INDEX

Name Subas Kumar Reddy Nare

Standard CSE Section C Roll No. 240

Subject OOJ - Lab

Sl.No.	Date	Title	Page No	Teacher's Sign
1)	—	Grocery		
2)	—	Book - maintenance program		
3)	—	Abstract class		
4)		Package		
5)		Textfield and io functioning		

19/1/24

OOJ - Lab - 2

Date :
Page No :

1)

```
class Overload {
    void print(int n) {
        int sum = 0;
        for(int i=1; i<=n; i++) {
            sum = sum + i;
        }
        System.out.println("sum of " +n+ " natural numbers is " +sum);
    }

    void print(int m, int n)
        System.out.println("prime number in the range are:");
        for(int i=m; i<=n; i++) {
            int p=0;
            for(int j=1; j<=(i/2); j++) {
            if(i%j==0) {
                p=1;
                break;
            }
            }
            if(p==0) {
                System.out.println(i);
            }
        }
    }
}
public class Output {
    public static void main(String[] args) {
        Overload a = new Overload();
        a.print(12)
        a.print(32, 38)
    }
}
```

Output:
The sum of 10 number is 75
The prime numbers are:

```
② class Grocery {
    String c_name;
    String c_phone;
    double total;
    Grocery (String c_name; String c_ph) {
      this.c_name = c_name;
      this.c_phone = c_phone;
    }
    void calc (double del, double pulses, double sugar) {
      bill = total = del*100 + pulses*60 + sugar*25;
    }
    void display () {
      System.out.println ("name is" + " phone " + " bill");
      System.out.println (c_name + " " + c_phone + " " + bill);
      System.out.println();
    }
}
public class Bill {
    public static void main (String[] args) {
      Grocery g = new Grocery ("suhas", "8023");
      g.calc(5,2,4);
      g.display();
    }
}
```

Lab - 3

import java.util.Scanner

class

Q1) Create a class Book that contains four members: name, author, price and num_pages. Include a constructor to set and get the details of the object. Include a to_string() method that could display the complete details of the book. Develop a java program to creat n book objects.

Program:-

```
import java.util.Scanner
class Books {
    String name;
    String author;
    int price;
    int num_pages;
    Books() { }
    Books(String name, String author, int price, int num_pages);
    {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }
    public string toString()
    {
        String name, author, price, num_pages
        {
            name = " Book name " + this.name + " \n ";
```

```
        author = "Author name " + this.author + "\n";
        price = "Price " + this.price + "\n";
        num_pages = "number of pages " + this.num_pages + "\n";
        return name + author + price + num_pages;
    }
}
class Main {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        int n;
        String name;
        String author;
        int price;
        int num_pages;
        System.out.println("Book " + (i+1) + ":");
        System.out.println("Enter the number of books");
        n = s.nextInt();
        Books b[];
        b = new Books[n];
        for (int i = 0; i < n; i++) {
            System.out.println("Book " + (i+1) + ":");
            System.out.println("Enter the name of the book");
            name = s.nextLine();
            System.out.println("Enter the name of author of the book");
            author = s.nextLine();
            System.out.println("Enter the price of the book");
            price = s.nextInt();
            System.out.println("Enter the number of pages");
            num_pages = s.nextInt();
            b[i] = new Books(name, author, price, num_pages);
        }
```

```
        for (int i=0 ; i<n ; i++) {
            System.out.println("Book" + (i+1) + " :\n" + b[i]);
        }
    }
}
```

Output :-

Enter the number of books :
2

Book 1 :
enter the name of the book :
harry potter
enter the author of the book :
Jk Rowling
enter the price of the book :
600
enter the number of pages in the book :
638

Book 2 :
enter the name of the book :
Percy Jackson
enter the author of the book :
Rick Riordan
enter the price of the book :
720
enthe the number of pages in the book
560

Book 1 :
Book name : Hairy Potter
Author name : Jk Rowling

Date :
Page No :

