## VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.



### LAB REPORT on

# Object Oriented Java Programming (23CS3PCOOJ)

Submitted by

Amal Roy (1BM23CS025)

in partial fulfillment for the award of the degree of BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING



## (Autonomous Institution under VTU) BENGALURU-560019 Sep-2024 to Jan-2025

#### **B.M.S.** College of Engineering,

Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)

Department of Computer Science and Engineering



#### **CERTIFICATE**

This is to certify that the Lab work entitled "Object Oriented Java Programming (23CS3PCOOJ)" carried out by **Amal Roy** (**1BM23CS025**), who is bonafide student of **B.M.S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements in respect of an Object Oriented Java Programming (23CS3PCOOJ) work prescribed for the said degree.

Lab faculty Incharge Mr. Basavaraj Jakalli Assistant Professor Department of CSE, BMSCE Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE

### Index

Sl. No.	Date	Experiment Title	Page No.
1	9/10/24	Implement Quadratic Equation	4
2	16/10/24	Student Details Program	8
3	23/10/24	Book Details Program	12
4	23/10/24	Shape Area Program	17
5	13/11/24	Week 5	22
6	13/11/24	Week 6	30
7	20/11/24	Week 7	35
8	27/11/24	Week 8	40
9	27/11/24	Week 9	44
10	27/11/24	Week 10	47

#### Github Link:

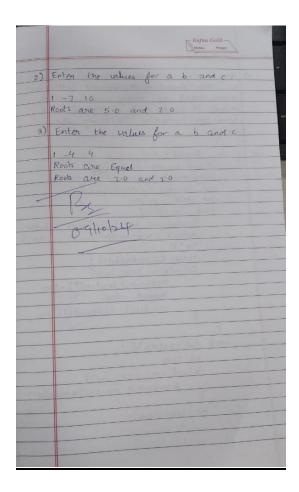
https://github.com/amalrtms/OOJLABBMS

#### Program 1

Develop a Java program that prints all real solutions to the quadratic equation ax2 +bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2 -4ac is negative, display a message stating that there are no real solutions.

	ive, dispiay a message stating that there a
9/10/24	Week   Bajna Gold
Ci	
1)	Program to g solve quadratric equations
	import java util Scanner; import java lang.*; class Weekl Guadralic
	public static void maia (String angst])
	double a, b, (;
- 1	double a a di
	Scanner sc=new Scanner (System.in);
	Scanner sc-new Scanner (System.in); System out println ("Enter the values for a b and c:\n");
100	a=sc.next Double();
	b=sc.next Double ();
	C=S(next Double ();
	if (a==0)
	3
	System.out.println("Invalid!");
	a hand a ma color and parties by
	d=Math-pow(b, 2.0)-4.0*a*c;
	if(d>o)
	{
30761	(1=((-b)+(Math.sgrt(d)))/(double)(2*a);
	12: ((-b) - (Math sgrt(d)))/(double) (2*a);
	System out println ("Roots are "+114" and

```
else if (dso)
     ri=(-b)/(2 *0 * a);
    (2: Math. sgrt (-d) / (2.0 xa);
    System out println ("Roots are imaginary.")
     and "+11+"-1"+ +2);
    e se
   System. out. println("Roots are Equal. ");
   ri= (-b)/(2.0*a);
   System out println ("Roots are "trit" and
   Output
1) Enter the values for a b and c;
  234 2 3 4
  Roots are imaginary.
  Roots are -0.75 + 11.1989578 and +0.754
  11.1989578
```



#### **Observation Book:**

#### **Code:**

```
import java.util.Scanner;
import java.lang.*;
class Week1Quadratic
{
  public static void main(String args[])
  {
    double a,b,c;
    double r1,r2,d;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the values for a b and c:\n");
    a=sc.nextDouble();
    b=sc.nextDouble();
    c=sc.nextDouble();
    if(a==0)
    {
        System.out.println("Invalid!");
    }
}
```

```
d=Math.pow(b,2.0)-4.0*a*c;
if(d>0)
{
    r1=((-b)+(Math.sqrt(d)))/(double)(2*a);
    r2=((-b)-(Math.sqrt(d)))/(double)(2*a);
    System.out.println("Roots are "+r1+" and "+r2);
}
else if(d<0)
{
    r1=(-b)/(2.0*a);
    r2=Math.sqrt(-d)/(2.0*a);
    System.out.println("Roots are imaginary.");
    System.out.println("Roots are "+r1+"+i"+r2+" and "+r1+"-i"+r2);
}
else
{
    System.out.println("Roots are Equal.");
    r1=(-b)/(2.0*a);
    System.out.println("Roots are "+r1+" and "+r1);
}
System.out.println("Roots are "+r1+" and "+r1);
}
System.out.println("AMAL ROY 1BM23CS025.");
}
}</pre>
```

