VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming
(21CS3PCOOJ)

Submitted by

MANAV TAKE (1BM21CS102)

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

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019

October-2022 to Feb-2023

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 (21CS3PCOOJ)" carried out by **MANAV TAKE (1BM21CS102)**, 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 during the year2022. The Lab report has been approved as it satisfies the academic requirements in respect of a Database Management Systems (22CS3PCDBM) work prescribed for the said degree.

Prof. Vikranth B.M.
Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor & Head
Department of CSE
BMSCE, Bengaluru

Index

1. Quadtratic Equations

Question:

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.

Code:

```
import java.util.Scanner;

public class Quadratic
{

public static void main(String args[])
{
Scanner ip = new Scanner(System.in);
System.out.println("Enter the value of a: ");
double a= ip.nextDouble();
System.out.println("Enter the value of b: ");
double b= ip.nextDouble();
System.out.println("Enter the value of c: ");
double c= ip.nextDouble();
double r1,r2;
double d= b*b-(4*a*c);

if(a==0)
{
  System.out.println("Quadratic doesnt exist");
}

else {

if(d>0)
{
    r1= (-b + Math.pow(d,0.5)  )/(2*a);
    r2= (-b - Math.pow(d,0.5) )/(2*a);
    System.out.println("Roots are real and distinct");
    System.out.println("The roots are :"+ " "+r1 +" and " +r2);
}

else if(d==0)
{

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

```
else if(d<0)
{
   r1= (-b)/(2*a);
   r2= Math.pow(-d,0.5)/(2*a);
   System.out.println("Roots are unreal and distinct");
   System.out.println("The roots are :");
   System.out.println("Root1 = "+r1+ "+ "+r2+"i");
   System.out.println("Root2 = "+r1+ "- "+r2+"i");
}

else
System.out.println("Invalid input");
   }

   }
}
```

Output:

```
C:\Manav CS102>javac Quadratic.java

C:\Manav CS102>java Quadratic
Enter the value of a:
0
Enter the value of b:
6
Enter the value of c:
8
Quadratic doesnt exist

C:\Manav CS102>
```

```
C:\Manav CS102>javac Quadratic.java

C:\Manav CS102>java Quadratic
Enter the value of a:
6
Enter the value of b:
1
Enter the value of c:
3
Roots are unreal and distinct
The roots are :
Root1 = -0.08333333333333333+ 0.70217914776469661
Root2 = -0.08333333333333333- 0.70217914776469661

C:\Manav CS102>
```

2. SGPA of a student

Question:

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.

Code:

```
import java.util.Scanner;
class Sgpa {
    Scanner sc = new Scanner(System.in);
    int marks[]  = new int[3];
    int credits[]  = new int[3];
    int gradePoints[] = new int[3];

    public void enterMarks() {
        System.out.println("Enter marks and credits");
        for (int i = 0; i < marks.length; i++) {
            System.out.println("Enter subject " + (i+1) + "'s marks");
            marks[i] = sc.nextInt();

            System.out.println("Enter subject " + (i+1) + "'s credits");
            credits[i] = sc.nextInt();
        }
    }

    float calculateSGPA() {
        float sgpa;
        int sumOfCredits = 0;
        int numerator = 0;

        // Evaluate grade points
        for (int i = 0; i < credits.length; i++) {
            if (marks[i] >= 90) {
                gradePoints[i] = 10;
            } else if (marks[i] < 90 && marks[i] >= 80) {
                gradePoints[i] = 9;
            } else if (marks[i] < 80 && marks[i] >= 70) {
                gradePoints[i] = 8;
            } else if (marks[i] < 70 && marks[i] >= 60) {
                gradePoints[i] = 7;
            } else if (marks[i] < 60 && marks[i] >= 50) {
                gradePoints[i] = 6;
            } else if (marks[i] < 50 && marks[i] >= 40) {
                gradePoints[i] = 5;
            } else {
                gradePoints[i] = 0;
```

