeException("Age cannot be negative");
        }
    }
}

class Son extends Father {

    int sonage;

    Son(int age, int sonage) throws WrongAgeException {
        super(age);
        this.sonage = sonage;
        if (sonage >= age) {
            throw new WrongAgeException("Son's age should be less than father's age");
        }
    }
}
```

```

    }

}

}

class excep {
    public static void main(String[] args) {
        Scanner s1=new Scanner(System.in);
        int age,sonage;
        try {
            System.out.println("Enter f age");
            age=s1.nextInt();
            System.out.println("Enter s age");
            sonage=s1.nextInt();
            Father father = new Father(40);
            Son son = new Son(4, 20);
        } catch (WrongAgeException e) {
            System.out.println("Exception occurred: " + e.getMessage());
        }
    }
}

```

```

C:\Users\Lakshitha.L\Desktop>javac excep.java
C:\Users\Lakshitha.L\Desktop>java excep
Enter f age
35
Enter s age
36
Error
Exception occurred: Son's age should be less than father's age
C:\Users\Lakshitha.L\Desktop>

```

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.

```
class thread1 extends Thread
{
    public void run()
    {
        while(true)
        {System.out.println("BMS College of Engineering");
        try
        {
            Thread.sleep(10000);
        }
        catch(InterruptedException e)
        {}
    }
}

class thread2 extends Thread
{
    public void run()
    {
        while(true){
            System.out.println("CSE");
            try
            {
                Thread.sleep(2000);
            }
            catch(InterruptedException e)
            {}
        }
    }
}
```

```
class Multithread
{
    public static void main(String xx[])
    {
        thread1 t1=new thread1();
        thread2 t2=new thread2();
        t1.start();
        t2.start();
    }
}
```

Output:

```
C:\Users\Lakshitha.L\Desktop>javac Multithread.java
C:\Users\Lakshitha.L\Desktop>java Multithread
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
```

9) 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.

```
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class ExceptionDemo extends JFrame implements ActionListener {
    private JTextField t1, t2, t3;
    private JLabel l1, l2;
    private JButton b1, b2;
    public ExceptionDemo() {
        setLayout(new FlowLayout());
        l1 = new JLabel("Num1 :");
        add(l1);
        t1 = new JTextField(5);
        add(t1);
        l2 = new JLabel("Num2 :");
        add(l2);
        t2 = new JTextField(5);
        add(t2);
        t3 = new JTextField(5);
        t3.setEditable(false);
        add(t3);
        b1 = new JButton("Divide");
        add(b1);
        b1.addActionListener(this);
        b2 = new JButton("Clear");
        add(b2);
        b2.addActionListener(this);
```

```

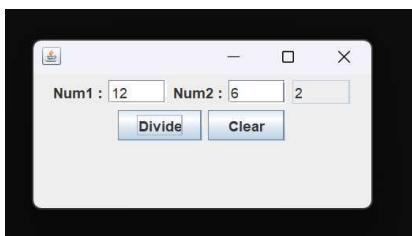
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 150); // Set an appropriate size
        setVisible(true); }

public void actionPerformed(ActionEvent act) {
    String str = act.getActionCommand();
    if (str.equals("Divide")) {
        try {
            int num1 = Integer.parseInt(t1.getText());
            int num2 = Integer.parseInt(t2.getText());
            int num3 = num1 / num2;
            t3.setText(num3);
        } catch (ArithmaticException e) {
            JOptionPane.showMessageDialog(this, "ArithmaticException: Cannot divide by zero!");
        } catch (NumberFormatException e) {
            JOptionPane.showMessageDialog(this, "NumberFormatException: Please enter valid
integers for Num1 and Num2." }
        } else {
            t1.setText("");
            t2.setText("");
            t3.setText(""); }
    }

public static void main(String[] args) {
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new ExceptionDemo();
        }});}}

```

OUTPUT:



18/12/23 Q. Develop a java program that prints all real solution to Q.E
 $ax^2 + bx + c = 0$.

Quadratic Equation:

```
import java.util.Scanner;  
class Quad  
{  
    public static void main (String xx[])  
    {  
        double a,b,c,d, r1, r2;  
        Scanner s1 = new Scanner (System.in);  
        a = s1.nextDouble();  
        b = s1.nextDouble();  
        c = s1.nextDouble();  
        if (a==0||b==0||c==0)  
        {  
            System.out.println ("Invalid Input");  
        }  
        else  
        {  
            d = b*b - 4*a*c;  
            if (d>0)  
            {  
                System.out.println ("Roots are real and distinct");  
                r1 = (-b+Math.sqrt(d))/(2*a);  
                r2 = (-b-Math.sqrt(d))/(2*a);  
                System.out.println ("r1 = " + r1 + " and " + "r2 = " + r2);  
            }  
            if (d<0)  
            {  
                System.out.println ("Roots are imaginary and distinct");  
                r1 = -b/(2*a);  
                r2 = Math.sqrt(-d)/(2*a);  
                System.out.println ("r1 = " + r1 + " and " + "r2 = " + r2);  
            }  
            else  
            {  
                System.out.println ("Roots are equal");  
                r1 = r2 = -b/(2*a);  
                System.out.println ("r1 = " + r1 + " and " + "r2 = " + r1);  
            }  
        }  
    }  
}
```

Outputs:

$$1 - 4 \quad 4$$

Roots are real and equal

$$g_{11} = g_{12} = 2.0$$

$$1 - 7 \quad 12$$

Roots are real and distinct.

$$g_{11} = 4.0 \text{ and } g_{12} = 3.0$$

$$1 \quad 4 \quad 5$$

Roots are imaginary.

$$g_{21}$$

1/1/24 Lab-2

Q. Develop a Java program to calculate the SGPA of a student accepting the USN, name, credits as array and marks as array. Use appropriate methods.

Class Student

```
{ int USN;
    String name;
    int Credits[] = new Int [len];
    int Marks[] = new Int [len];
    int len;
    Scanner S1 = new Scanner ();
```

void Accept()

```
{ USN = S1.nextInt(); → S.O.P ("Enter the USN and name: ");
    name = S1.nextLine();
    System.out.println ("Enter the subject details:");
    len = S1.nextInt();
    for (int i=0; i<len; i++)
        { System.out.println ("Enter the credits of each subject");
            credit[i] = S1.nextInt();
            System.out.println ("Enter the marks of each subject:");
            Marks[i] = S1.nextInt();
        }
```

Void Display()

```
{ for (int i=0; i<len; i++)
    System.out.println ("USN :" + USN + "Name :" + name);
    System.out.println ("Marks is / SGPA is :" + sgpa);
    Marks[i] }
```

```
int MarksCalculation (Marks[])
{
    if (Marks[0] >= 90) {
        return 10;
    }
    else if (Marks >= 80) {
        return 9;
    }
}
```

```

        else if (Marks >= 70)
        {
            return 8;
        }
        else if (Marks >= 60)
        {
            return 7;
        }
        else if (Marks >= 50)
        {
            return 6;
        }
        else
        {
            return 0;
        }
    }

double void Calculation()
{
    double totalredits = 0; totalgradepointsum = 0; float sgpa;
    for (int i = 0; i < len; i++)
    {
        totalredits += credits[i];
        totalgradepointsum += MarksCalculation[Marks[i]] * credits[i];
    }
    return sgpa = totalgradepointsum / totalredits;
}

class Class SGPA Cal
{
    public static void main (String xx[])
    {
        Student a1 = newStudent();
        a1.Accept();
        a1.Display(); double sgpa = a1.Calculation();
        System.out.println ("SGPA is " + sgpa);
    }
}

```

X 11/24

```

        else if (Marks >= 70)
        {
            return 8;
        }
        else if (Marks >= 60)
        {
            return 7;
        }
        else if (Marks >= 50)
        {
            return 6;
        }
        else
        {
            return 0;
        }
    }

double void Calculation()
{
    double totalredits = 0; totalgradepointsum = 0; float sgpa;
    for (int i = 0; i < len; i++)
    {
        totalredits += credits[i];
        totalgradepointsum += MarksCalculation[Marks[i]] * credits[i];
    }
    return sgpa = totalgradepointsum / totalredits;
}

class Class SGPA Cal
{
    public static void main (String xx[])
    {
        Student a1 = newStudent();
        a1.Accept();
        a1.Display(); double sgpa = a1.Calculation();
        System.out.println ("SGPA is " + sgpa);
    }
}

```

X 11/24

Q) `import java.util.Scanner;`
 Class Book
 {
 String name;
 String author;
 double price;
 int num-page;
 Book(String name, String author, double price, int num-page)
 {
 this.name = name;
 this.author = author;
 this.price = price;
 this.num-page = num-page;
 }
 public String toString()
 {
 return "Details of book:\n" + "Name of the book\n" + name
 + "Author\n" + author + "Price\n" + price + "num-page"
 + num-page;
 }
 void accept()
 {
 int n;
 Scanner s1 = new Scanner(System.in);
 n = s1.nextInt();
 Book a[] = new Book[n];
 name = s1.next();
 System.out.println("Enter the name of the book:");
 author = s1.next();
 System.out.println("Enter the author name");
 price = s1.nextDouble();
 System.out.println("Enter the price");
 num-pages = s1.nextInt();
 a[i] = new Book(name, author, price, num-page);
 for (int i=0; i<n; i++)
 }