#### **OUTPUT**

```
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop

C:\Users\Admin\Desktop>javac WeeklQuadratic.java

C:\Users\Admin\Desktop>java WeeklQuadratic
Enter the values for a b and c:

2 3 4

Roots are imaginary.
Roots are -0.75+i1.1989578808281798 and -0.75-i1.1989578808281798

AMAL ROY 18M23CS025.

C:\Users\Admin\Desktop>java WeeklQuadratic
Enter the values for a b and c:

1 -7 10

Roots are 5.0 and 2.0

AMAL ROY 18M23CS025.

C:\Users\Admin\Desktop>java WeeklQuadratic
Enter the values for a b and c:

1 -4 4

Roots are 5.0 and 2.0

AMAL ROY 18M23CS025.

C:\Users\Admin\Desktop>java WeeklQuadratic
Enter the values for a b and c:

1 -4 4

Roots are Equal.

Roots are 2.0 and 2.0

AMAL ROY 18M23CS025.

C:\Users\Admin\Desktop>java MeeklQuadratic
Enter the values for a b and c:

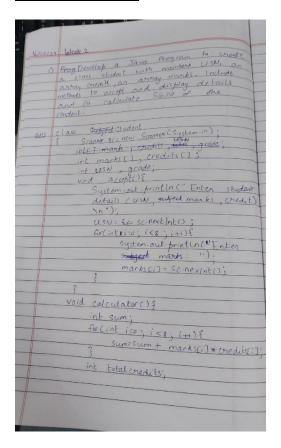
1 -4 4

Roots are 2.0 and 2.0

AMAL ROY 18M23CS025.

C:\Users\Admin\Desktop>
```

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.



	Refras Gold —
-	
-	for (int i=0; i < i+4){
1	Lotal credit = credits 63
1-	2
-	int
	grade = (int) (batel cum / total credits);
1 3 5	the allestone of source of the
	Void alisplayers
	System. out printing "usn: "+usn);
	System out print Int " swedt Point SSIPA = "+ grade);
	3
	public static void main()?
	*
	Student S=new Student();
	S. accepter;
	S. calculate();
	S-displayer;
	3
	2
	5
P	t teams and the
+	Vr

Code:

```
import java.util.Scanner;
class SGPA{
Scanner sc=new Scanner(System.in);
int marks[]=new int[8];
int credits[]=new int[8];
int USN, sum, total credits;
float grade;
void accept()
System.out.println("\nEnter Student USN: ");
USN=sc.nextInt();
for(int i=0; i<8; i++)
System.out.println("Enter the subject marks: ");
marks[i]=sc.nextInt();
System.out.println("Enter the respective credits: ");
credits[i]=sc.nextInt();
}
void calculate()
for(int i=0; i<8; i++)
sum+=marks[i]*credits[i];
for(int i=0; i<8; i++)
totalcredits+=credits[i];
grade=(float)(sum/(totalcredits*10));
void display()
System.out.println("USN: "+USN);
System.out.println("SGPA="+grade);
public static void main(String args[])
SGPA s[]=new SGPA[3];
for(int i=0; i<3; i++)
s[i]=new SGPA();
```

```
}
for(int i=0;i<3;i++)
{
s[i].accept();
s[i].calculate();
s[i].display();
System.out.println("AMAL ROY 1BM23CS025");
}
}
}
</pre>
```

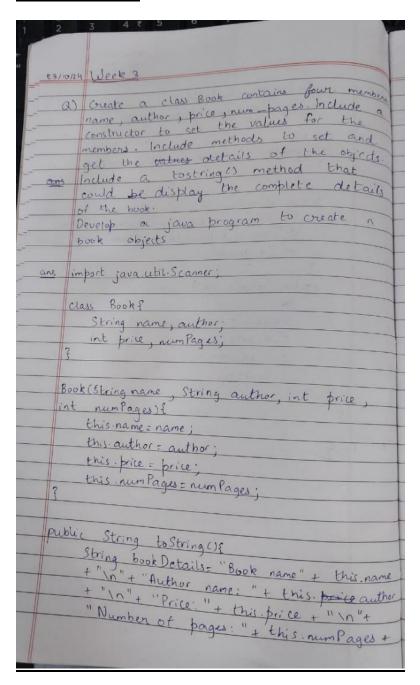
#### **OUTPUT**

```
Enter Student USN:
Enter the subject marks:
Enter the respective credits:
Enter the subject marks:
99
Enter the respective credits:
Enter the subject marks:
Enter the respective credits:
USN: 1
SGPA=9.0
```

```
Enter Student USN:
Enter the subject marks:
Enter the respective credits:
SGPA=2.0
```

```
Enter Student USN:
Enter the subject marks:
Enter the respective credits:
Enter the subject marks:
Enter the respective credits:
Enter the subject marks:
Enter the respective credits:
Enter the subject marks:
22
Enter the respective credits:
Enter the subject marks:
Enter the respective credits:
Enter the subject marks:
34
Enter the respective credits:
Enter the subject marks:
45
Enter the respective credits:
Enter the subject marks:
Enter the respective credits:
USN: 2
SGPA=3.0
```

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.



```
"(0"-13
return book Details;
public class libra
     Public Static void main (String [] orgs)
     3 Scanner S = new Scanner (System . in);
        System out print C'Enter the number of books: ");
        int n = s nextInt();
      Book[] books = new Book[n];
         for (int i=0; i(n; i++){
            System.out. print(" Enter name of
            book "+ (i+1) +": ");
            String name = sinext();
            Bystem.out. print C" Enter author of
            book " + (i+1)+"; ");
            String author=s.next();
            System. out . print ("Enter price
             of book "+ (i+1)+ ": ").
             int price = s. nextInt();
             System.out.print ("Enter number of pages in book" + (i+1)+": ");
             int numPages = s.nextInt();
             books [i] = new Book (name, author,
            price, num Pages);
```

```
& system. out println("In Book Details: "),
for (Book book: books) &
              System out print(n(book);
    Enter the number of books: 2
    Enter name of book 1: ASOIAF
    Enter author of book 1: GRRM
   Enter prize of book 1: 200
   Enter number of pages in book 1: 2999
   Enter name of book 2: Shadow
   Enter author of book 2: Bardeigh
  Enter price of book 2: 350
  Enter number of pages in book 2: 350
  Book Details
  Book name: ASOIAF
 Author name: GIRRH
 Price: 2000
Number of Pages 2999
Book name: Shadow
Author name: Bardeigh
Number of Pages: 1000
```