```
            }

            sumOfCredits += credits[i];
            numerator += (credits[i]*gradePoints[i]);
        }

        sgpa = (float) (numerator/sumOfCredits);
        return sgpa;
    }

    public static void main(String args[]) {
        Sgpa ob = new Sgpa();
        ob.enterMarks();
        float result = ob.calculateSGPA();
        System.out.println("The SGPA of the student is " + result);
    }
}
```

Output:

```
Enter marks and credits
Enter subject 1's marks
50
Enter subject 1's credits
6
Enter subject 2's marks
90
Enter subject 2's credits
2
Enter subject 3's marks
30
Enter subject 3's credits
1
The SGPA of the student is 6.0
```

```
Enter marks and credits
Enter subject 1's marks
85
Enter subject 1's credits
3
Enter subject 2's marks
95
Enter subject 2's credits
3
Enter subject 3's marks
75
Enter subject 3's credits
3
The SGPA of the student is 9.0
```

3. SGPA of a student

Question:

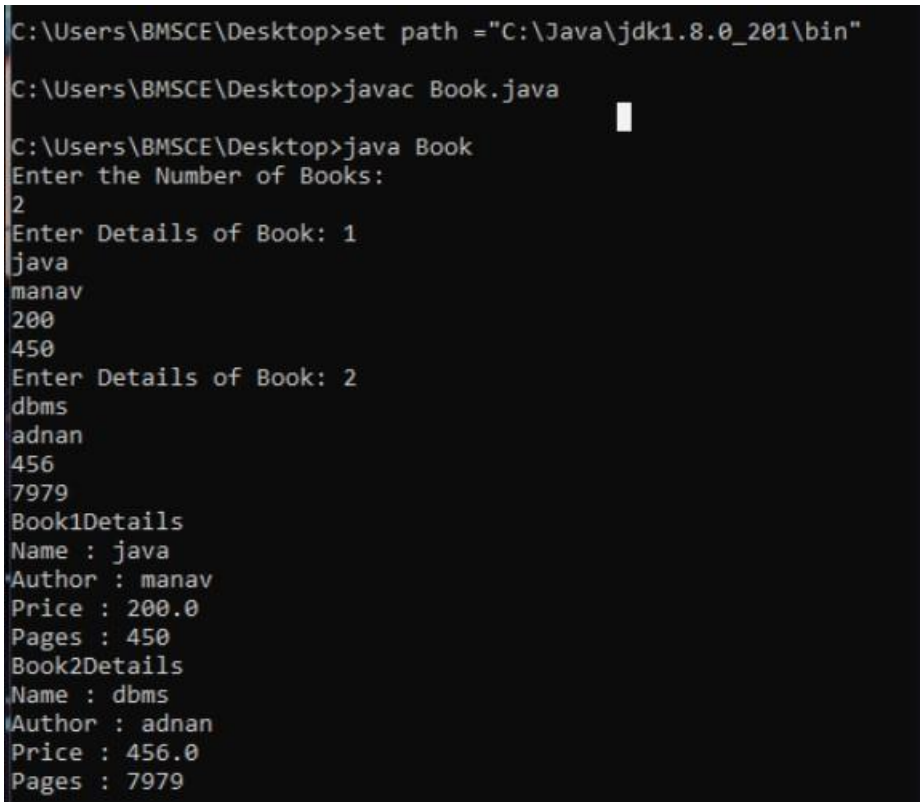
Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

Code:

```
import java.util.Scanner;

class Book
{

String title;
String author;
int numPages;
double price;


Book(){
title="Default";
author="Default";
price=0.0;
numPages=0;
}

void setTitle(String t)
{
title=t;
}

void setAuthor(String a)
{
author=a;
}

void setPrice(double p)
{
price=p;
}

void setPages(int np)
{
numPages=np;
}
```

```
public String toString()
{
return title+ "\t"+ author +"\t"+price +"\t"+ numPages ;
}

}



class BookDetails
{

public static void main(String args[])
{
String t,a;
double p;
int np,n;
Scanner sc = new Scanner(System.in);
System.out.println("Enter the num of books: ");
n=sc.nextInt();