```
{ System.out.println(a[i].toString());  
}  
}  
}  
}
```

Class BookDetails

```
{ public static void main (String xx[])  
{  
    Book a1 = new Book();  
    a1.get  
    a1.accept();  
}
```

Output:

Enter no of books

1

Enter the name of the book

ABC

Enter the author name

X Y Z

Enter the price

599

Enter the num.pages

100

Details of book :

Name of the book ABC

Author X Y Z

Price 599

num.pages 100

Q) 8/1/24

Develop a java program to create abstract class named Shape that contains two integers and an empty method named printArea(). Provide 3 subclasses Rectangle, triangle and circle such that classes extend the class Shape.

abstract class Shape

```
{ int num1;  
    int num2;  
    abstract void printArea();  
}
```

class Rectangle extends Shape

```
{  
    rectangle (int x, int y)  
    {  
        num1 = x;  
        num2 = y;  
    }  
    void printArea()  
    {  
        int result = num1 * num2;  
        System.out.println("The area is :" + result);  
    }  
}
```

class triangle extends Shape

```
{  
    triangle (int x, int y)  
    {  
        num1 = x;  
        num2 = y;  
    }  
    void printArea()  
    {  
        int result = 0.5 * num1 * num2;  
        System.out.println("The area is :" + result);  
    }  
}
```

Class Circle extends Shape

```
{ circle(int x)
```

```
{ num1 = x;
```

```
}
```

```
void printarea()
```

```
{ double result = num1 * num1 * Math.PI;
```

```
System.out.println("The area is "+result);
```

```
}
```

```
}
```

Class Area

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

```
{ rectangle r1 = new rectangle(2,3);
```

```
r1.printarea();
```

```
triangle t1 = new triangle(2,3);
```

```
t1.printarea();
```

```
circle c1 = new circle(2);
```

```
c1.printarea();
```

```
}
```

Output:

The area of rectangle is : 6

The area of triangle is : 3.0

The area of circle is 12.566370614

✓ 8/11/24

Q) Create a package CIE which has two classes - Student and Internals. The class Personal has members USN, name, sem. Class Internals stores internal marks of five courses. Create another package SEE which has the class Externals derived from class of Student. This class stores the SEE marks in five courses. Import the two packages in a file that declare the final marks of n students in all five courses.

```
import java.util.Scanner;  
Package CIE;  
Public class Student  
{  
    int USN;  
    String name;  
    int Sem;  
    Scanner s1 = new Scanner(System.in);  
    Public void accept()  
    {  
        System.out.println("Enter USN:");  
        USN = s1.nextInt();  
        System.out.println("Enter name:");  
        name = s1.next();  
        System.out.println("Enter Sem:");  
        Sem = s1.nextInt();  
    }  
    Public void display()  
    {  
        System.out.println("Name is "+name);  
        System.out.println("USN is "+USN);  
        System.out.println("Sem is "+Sem);  
    }  
}  
Package CIE;  
Public class Internals extends Student  
{  
    int Marks = new int[5];  
    System.out.println("Enter the marks in 5 courses.");
```

```
public void accept()
{
    for (int i=0; i<5; i++)
    {
        M[i] = s1.nextInt();
        System.out.println(" Marks scored in " + i+1 + " subject is : " + M[i]);
    }
}
```

```
public void display()
{
    System.out.println(" Internal Details ");
    for (int i=0; i<n; i++)
    {
        System.out.println(" Marks in " + i+1 + " Subject is : " + M[i]);
    }
}
```

Package SEE;

```
import CIE.Student;
import java public class External extends Student
{
    int seemarks [] = new Seemarks [5];
    public void accept()
    {
        super.accept();
        System.out.println(" Enter the marks in 5 subjects : ");
        for (int i=0; i<5; i++)
        {
            System.out.println(" Marks in " + i+1 + " Subject is : ");
            seemarks [i] = s1.nextInt();
        }
    }
}
```