import java.util.Scanner;

```
class Book {
    String name;
    String author;
    int price;
    int numPages;

Book(String name, String author, int price, int numPages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.numPages = numPages;
}
```

```
public String toString() {
     String bookDetails = "Book name: " + this.name + "\n" +
                  "Author name: " + this.author + "\n" +
                  "Price: " + this.price + "\n" +
                  "Number of pages: " + this.numPages + "\n";
     return bookDetails;
  }
}
public class BookProgram{
  public static void main(String[] args) {
     Scanner s = new Scanner(System.in);
        System.out.print("AMAL ROY 1BM23CS025\n");
     System.out.print("Enter the number of books: ");
     int n = s.nextInt();
     Book[] books = new Book[n];
     for (int i = 0; i < n; i++) {
       System.out.print("Enter name of book " + (i + 1) + ": ");
       String name = s.next();
              s.nextLine();
       System.out.print("Enter author of book " + (i + 1) + ": ");
       String author = s.next();
              s.nextLine();
       System.out.print("Enter price of book " + (i + 1) + ": ");
       int price = s.nextInt();
       System.out.print("Enter number of pages in book " + (i + 1) + ": ");
       int numPages = s.nextInt();
       books[i] = new Book(name, author, price, numPages);
     }
     System.out.println("\nBook Details:");
     for (Book book: books) {
       System.out.println(book);
```

```
}
```

#### **OUTPUT**

```
AMAL ROY 1BM23CS025
Enter the number of books: 3
Enter name of book 1: ASOIAF
Enter author of book 1: GRRM
Enter price of book 1: 200
Enter number of pages in book 1: 2999
Enter name of book 2: Shadow and Bone
Enter author of book 2: Bardeigh
Enter price of book 2: 230
Enter number of pages in book 2: 1000
Enter name of book 3: The Alchemist
Enter author of book 3: Paulo Coehlo
Enter price of book 3: 260
Enter number of pages in book 3: 340
Book Details:
Book name: ASOIAF
Author name: GRRM
Price: 200
Number of pages: 2999
Book name: Shadow
Author name: Bardeigh
Price: 230
Number of pages: 1000
Book name: The
Author name: Paulo
Price: 260
Number of pages: 340
```

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.

ISBN VICE	
23/101	week 4
2	) Devlotab a T
	Develop a Java program to create an abstract class named ship interest
	an abstract class named snape that
	contains two integers and an empty method named printfrea() provide in
	method named printAreal). Provide three classes
	named Rectangle Triangle and Circle such
	that each one of the classes extends the
1 44	
	area of the given shafe.
ans	import java util. Scanner;
	import java. lang. Math. *:
	abstract class Shape { int dim 1;
	int dim 1;
	int dim 2;
	public shape Cint dimi, int dim 2) ?
	public shape () {
· V	
	this.dim1=0;
	this dim 2=0;
	3
	public. Shape Cint dim1, int dim2) {
	this.dim = dim 1;
	this dim 2: dim 2;
	Manual 3
	Box Millianne 12 mars 52 assessment
	public abstract void printAreal);
- 11	para asside void principleat),
1 3	

```
class Rectangle extends Shape &

public Rectangle Cint Length, int width);
           dim 1= length;
           dime= width;
        public void printAreaDE
           int area = dim(*dim2)
          System. out. println cuarea of Rectargle:
          + area);
  Class Circle extends Shapef
     public Circle Cint radius) {
       dim 1= radius;
       dim2=0',
    public void print Area () {
         double area = Math. P1 *dim1 *dim1;
         System.out. printin ("Area of Circle: "+
         area);
public class shapeareas
    public static void main (String[] angs) {
       Scanner SC = new Scanner (System.in);
         System.out.printlnc" Enter length and width for Rectangle: ");
```

```
int length = sc nextint is;
            int width = scineatint();
            Shape rectangle = new Rectargle Clength,
            rectangle. print Area ();
            System. out. println1 "Enton base and height for Triangle: ");
            int base = Sc next Int()
int height = Sc next Int()
             Shape triangle = new Triangle Chase,
             height);
             triangle. print Areal);
             System-out.println("Enter radius for
             Circle ");
             Sc. radius = Sc. next Int();
             Shape circle = new circle (radius);
             circle print Area();
             30 close
 Output
Enter length and width for rectangle:
Area of Rectangle: 6
Enter base and height for Triangle:
45
Area of Triangle: 10.0
                                          23/0/24
 Enter radius for Circle
Area of Circle: 201. 0619298297 4676
```