Book b[]=new Book[n];
for(int i=0;i<n;i++)
{
System.out.println("Enter the Title of book"+(i+1)+" : ");
t=sc.next();

System.out.println("Enter the Name of book author : ");
a=sc.next();

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

System.out.println("Enter the num of pages : ");
np=sc.nextInt();

b[i]=new Book();
b[i].setTitle(t);
b[i].setAuthor(a);
b[i].setPrice(p);
b[i].setPages(np);
}

System.out.println("Title\tAuthor\tPrice\tPages\n");
for(int i=0;i<n;i++)
System.out.println(b[i]);
}

}
```

Output:


C:\Users\BMSCE\Desktop>set path ="C:\Java\jdk1.8.0_201\bin"
C:\Users\BMSCE\Desktop>javac Book.java
C:\Users\BMSCE\Desktop>java Book
Enter the Number of Books:
2
Enter Details of Book: 1
java
manav
200
450
Enter Details of Book: 2
dbms
adnan
456
7979
Book1Details
Name : java
Author : manav
Price : 200.0
Pages : 450
Book2Details
Name : dbms
Author : adnan
Price : 456.0
Pages : 7979

4. Shapes (using abstract)

Question:

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.

Code:

```
import java.util.*;
abstract class Shape
{
   int a,b;
   abstract public void printArea();
   void assign_twovar(int x,int y)
   {
      a=x;
      b=y;
   }
   void assign_onevar(int x)
   {
      a=x;
   }
}
class Rectangle extends Shape
{
   public void printArea()
   {

     System.out.println("The area of rectangle is "+(a*b)+"cm sq");
   }
}
class Triangle extends Shape
{
   public void printArea()
   {
     System.out.println("The area of triangle is "+(0.5*a*b)+"cm sq");
   }
}
class Circle extends Shape
{
   public void printArea()
   {
     System.out.println("The area of circle is "+(3.14*a*a)+"cm sq");
   }
}
public class Main
```

```
{
   public static void main(String args[])
   {
      Scanner sc=new Scanner(System.in);
      int c,dim1,dim2;
      while(true)
      {
         System.out.println("Enter 1 to find area of rectangle.\nEnter 2 to find area of
triangle.\nEnter 3 to find area of circle.\nEnter 4 to exit!");
         c=sc.nextInt();
         switch(c)
         {
            case 1:
              Rectangle rec=new Rectangle();
              System.out.println("Enter the length and breadth of rectangle in cm:");
              dim1=sc.nextInt();
              dim2=sc.nextInt();
              rec.assign_twovar(dim1,dim2);
              rec.printArea();
              break;

            case 2:
              Triangle tri=new Triangle();
              System.out.println("Enter the length and height of triangle in cm:");
              dim1=sc.nextInt();
              dim2=sc.nextInt();
              tri.assign_twovar(dim1,dim2);
              tri.printArea();
              break;

            case 3:
              Circle cir=new Circle();
              System.out.println("Enter the radius of circle in cm:");
              dim1=sc.nextInt();
              cir.assign_onevar(dim1);
              cir.printArea();
              break;

            case 4:
               System.exit(0);

            default:
            System.out.println("You have entered a wrong choice!");
            sc.close();
         }

      }
   }

}
```

Output:

```
Enter 1 to find area of rectangle.
Enter 2 to find area of triangle.
Enter 3 to find area of circle.
Enter 4 to exit!
1
Enter the length and breadth of rectangle in cm:
5
5
The area of rectangle is 25cm sq
Enter 1 to find area of rectangle.
Enter 2 to find area of triangle.
Enter 3 to find area of circle.
Enter 4 to exit!
2
Enter the length and height of triangle in cm:
10 5
The area of triangle is 25.0cm sq
Enter 1 to find area of rectangle.
Enter 2 to find area of triangle.
Enter 3 to find area of circle.
Enter 4 to exit!
3
Enter the radius of circle in cm:
5
The area of circle is 78.5cm sq
```

5. Bank Account

Question:

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.

Code:

```
import java.util.Scanner;

class Account
{
	String name;
	int type;
	long accno;
	double balance;
	void setA()
	{
		Scanner s=new Scanner(System.in);
		System.out.print("Enter customer name: ");
		name=s.nextLine();

		System.out.print("Enter account number: ");
		accno=s.nextLong();
		System.out.print("Enter bank balance: ");
		balance=s.nextDouble();
	}
	void display()
	{
		System.out.println("Customer name is: "+name);
		if(type==1) {
			System.out.println("Customer account type is: Savings");
		}
		else {
			System.out.println("Customer account type is: Current");
		}
		System.out.println("Customer account number is: "+accno);
```

```
                System.out.println("Current balance is: "+balance);
        }
        void deposit()
        {
                System.out.print("Enter the amount to be deposited: ");
                Scanner x=new Scanner(System.in);
                double amt=x.nextDouble();
                balance+=amt;
        }
}


class Sav_acct extends Account
{
        double interest;
        Scanner s=new Scanner(System.in);

        Sav_acct() {
                type=1;
        }
        void cinterest()
        {
                int timey;
                float irate;
                System.out.println("Compound Interest details:");

                System.out.println("Enter time in years: ");
                timey=s.nextInt();
                System.out.println("Enter rate of interest: ");
                irate=s.nextFloat();
                System.out.println("Interest will be compunded 5 times a year");
                interest=balance*(Math.pow((1+irate/5),(5*timey)));
                balance+=interest;
        }
        void withdraw()
        {
                System.out.println("Enter the amount to be withdrawn: ");
                double amt=s.nextDouble();
                if(balance>amt)
                {balance-=amt;}
                else
                {System.out.println("Amount to be withdrawn greater than balance!!!");}
        }

}

class Curr_acct extends Account
{
        double check_amt;
```

```
	Curr_acct() {
		type=2;
	}
	void cheque()
	{
		System.out.print("Enter the cheque amount: ");
		Scanner s=new Scanner(System.in);
		check_amt = s.nextDouble();
		if(check_amt>balance-5000)
		{
			System.out.println("Rs. 500 penalty imposed...Is it ok to proceed?
Enter y for yes and n for no");
			String option=s.next();
			if(option.equals("y")) {balance=balance-check_amt-500;}
			else {System.out.println("no check debited");}
		}
		else
		{
			System.out.println("Rupees "+check_amt+" debited"); balance-
=check_amt;
		}
	}
	void withdraw()
	{
		System.out.println("Enter the amount to be withdrawn: "); Scanner s=new
Scanner(System.in);
		double amt=s.nextDouble();
		if(balance>amt)
		{balance-=amt;}
		else
		{System.out.println("Amount to be withdrawn greater than balance!!!");}
	}
}
class Bank {
	public static void main(String ss[]) {
		String op1,op2;
		Scanner s=new Scanner(System.in);
		System.out.println("1. Savings or   2. Current?");
		int q;
		q=s.nextInt();
		if(q==1) {
			Sav_acct s1 = new Sav_acct();
			while(true) {
			System.out.print("Enter the choice: \n1 .Set the values for savings
acc\n2. display\n3. deposit\n4. Interest\n5. Withdraw\n6. exit\n");
			op1=s.next();
			switch(op1)
			{
			case "1":s1.setA();
				break;
```

```
                    case "2":s1.display();
                            break;
                    case "3":s1.deposit();
                            break;
                    case "4":s1.cinterest();
                            break;
                    case "5":s1.withdraw();
                            break;
                    case "6":System.exit(0);
                    }
                    }
            }
            else if(q==2) {
                    Curr_acct c1 = new Curr_acct();
                    while(true) {
                    System.out.print("Enter the choice: \n1.Set the values for current
account\n2. display\n3. deposit\n4. transferCheck\n5. Withdraw\n6. exit\n");
                    op2=s.next();
                    switch(op2)
                    {
                    case "1":c1.setA();
                            break;
                    case "2":c1.display();
                            break;
                    case "3":c1.deposit();
                            break;
                    case "4":c1.cheque();
                            break;
                    case "5":c1.withdraw();
                            break;
                    case "6":System.exit(0);
                    }
                    }
            }
        }
}
```

Output:

```
Enter the type of account you want to create, 1 for saving and 2 for current account
2
Enter customer name
manav
Enter account number
2580
Enter initial balance  minimum amt :1000
2000
1.Deposit
2.Display
3.Withdraw
4.Issue cheque book
5.Exit
Enter your choice
4
Cheque book issued
1.Deposit
2.Display
3.Withdraw
4.Issue cheque book
5.Exit
Enter your choice
3
Enter amount to be withdrawn
500
Balance=1500.0
```

```
1. Savings or   2. Current
1
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
1
Enter customer name: Manav
Enter account number: 123
Enter bank balance: 2000
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
2
Customer name is: Manav
Customer account type is: Savings
Customer account number is: 123
Current balance is: 2000.0
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
3
Enter the amount to be deposited: 500
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
4
```

```
Compound Interest details:
Enter time in years:
2
Enter rate of interest:
5
Interest will be compunded 5 times a year
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
2
Customer name is: Manav
Customer account type is: Savings
Customer account number is: 123
Current balance is: 377656.25
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
5
Enter the amount to be withdrawn:
30000
Enter the choice:
1 .Set the values for savings acc
2. display
3. deposit
4. Interest
5. Withdraw
6. exit
2
Customer name is: Manav
Customer account type is: Savings
Customer account number is: 123
Current balance is: 347656.25
```

6. Exception Handling

Question:
Develop a Java program to create a user defined exception called MyException.

Code:

```
class MyException extends Exception{
        private int detail;

        MyException(int a){
                detail=a;
        }

        public String toString(){
                return "MyException [" +detail + " user defined Exception" + "]";
        }

}

class ExceptionDemo{
        static void compute(int a) throws MyException
        {
                System.out.println("Called compute(" + a+") " );

                if(a>10)
                throw new MyException(a);

                System.out.println("Normal exit");
        }

        public static void main(String args[]){
                try{
                compute(1);
                compute(20);
            }


                catch(MyException e){
                System.out.println("Caught " +e);
                }


                try{
                int l = args.length;
                System.out.println("l = "+ l);
                int b=40/l;
                }
```

```
            catch(ArithmeticException e){

            System.out.println("Divide by 0: "+e);
            }

        }
}
```

Output:

```
Called compute(1)
Normal exit
Called compute(20)
Caught MyException [20 user defined Exception]
1 = 0
Divide by 0: java.lang.ArithmeticException: / by zero
```

7. Father – Son Age Exception

Question:

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.

Code:

```
import java.util.Scanner;
class WrongAge extends Exception
{
	private String detail;
	WrongAge(String a)
	{
		detail=a;
	}
	public String toString()
	{
		return "WrongAge [" +detail +"]";
	}
}
class Father
{
	int father_age;
	Father(int a)
	{
		father_age=a;
	}
}
class Son extends Father
{
	int son_age;
	Son(int a,int b)
	{
		super(a);
		son_age=b;
		try
		{
			if(son_age<=0 || father_age<=0)
			{
				throw new WrongAge("Son's age or Father's age is less than
zero");
			}
			if(father_age<=son_age)
			{
				throw new WrongAge("Father's age is less than Son's age");
			}
```

```
                else
                        System.out.println("Entered age is valid");
            }
            catch(WrongAge e)
            {
                System.out.println("Caught "+e);
            }
        }
}
class Age
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter Father's age and Son's age");
        int fa=sc.nextInt();
        int sa=sc.nextInt();
        Son ob=new Son(fa,sa);
    }
}
```

Output:

```
Enter Father's age and Son's age
50
20
Entered age is valid
```

```
Enter Father's age and Son's age
-50
20
Caught WrongAge [Son's age or Father's age is less than zero]
```

```
Enter Father's age and Son's age
50
60
Caught WrongAge [Father's age is less than Son's age]
```