```
public void display()
{
    Super.display();
    System.out.println(" SEE Marks : ");
    for (int i=0; i<5; i++)
    {
        System.out.println(" Marks in " + i+1 + " Subject is : " + seemarks[i]);
    }
}
```

```
{ System.out.println("The marks in " + i + " Subject is " + marks[i]);  
}  
}  
}  
}
```

```
import CIE.Student;  
import CIE.Internals;  
import SEE.Externals;  
  
public class main  
{  
    public static void main (String xx[])  
    {  
        int n;  
        CIE.Student s[] = new Student [n];  
        System.out.println ("Enter the value of n:");  
        n = sc.nextInt();
```

```
        for (int i=0; i<n; i++)  
        {  
            s[i] = new CIE.Student();  
            s[i].accept();  
            s[i].display();  
        }
```

```
CIE.Internals I[] = new CIE.Internals [5];  
SEE.Externals E[] = new SEE.Externals [5];
```

```
for (int i=0; i<5; i++)  
{  
    I[i] = new CIE.Internals();  
    E[i] = new SEE.Externals();  
    I[i].accept();  
    E[i].accept();  
    I[i].display();  
    E[i].display();  
}
```

```
for (int j=0; j<5; j++)  
{  
    double result = I[i]
```

```

import CIE.Student;
import CIE.Internals;
import SEE.Externals;
import java.util.Scanner;
public class Final
{
    public static void main (String xx[])
    {
        Scanner s1 = new Scanner (System.in);
        int n;
        System.out.println ("Enter the value of n");
        n = s1.nextInt();
        CIE.Student S[] = new CIE.Student[n];
        CIE.Internals I[] = new CIE.Internals[n];
        SEE.Externals E[] = new SEE.Externals[n];
        for (int i=0; i<n; i++)
        {
            S[i] = new CIE.Student();
            I[i] = new CIE.Internals();
            E[i] = new SEE.Externals();
            S[i].accept();
            I[i].accept();
            E[i].accept();
            S[i].display();
            I[i].display();
            E[i].display();
        }
        for (int j=0; j<5; j++)
        {
            int finalmarks = I[j].IMarks[j] + E[j].EMarks[j];
            System.out.println ("Marks in "+(j+1)+" subjects is "+finalmarks);
        }
    }
}

```

22/1/2024

Output:

Enter the value of n:

2

Enter the USN:

1

Enter the Name:

ABC

Enter the Sem:

2

Enter the marks in 5 subjects

Marks in 1 subject

100

100

99

77

88

Enter the marks in 5 subjects

Marks in 1 subject

100

100

99

77

88

Final Marks

Marks in 1 subject 200

Marks in 2 subject 200

Marks in 3 subject 198

Marks in 4 subject 154

Marks in 5 subject 176.

LAB - Program - 5

```
5) import java.util.Scanner;  
class Bank  
{ float balance;  
}  
  
class Account extends Bank  
{ String cus-name, acc-type; int acc-no;  
Account (String name, int acc-no, String acc-type)  
{ this.cus-name = name;  
this.acc-no = acc-no;  
this.acc-type = acc-type;  
balance = 0;  
}  
void setBal (float amt)  
{ balance += amt; }  
void disBal ()  
{ System.out.println("Balance: Rs." + this.balance);  
}  
  
class sav-acct extends Account  
{ float comp-int, withdrawal;  
sav-acct (String name, int acc-no, String acc-type)  
{ super (name, acc-no, acc-type);  
}  
void inter (float rate)  
{ System.out.println("Interest: " + (balance * rate / 100));  
balance += (balance * rate / 100);  
}  
}
```

```

class cur-acct extends Account
{
    float comp-int, withdrawal;
    cur-acct (String name, int acc-no, String acc-type)
    {
        super (name, acc-no, acc-type)
        balance = 0.0f;
    }
    void withdraw (float amt)
    {
        if (amt < balance)
            balance -= amt;
        else
            System.out.println ("Amount exceeds balance!");
        if (balance < 500)
            System.out.println ("No minimum balance maintained!
RS. 500. fine levied");
        balance -= 500;
    }
}

```