import java.util.Scanner;
abstract class Shape {
 int dim1;
 int dim2;

```
public Shape() {
     this.dim1 = 0;
     this.dim2 = 0;
  }
  public Shape(int dim1, int dim2) {
     this.dim1 = dim1;
     this.dim2 = dim2;
  public abstract void printArea();
}
class Rectangle extends Shape {
  public Rectangle(int length, int width) {
     dim1 = length;
     dim2 = width;
  public void printArea() {
     int area = dim1 * dim2;
     System.out.println("Area of Rectangle: " + area);
  }
class Triangle extends Shape {
  public Triangle(int base, int height) {
     dim1 = base;
     dim2 = height;
  public void printArea() {
     double area = 0.5 * dim1 * dim2;
     System.out.println("Area of Triangle: " + area);
class Circle extends Shape {
```

```
public Circle(int radius) {
     dim1 = radius;
     dim2 = 0;
  }
  public void printArea() {
     double area = Math.PI * dim1 * dim1;
     System.out.println("Area of Circle: " + area);
}
public class shapearea{
  public static void main(String[] args) {
     Scanner sc = new Scanner(System.in);
        System.out.println("AMAL ROY 1BM23CS025\n\n");
     System.out.println("Enter length and width for Rectangle:");
     int length = sc.nextInt();
     int width = sc.nextInt();
     Shape rectangle = new Rectangle(length, width);
     rectangle.printArea();
     System.out.println("Enter base and height for Triangle:");
     int base = sc.nextInt();
     int height = sc.nextInt();
     Shape triangle = new Triangle(base, height);
     triangle.printArea();
     System.out.println("Enter radius for Circle:");
     int radius = sc.nextInt();
     Shape circle = new Circle(radius);
     circle.printArea();
```

```
OUTPUT

AMAL ROY 1BM23CS025
Enter length and width for Rectangle:
2 3
Area of Rectangle: 6
Enter base and height for Triangle:
4 5
Area of Triangle: 10.0
Enter radius for Circle:
8
Area of Circle: 201.06192982974676
```

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.

-	Week 5
(Q)	Book that sava program to create a day
	war wan ain ains wo kinds of account
	for it's customers, one called savings
	account provides compound interest and
	with drawal facilities but no cheque book
	facility but no interest : Current account holders should also maintain a minimum
	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
	lot account. From this acrive
	Cur- acct and Sav-acct to make sun
	more checific to their requirements.
	the necessary methods in order too
	cichieve the following tasks:
	a) Accept deposit from customer and update
	a) Accept copision of
	the balance:
	6) Display the balance
	c) Compute and deposit interest
	b) Display the balance c) Compute and deposit interest d) Permit withdrawal up and update the
	balance Lland impose
	balance e) Check for the minimum balance, impose penalty is necessary and update the
	penalty is necessary and upon
	balance

```
abstract class Account &
   Account estring Gust name, String
   num, double inti-bala) {
      this cust_name = cust_name;
     this are num = Acc num;
     this balance = inti _ bala;
   abstract void deposit (double ant);
   abstract void display Balance ();
   abstract void with draw (double amt);
class Sav acct extends Account?
   double interest Rate;
   Saw Acet Cstring cust name, String ace nun
   double inti bala) ?
      super (aust name, acc num, inti-bala);
       this interest Rate - interest Rate;
   void deposit (double amt) {
     balance+-amt;
   void display Balance USE
      System-out-println ("saving Balance: "+
     halano);
   void withdraw (double ant) {
     if (amt <= balance) {
```

```
balance - = amt;
  void compute And De positeInterest() {
      balance += balance * interest Rate (100)
class own acet extends Accounts
   static final double MIN_BALANCE = 1000
   SERVICE - CHARGE = 50.
   Cur acut ( String cust name, String Ace.
   String are num, double inti-bala ) {
       super (cust name, String ace-num, double inti-bala) &
          super Coust name, String are num,
           inti-bala) {
             super cust name, orce rum
             inh bala),
     void deposit (double ant) {
        balance += ant,
     void display Balance () 3
       System out pointln [" sawing Balance: "+
      balance);
     void withdraw (double amt) {
         if Cank = balance ) {
             balance -= ant
```

```
( + Chalance MIN BALANCE) &
               balance - = SERVICE CHARGE.
    bublic static void main (String[] angs) {