```

}

class main
{
    public static void main (String args[])
    {
        Scanner sc = new Scanner (System.in);
        int k=0, j=0, int ch, boolean t = true;
        System.out.println ("Enter the number of customers:");
        int n = sc.nextInt();
        Sav-acct a[] = new Sav-acct [n];
        cur-acct b[] = new cur-acct [n];
        for (int i=0; i<n; i++)
        {
            System.out.println ("Enter the name, Account-type
and AccountNumber:");
            String name = sc.next();
            int acc-no, string acc-type = sc.next();
            int acc-no = sc.nextInt();
        }
    }
}

```

```
if (acc-type.equalsIgnoreCase("saving"))
{
    a[k] = new Sav-Acc(name, acc-no, acc-type);
    while(t)
    {
        System.out.println("Enter 1. Update balance\n 2. Display balance\n 3. Compute and deposit interest\n 4. End");
        ch = sc.nextInt();
        switch(ch)
        {
            case 1: System.out.println("Enter the deposit amount:");
                      a[k].setBal(sc.nextFloat());
                      break;
            case 2: a[k].disBal();
                      break;
            case 3: System.out.println("Enter the rate of interest:");
                      a[k].inter(sc.nextFloat());
                      break;
            case 4: t = false;
                      break;
            default: System.out.println("Invalid choice");
        }
    }
    k++;
}
t = true;
if (acc-type.equalsIgnoreCase("current"))
{
    b[j] = new Cur-Acc(name, acc-no, acc-type);
    while(t)
    {
        System.out.println("Enter 1. Update balance\n 2. Display balance\n 3. withdraw and update balance\n 4. End");
        ch = sc.nextInt();
        switch(ch)
        {
            case 1: System.out.println("Enter deposit amount:");
                      b[j].setBal(sc.nextFloat()); break;
            case 2: b[j].disBal();
                      break;
        }
    }
}
```

```
1 Case 3: System.out.println("Enter the withdraw amount");  
2 b[j].with(sc.nextFloat());  
3 break;  
4 Case 4: t = false; break;  
5 default: System.out.println("Invalid choice.");  
6 }  
7 j++;  
8 t = true;  
9 }  
10 }
```

E Output:

1 Enter the number of customers:
2

Enter the name, Acc type and Acc. num. of user

A

savings

123

Enter 1. Update balance

2. Display balance

3. Compute and deposit interest

4. End

1

Enter the deposit amount:

1200.

Enter 1. Update balance

2. Display balance

3. Compute and deposit interest

4. End

3

Enter the rate of interest:

8

Interest : 96.0

Enter 1. Update balance

2. Display balance

3. Compute and deposit interest

4. End

2

Balance : Rs. 1296.0

Enter 1. Update balance

2. Display balance

3. Compute and deposit interest

4. End

4

Enter the name, Acc type and Acc - number of user 2 :

B

current

234.

Enter 1. update balance

2. Display balance

3. withdraw and update balance

4. End

1.

Enter the deposit amount :

1200

Enter 1. Update balance

2. Display balance

3. withdraw and update balance

4. End

2

Balance : Rs. 1200.0

Enter 1. Update balance

2. Display balance

3. withdraw and update balance

4. End .

3

Enter the withdraw amt :

500

Enter 4. Update balance

2. Display balance

3. withdraw and update balance

4. End

3

Enter the withdraw amount:

701

Amount exceeds balance!

Enter 1. Update balance

2. Display balance

3. Withdraw and update balance

4. End.

2.

Balance : RS 400.0.

Enter 1. Update balance

2. Display balance

3. Withdraw and update balance.

4. End.

4.

X ~~WIPM~~

29) Exception Handling

import java.util.Scanner;

Class WrongAge extends Exception

WrongAge (String error)

{ S. O. P (error);

}

}

Class Father

{ int age;

Father (int age) throws WrongAge {

if (age <= 0) {

throw new WrongAge ("Father's age cannot be -ve");
this. age = age;

}

}

Class Son extends Father

{ int age;

Son (int age, int sage) throws WrongAge {

super (age);

if (sage >= age)

throws WrongAge ("Son age > than Father's age");

this.sage = sage;

}

}

Class Age

{ public void main (String xx[])

{

Scanner S = new Scanner (System. in)

try {

S. O. P ("Enter Age:");

```
int F-age = s.nextInt();
s.o.p("Son's age:");
int S-age = s.nextInt();
Son a = new Son(F-age, S-age);
s.o.p("Father's age"+ F-age);
s.o.p("Son's age"+ S-age);
```

Catch (WrongAge e)

{ s.o.p("Wrong Age entered:"); }

Catch (Exception ee)

{ s.o.p("Unexpected error :" + ee); }

Output:

Enter Fage -5

Enter Sage 10

Father's age cannot be -ve

Wrong Age entered.

~~22/12/2011~~

5/2/23

- 8) Write a program which creates two threads, one thread displaying "BMS College of Engineering" once every 10 seconds and another displaying "CSE" once every two seconds.

Class Thread1 extends Thread

```
{ public void run()
{ while(true)
    System.out.println("BMS College of Engineering");
    while(true)
        try
        {
            Thread.sleep(10000);
        }
    catch(InterruptedException e)
    {
    }
}
```

Class Thread2 extends Thread

```
{ public void run()
{ while(true)
    {
        System.out.println("CSE");
        try {
            Thread.sleep(2000);
        }
    catch(InterruptedException e):
    {
    }
}}
```

Class Multithreading

```
{ public static void main(String xx[])
{
    Thread1 t1 = new Thread1();
    Thread2 t2 = new Thread2();
    t1.start();
    t2.start();
}}
```

Output:

BMS College of Engineering

CSE

CSE

CSE

CSE

CSE

BMS College of Engineering

CSE

CSE

CSE

CSE

CSE

BMS College of Engineering

CSE

~~S/Unit~~

19/2/24 LAB-Program-9

```
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;

public class EventDemo extends JFrame implements ActionListener {
    JTextField t1, t2, t3;
    JLabel l1, l2;
    JButton b1, b2;

    public EventDemo() {
        setLayout(new FlowLayout());
        l1 = new JLabel("Num1:");
        add(l1);

        t1 = new JTextField(5);
        add(t1);

        l2 = new JLabel("Num2:");
        add(l2);

        t2 = new JTextField(5); add(t2);
        t3 = new JTextField(5);
        t3.setEditable(false);
        add(t3);

        b1 = new JButton("Divide");
        add(b1);
        b1.addActionListener(this);

        b2 = new JButton("Clear");
        add(b2);
        b2.addActionListener(this);
    }

    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setSize(300, 150);
    setVisible(true);
}

public void actionPerformed(ActionEvent e) {
    String str = e.getActionCommand();
    if (str.equals("Divide")) {
        try {

```

```

int num1 = Integer.parseInt (t1.getText());
int num2 = Integer.parseInt (t2.getText());
int num3 = num1/num2;
t3.setText (" " + num3);
} catch (ArithmaticException e) {
JOptionPane.showMessageDialog (this, "Arithmatic Exception");
}
catch (NumberFormatException e) {
JOptionPane.showMessageDialog (this, "Number entered is invalid");
}
else {
t1.setText (" ");
t2.setText (" ");
t3.setText (" ");
}
}

public static void main (String xx[])
{
SwingUtilities.invokeLater (new Runnable() {
public void run () {
new ExentDemo (); } });
}
}

```

Output:

num1:	<input type="text" value="12"/>	num2:	<input type="text" value="6"/>	<input type="text" value="2"/>	- □ X
<input type="button" value="Divide"/> <input type="button" value="Clear"/>					

REPORT

1. ButtonDemo.

In this program a java file is first compiled and later the applet code is saved in another text file with .html extension. later it is compiled to obtain the output as follows. It opens a window with 3 buttons yes, no and undecided. As these buttons are clicked messages are displayed on that window.

2. Buttontag

On executing the above program a small window appears with number blocks from 1 to 9. along with reset, start and restart button. When the Number block (any one block) is clicked it gets swapped with the other block. This happens after clicking the start button. Once the reset button is clicked the whole set up starts from first.

3. Buttonlist.

On compiling the above program a pop up window appears with yes, no and undecided buttons. On clicking these buttons a message is being displayed on the same window just below these three buttons.

4. Buttonlist D.

The only difference between the 4th and 3rd program is that once any of these three buttons are clicked a new ~~new~~ window opens and a message is displayed on that window.

5. Division main:

Once this program is compiled Division of Integers window appears with three text fields named num1, num2 and the other with the result. Once two numbers are entered in each textfield the result along with the two numbers is displayed as floating point numbers.

6. Divisionmain

This program is similar to the above but the only difference is in the result the modulus value is only displayed along with the two numbers num1 and num2.

7. TextfieldDemo

On compiling the above program a window named TF-labDemo appears with two textfields Name and Password along with these name selected text in name and password is displayed in the same window.