       Scanner so new Scanner (System.in)
      Syxtem out pint in C" Enter diceount type
      (Savigs/aurren : "1")
             Jape = SC. next Line();
      System out printle (" Enter account
      name : ") ;
     String name = Sc nextline();
     Account account;
     FCtype
public void withdraw (double aimt) ?
    if (antx = balance) {
     balance -= amt 'i
      System out print loc "Withdrawal
     amount = " + amount );
   if balance < MIN_ BOLANCE) {
      babance = SERVI CE _ CHARGE ;
      Systemout. println("Service charge =
     SERVICE_CHARGE)!
```

```
else E
     System out printle ("Sorvice Change
     deduct losifficient before . ");
public class Bank &
   public Static void main (String []
   ang () {
       Scanner Sc= new Scanner (System.in);
        Sav_acta acct = new Sav_actc
        "AMAL", 123, 12345, 15-);
        Cun-out accz = new - Cun out C
       "A", 1754, 100000, 5002, 500);
       System out printer (" Choose ACC by de
       In 1. Savings In 2. Current In').
       int on = scinert Into);
       Switch (ch) {
         case " System out println (" savings
          account selected . ");
         ocel deposit (500);
         acci . compute And Deposit Intersters.
          acci withdraw (300);
          ace 1. display Balance (1)
          break;
        Case 2: System. Gut. println C"Current
        Account Selected ").
         acez de posit (500)
         acce with draw ( 10001;
         acc 1. display Balane ();
         bieat,
```

```
Choose Account
    1 Savings Account
   Sovings Account Selected
   Deposited: 500.0
   Interest added: 75
  Withdrawn: 300.0
  Balance: 290
  Choose Account Type
 1. Savings Account
 2. Current Account
aurorent Account Selected
Deposit: 500.0
Withdrain: 1800 id
Balane. 200
```

```
import java.util.*;
abstract class Account {
    String Cust_name ,Acc_num;
    double balance;
    Account(String Cust_name,String Acc_num, double inti_bala) {
        this.Cust_name=Cust_name;
        this.Acc_num=Acc_num;
        this.balance=inti_bala;
    }
    abstract void deposit(double amt);
    abstract void displayBalance();
    abstract void withdraw(double amt);
}
class Sav_Acct extends Account {
        double intrestRate;
```

```
Sav_Acct(String Cust_name,String Acc_num, double inti_bala){
           super(Cust_name,Acc_num,inti_bala);
           this.intrestRate=intrestRate;
  void deposit(double amt){
    balance+=amt;
  void displayBalance(){
    System.out.println("Saving Balance:"+balance);
  void withdraw(double amt){
  if(amt<=balance){</pre>
    balance-=amt;
  void computeAndDepositeIntrest(){
     balance+=balance*intrestRate/100;
  }
}
class Cur_Acct extends Account{
 static final double MIN BALANCE=1000, SEVICE CHARGE=50;
 Cur_Acct(String Cust_name,String Acc_num, double inti_bala){
        super(Cust_name,Acc_num,inti_bala);
 void deposit(double amt){
    balance+=amt;
 void displayBalance(){
    System.out.println("Saving Balance:"+balance);
 void withdraw(double amt){
  if(amt<=balance){
    balance-=amt;
    if(balance<MIN_BALANCE){
      balance-=SEVICE_CHARGE;
class Bank{
  public static void main(String[] args){
    Scanner scn=new Scanner(System.in);
    System.out.println("Enter account type(savings/current):");
```

```
String type=scn.nextLine();
    System.out.println("Enter account name:");
    String name=scn.nextLine();
    System.out.println("Enter account number:");
    String number=scn.nextLine();
    Account account;
   if(type.equals("Savings")){
      System.out.println("Initial balance and interest rate:");
      account=new Sav_Acct(name,number,scn.nextDouble());
   }
   else{
      System.out.println("Intial balance :");
      account =new Cur_Acct(name,number,scn.nextDouble());
   while(true){
      System.out.println("1.Deposit 2.Display Balance 3.Withdraw 4.Interest 5.Exit");
      int choice=scn.nextInt();
      switch(choice){
        case 1:account.deposit(scn.nextDouble());
            break;
        case 2: account.displayBalance();
             break:
        case 3: account.withdraw(scn.nextDouble());
        case 4: if(account instance of Sav Acct){
             ((Sav_Acct)account).computeAndDepositeIntrest();
             break;
        case 5 : return;
OUTPUT
AMAL ROY 1BM23CS025
Choose Account Type:
```

```
AMAL ROY 1BM23CS025
Choose Account Type:
1. Savings Account
2. Current Account
1
Savings Account Selected
Deposited: 300.0
Interest added: 65.0
Withdrawn: 100.0
Balance: 1265.0
```

Create a package CIE which has two classes - Personal 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 Personal. 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.

	Student java
	Contraction City
	pachage CIE; public class Student?
	bublic String Usn:
	public String usn; faublic String name;
	public int semestor;
	public Student (String usn, String na
	int sem) {
	this. wn = usn
- 1	this name = name
	this semester = semester
	3
	3
pub	evinals java  kage CIE;  wie class Internal Marks;  public int[] internal Marks;
<i>f</i> 1	subtice Internals (String us) ; some sem, internal Marks ) { nt sem, internal Marks of semesten); super (uso, name, sembly, semesten); this internal Marks = internal Marks of ;
- <del> </del>	subtice Internals (String us) ; some sem, internal Marks ) { nt sem, internal Marks of semesten); super (uso, name, sembly, semesten); this internal Marks = internal Marks of ;
3	sublic Internals (String List ) 2  nt sen, intel internal Marks ) 2  super (List , name , semb); semester);  this internal Marks = internal Marks 346;
3	subtice Internals (String us) ; some sem, internal Marks ) { nt sem, internal Marks of semesten); super (uso, name, sembly, semesten); this internal Marks = internal Marks of ;
3 3 Exta	internals (String us);  nt sen, intel internal Marks ) {  nt sen, intel internal Marks ) {  super (usn, name, sembli, semesten);  super (usn, name, sembli, semesten);  bus internal Marks = internal Marks 3 (6);  thus internal Marks = internal Marks 3 (6);  and : java  age SEE;
3 3 Exta	sublic Internals (String us);  nt sen, intel internal Marks ) {  super (usn, name, sembli, semesten);  super (usn, name, sembli, semesten);  thus internal Marks = internal Marks 3 (6);  thus internal Marks = internal Marks 3 (6);
3 3 Exta packa impos	aublic Internals (String us) ; comp rume  nt sen, internal Marks ) {  nt sen, internal Marks internal Marks } ;  this internal Marks = internal Marks } ;  erad : java  age SEE;  et CIE Student;
3 3 2 5 acka	aublic Internals (String us);  nt sen, internal Marks ) {  nt sen, internal Marks   Semester);  super (usn, name, sempl; semester);  this internal Marks = internal Marks 3 %;  this internal Marks = internal Marks 3 %;  erad : java  age SEE;  et CIF Student;  class External extends Student {
packa impos	applic Internals (String us);  nt sen, internal Marks ) {  nt sen, internal Marks of semesten);  super (usn, name, sembli, semesten);  thus internal Marks = internal Marks of;  thus internal Marks = internal Marks of;  thus internal Marks = internal Marks of;  thus internal extends Student f  blic internal extends Student f  blic internal see Marks;
packa impos	applic Internals (String us);  nt sen, internal Marks ) {  nt sen, internal Marks of semesten);  super (usn, name, sembli, semesten);  thus internal Marks = internal Marks of;  thus internal Marks = internal Marks of;  thus internal Marks = internal Marks of;  thus internal extends Student f  blic internal extends Student f  blic internal see Marks;
to the state of th	aublic Internals (String us);  nt sen, internal Marks ) {  nt sen, internal Marks   Semester);  super (usn, name, sempl; semester);  this internal Marks = internal Marks 3 %;  this internal Marks = internal Marks 3 %;  erad : java  age SEE;  et CIF Student;  class External extends Student {

```
Main Program. Java
import CIE Internals;
import SEE External;
import java util. Scanner;
public class Final Marks ?
  public static void main (string[] args) {
    Scanner Sc= New Scanner (System in);
  System out print("Enter number of students: "S.
   int n = sc.nextInt();
   Internals [] internal Students = new Internaling
   External [] external Students = new External [n];
    for Cin(-1:=0; i<0; i++) {
       System.out.print(n("\nEnter details for
student "+(i+1)+":");
System.out.print("Enter USN: ");
String usn=sc.next();
        System out print ("Enter internal marks
        for 5 courses: ");
        for Cint j=0; j <5; j++) {
           internal Marks [j] = sc. nextInt ()
        int[] sec Marks = new int (5);
        System.out. println C'Enter SEE marks
        for 5 courses: ");
        for (int j=0; j(5; j+1){
          See Markstjj=sc-nextInt();
```

```
internal Students [i] = new Internal's Cusn,

name, semester, internal Marks);

external Students [i] = new External Cusn,

name, semester, see Marts);

3

System out printly ("In Final Marks of

Students: ");

Ar (int i = 0; i < n; i++) f

system out printly ("Instudent" "(i+i) +"

- usn: "finternal Students [i] - usn);

for Cint j = 0; j < 5; j ++) {

int final Mark = internal Students [i] -

internal Mark S[j] + Cexternal Students [i] -

see Mark [j] /2);

System out printly ("Course" + Cj+1) +

"Final Mark: " + Binal Mark);

3

Sea Sc. close (1;

3

3
```

```
Internals.java
```

```
package CIE;

public class Internals extends Student {
   public int[] internalMarks;

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

Student.java
package CIE;

public class Student {
   public String usn;
   public String name;
   public int semester;
```

```
public Student(String usn, String name, int semester) {
    this.usn = usn;
    this.name = name;
    this.semester = semester;
  }
External.java
package SEE;
import CIE.Student;
public class External extends Student {
  public int[] seeMarks;
  public External(String usn, String name, int sem, int[] seeMarks) {
     super(usn, name, sem);
    this.seeMarks = seeMarks;
  }
MainProgram.java
import CIE.Internals;
import SEE.External;
import java.util.Scanner;
public class MainProgram{
  public static void main(String[] args) {
       System.out.println("AMALROY 1BM23CS025");
     Scanner scanner = new Scanner(System.in);
     System.out.print("Enter number of students: ");
    int n = scanner.nextInt();
     Internals[] internalStudents = new Internals[n];
     External[] externalStudents = new External[n];
     for (int i = 0; i < n; i++) {
       System.out.println("\nEnter details for student " + (i + 1) + ":");
       System.out.print("USN: ");
       String usn = scanner.next();
       System.out.print("Name: ");
       String name = scanner.next();
       System.out.print("Semester: ");
       int sem = scanner.nextInt();
```

```
int[] internalMarks = new int[5];
        System.out.println("Enter internal marks for 5 courses:");
        for (int j = 0; j < 5; j++) {
          internalMarks[j] = scanner.nextInt();
       int[] seeMarks = new int[5];
        System.out.println("Enter SEE marks for 5 courses:");
        for (int j = 0; j < 5; j++) {
          seeMarks[j] = scanner.nextInt();
       internalStudents[i] = new Internals(usn, name, sem, internalMarks);
        externalStudents[i] = new External(usn, name, sem, seeMarks);
     System.out.println("\nFinal Marks of Students:");
     for (int i = 0; i < n; i++) {
       System.out.println("\nStudent " + (i + 1) + " - USN: " + internalStudents[i].usn);
       for (int j = 0; j < 5; j++) {
          int finalMark = internalStudents[i].internalMarks[j] + (externalStudents[i].seeMarks[j]
/ 2);
          System.out.println("Course" + (j + 1) + "Final Mark: " + finalMark);
     scanner.close();
       System.out.println("AMALROY 1BM23CS025");
  }
OUTPUT
 nter details for student 1:
    SEE marks for 5 courses:
97 94 93
     internal marks for 5 courses:
2 23 49
     SEE marks for 5 courses:
45 3 46
```

Write a program that demonstrates handling of exceptions in inheritance tree. Create a case class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge() when the input age<0. In Son class, implement a constructor that uses both father and son's age and throws an exception if son's age is >=father's age.

	Week 7
0	Define a subclass of Exception.
	The exception class does not define any methods
Q'	Write a Program that demonstrates, handling
	of Exceptions in an Inheritance Strong Tree.
	Create a base class called father and a
	derived class class called oy 'son' that
	extends from base class.
	In father's class, implement a constructor
	which takes the age, and throws the
	exception: Wrong Age(), when the input age
	i's less than zero. In son's class, imprement
I to he	a constructor that uses both father and
	son's age and throws an exception it
3	Son's age is greater wan or equal to
	fother's age.
ans	class Age extends Exceptions
	public Age (String message)?
	public Age(String message)?  super(message)  3
	3 1 0
	?
	The state of the s
	al - la Harry S
	class fathers
+	int age;
	public father (int age) throws Age ?
	18(000<0)9
	throw new Age ("Age cannot be cess
	than 7000"):
	than zero");
	S

Unis age = age;
System-out-printlnC"Father's age is="+
this age);
3
3
class son extends father {
to be a second to be
public son (int fatherage, int sonage) throws
Super (fatherage);
if (sonage < 0) ?
throw new Age ("Son's age cannot be
throw new Age ("Son's age cannot be dess than or equal to father's
age ");
son sont = new sonte
7
11 Cs. age Pathanagas S
i'f (sonage >= fatherage) &
throw new Age ("Son's age cannot be
more than or equal to Father's age"
4
this . son age = sonage;
Systemout. println ("Son's age 13="+
Enis - sonage);
3
?

public dass Inheritance Tree ?
public static void main (string[] angs) {
System out println("ana
try 8
System.out.println C "CASE 1: Valid care
Son Son = new Son (60, 70);
System.out printlnc" in CASE 2: Son's
age is more than or equal to
Father's age ");
Son sonz=new son(20,30);
3 catch (Age a) ?
System out println ("Exception: "+
a.getMessage(s);
\$
try?
System out pint ("In CASE 3: Fathers
age is (ess than zero");
father father = new father (-20);
Fratch (Age a) ?
System. out.print(n ("Exception: " + a get Hessage
());
3
3
2
OUTPUT
CASEI: Valid case
Father's age is = 60
Son's age is = 20

CASE 2: Son's age is more than or equal to father's age is = 20

Exception son's age cannot be more than see or equal to father's age

CASE3: Father's age is less than zero

Exception: Age cannot be less than zero

Exception: Age cannot be less than zero

```
class Age extends Exception{
  public Age(String message){
     super(message);
}
class father{
  int age;
  public father(int age) throws Age {
     if (age < 0) {
       throw new Age("Age cannot be less than zero");
     this.age = age;
     System.out.println("Father's age is= " + this.age);
}
class son extends father{
  int sonage;
  public son(int fatherage, int sonage) throws Age {
     super(fatherage);
     if (sonage < 0) {
       throw new Age("Son's age cannot be less than zero");
     if (sonage >= fatherage) {
       throw new Age("Son's age cannot be more than or equal to Father's age");
```

```
this.sonage = sonage;
     System.out.println("Son's age is= " + this.sonage);
}
public class InheritanceTree{
  public static void main(String[] args) {
       System.out.println("AMALROY 1BM23CS025");
       System.out.println("CASE 1: Valid case");
       son son 1 = new son(60, 20);
       System.out.println("\nCASE 2: Son's age is more than or equal to Father's age");
       son son2 = new son(20, 30);
     } catch(Age a){
       System.out.println("Exception: " + a.getMessage());
     }
     try{
       System.out.println("\nCASE 3: Father's age is less than zero");
       father father 1 = \text{new father}(-20);
       catch (Age a){
       System.out.println("Exception: " + a.getMessage());
  }
```

```
AMALROY 1BM23CS025
CASE 1: Valid case
Father's age is= 60
Son's age is= 20

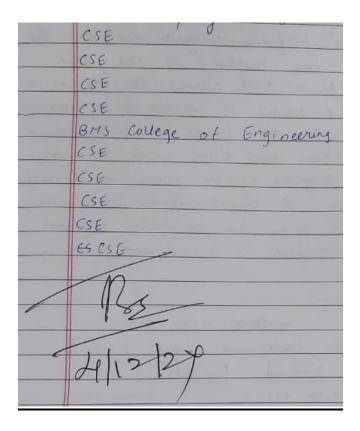
CASE 2: Son's age is more than or equal to Father's age
Father's age is= 20
Exception: Son's age cannot be more than or equal to Father's age
CASE 3: Father's age is less than zero
Exception: Age cannot be less than zero
```

## **Program 8**

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

**Observation Book** Week 8 Program to which creates two threads, one thread displaying "BHS College of Engineering" once every ten seconds and another displays "CSE" once every two seconds. class College thread extends Throad? bublic wid run (){ trys for Cint 1=01,1<5; 1+1){ System out prititing "BMS courge of Engineering "); Thread sleep (10000); 3 catch (Interrupted & Exception e) { System. out brintln ("College Thread Interrupted "); Thread program extends Thread &

The Day of S
public class Thread Program {  public state void main(String[] args) {  public state void main(String[] args) {
College thread college thread = new college inreads;
(ollege thread starte);
csethread.stant();
3
3
output
CSE
BMS College of Engineering
CSG
CSE
cse
CSE
CSE
BMS College of Engineoung
cse
CSE
C86
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE .
cse
CSØ
The state of the s
SE



```
class Collegethread extends Thread{
  public void run(){
     try{
       for(int i=0; i<5; i++){
         System.out.println("BMS College of Engineering");
         Thread.sleep(10000);
     }
     catch(InterruptedException e){
       System.out.println("College Thread interrupted.");
  }
}
class CSEthread extends Thread{
  public void run(){
    try{
       for (int i = 0; i < 25; i++){
         System.out.println("CSE");
         Thread.sleep(2000);
       }
    catch (InterruptedException e){
```

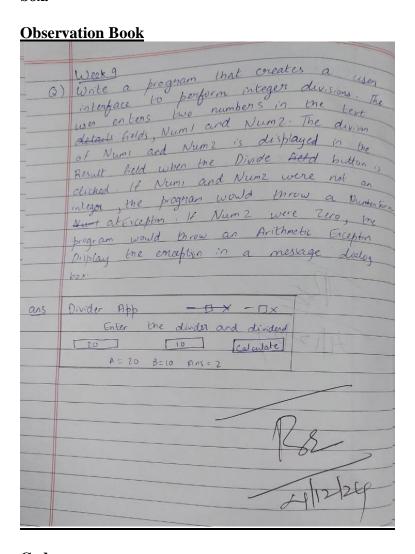
```
System.out.println("CSE Thread interrupted.");
}

public class ThreadProgram{
  public static void main(String[] args){
    System.out.println("AMAL ROY 1BM23CS025");
    Collegethread collegethread=new Collegethread();
    CSEthread csethread=new CSEthread();
    collegethread.start();
    csethread.start();
}
```

```
AMAL ROY 1BM23CS025
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
```

#### Week 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 Arithmetic Exception Display the exception in a message dialog box.



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

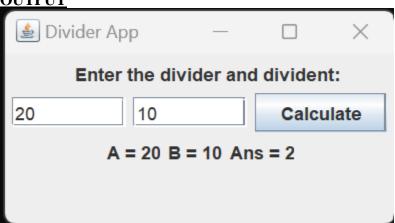
class SwingDemo {
    SwingDemo() {
        JFrame jfrm = new JFrame("Divider App");
        jfrm.setSize(275, 150);
    }
}
```

```
ifrm.setLayout(new FlowLayout());
jfrm.setDefaultCloseOperation(JFrame.EXIT ON CLOSE);
JLabel ilab = new JLabel("Enter the divider and dividend:");
JTextField aitf = new JTextField(8);
JTextField bjtf = new JTextField(8);
JButton button = new JButton("Calculate");
JLabel err = new JLabel();
JLabel alab = new JLabel();
JLabel blab = new JLabel();
JLabel anslab = new JLabel();
jfrm.add(err); // to display error messages
ifrm.add(ilab);
jfrm.add(ajtf);
ifrm.add(bitf);
jfrm.add(button);
jfrm.add(alab);
ifrm.add(blab);
ifrm.add(anslab);
ActionListener l = new ActionListener() {
  public void actionPerformed(ActionEvent evt) {
     System.out.println("Action event from a text field");
  }
};
ajtf.addActionListener(l);
bitf.addActionListener(l);
button.addActionListener(new ActionListener() {
  public void actionPerformed(ActionEvent evt) {
     try {
       int a = Integer.parseInt(ajtf.getText());
       int b = Integer.parseInt(bjtf.getText());
       int ans = a / b;
       alab.setText("\nA = " + a);
       blab.setText("\nB = " + b);
       anslab.setText("\nAns = " + ans);
     } catch (NumberFormatException e) {
       alab.setText("");
       blab.setText("");
       anslab.setText("");
       err.setText("Enter Only Integers!");
     } catch (ArithmeticException e) {
```

```
alab.setText("");
blab.setText("");
anslab.setText("");
err.setText("B should be NON zero!");
}

});
jfrm.setVisible(true);
}

public static void main(String args[]) {
    // create frame on event dispatching thread
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new SwingDemo();
        }
     });
}
```



# $\frac{Week\ 10}{Demonstrate\ Inter\ process\ Communication\ and\ deadlock}$

# **Observation Book**

	Week 10
(a)	brows communication and dendlock
	brows communical demonstrate inter
	and deadlack
ans	QUIPUT
	PCFixed java
	Put = 0
	Chot: 0
	Put=1
	Consumed = 0
	Got=1
	Consumed = 1
	Put = 2
	Got = 2
	Consumed = 2
	Put=3
	Got = 3
	Consummed = 3
0	Pendlock, inva
	<u>Neadlock.java</u> Main Thread entered A.foo
R	oring Thread entered B. bar
	dang threat entering but
1	Igin Thread trying to call B. Last()
	nside A last
В	ack in Main Thoread
R	acing Thread trying to call A laster
ln	side A last
	ack in other thread
10	and the said
	4
4	4
	TV
1	8/2/29
1	f JV

```
class A {
  synchronized void foo(B b) {
    String name = Thread.currentThread().getName();
    System.out.println(name + " entered A.foo");
```

```
try {
       Thread.sleep(1000);
     } catch (Exception e) {
       System.out.println("A Interrupted");
     System.out.println(name + " trying to call B.last()");
     b.last();
  }
  synchronized void last() {
     System.out.println("Inside A.last");
}
class B {
  synchronized void bar(A a) {
     String name = Thread.currentThread().getName();
     System.out.println(name + " entered B.bar");
     try {
       Thread.sleep(1000);
     } catch (Exception e) {
       System.out.println("B Interrupted");
     System.out.println(name + " trying to call A.last()");
     a.last();
  }
  synchronized void last() {
     System.out.println("Inside B.last");
  }
}
class Deadlock implements Runnable {
  A a = new A();
  B b = new B();
  Deadlock() {
     Thread.currentThread().setName("MainThread");
     Thread t = new Thread(this, "RacingThread");
     t.start();
     a.foo(b);
```

```
System.out.println("Back in main thread");
}

public void run() {
   b.bar(a);
   System.out.println("Back in other thread");
}

public static void main(String args[]) {
   System.out.println("AMAL ROY 1BM23CS025");
   new Deadlock();
}
```

```
AMAL ROY 1BM23CS025
MainThread entered A.foo
RacingThread entered B.bar
RacingThread trying to call A.last()
MainThread trying to call B.last()
```