Price 600
number of page : 638

Book 2:
Book name : Percy Jackson
Author name : Rick Riordan
Price : 720
Number of pages : 560

12/1/24

Lab - 4

1) 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 the methods printArea() that prints the area of the given shape

pgm:

```
abstract class Shape {
    int d1, d2;
    public Shape(int d1, int d2){
        this.d1 = d1;
        this.d2 = d2;
    }
    public abstract int printArea();
}
class Rectangle extends Shape {
    public Rectangle(int length, int width){
        super(length, width);
    }
    public int printArea() {
        int area = d1*d2;
        System.out.println("Rectangle's area is = "+area);
    }
}
class Triangle extends Shape {
    public Triangle(int base, int height){
        super(base, height);
    }
    public int printArea() {
        int area = 0.5*d1*d2;
        System.out.println("Triangle's area is = "+area);
    }
}
```

```
class Circle extends Shape {
    public Circle (int radius, int unnecessary) {
        super (radius, unnecessary);
    }
    public int printArea () {
        double area = Math.pi * d1 * d1;
        System.out.println(" Circle's area is = " + area);
    }
}
public class Main {
    public static void main (String[] args) {
        Rectangle rec = new Rectangle (10, 12);
        Triangle tri = new Triangle (8, 12);
        Circle cir = new Circle (12, 5000);
        rec.printArea();
        tri.printArea();
        cir.printArea();
    }
}
```

output :-

Rectangle area is = 120
Triangle area is = 48
Circle area is = 452.389342

Date : switch
Page No :

2) Develop a Java program to create a class Bank that maintains two kinds of accounts 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 class Cur-acc and Sav-acc 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.

Pgm:- class Accounts {

int acc_no;

String

Pgm:-

```
class Account {
    int balance;
    int acc_no;
    String Customer;
    String type;
public Account (int balance, int accno, String Customer, String type) {
        this.customer = customer;
        this.balance = balance;
        this.accno = accno;
        this.type = type;
    }
public void display () {
        System.out.println("Balance is : " + balance);
    }
public void deposit (int dep) {
        balance = balance + dep;
        System.out.println(" Current balance is :" + balance);
    }
public void withdraw (int draw) {
        if (balance > draw) {
            balance = balance - draw;
        }
        else {
            System.out.println("Insufficient balance");
        }
    }
}
class Saving extends Account {
    public Saving (
```

```
class CurAct extends Accounts
{
    private double minBalance = 1000;
    private double serviceCharge = 500;
    public CurAct (String CustomerName, long accountNo,
                   double Balance)
    {
        super (CustName, AccNo., Balance);
    }

class SavAct extends Account
{
    public double interest rate = 0.05;
    public SavAct (String CustName, Long AccNo, double balance)
    {
        Super (CustName, AccNo., balance);
    public void computeInterest ()
    {
        double interest = balance * interestrate;
        balance += interest
        System.out.println ("Interest "+interest+" credited");
        displayBalance();
    }}

class Main
{
    public static void main (String[] args)
    {
        Scanner s = new Scanner (System.in);
        SavAct sa = new SavAct ("Albert, 1234, 5000);
        SA. DisplayBalance;
        S. Deposit (1000);
```

```
SA.computeInterest();
SA.Withdraw(2000)
{
}
```

Lab - 4

Q

pgm:-

```
package CIE;
class Student {
    protected String usn;
    protected String name;
    protected int sem;
    Student (String usn, String name, int sem){
        This.sem = sem;
        this.name = name;
        this.usn = usn;
    }
}
public class Internals extends Student {
    int [] ciemarks;
    public Internals ( String usn , String Nam
                       int sem, int [] cie marks
    {
        super(usn, name, sem);
        this.ciemarks = ciemarks;
    }
}
```

```
package SEE
public class Externals extends Student {
    int [] seeMarks;
public Externals (String usn, String name, int sem,
 {
    super (usn, name, sem)
    this.seeMarks = seeMarks
 }
}

public
import CIE.Internals;
import SEE.Externals;

public class Main {
    public static void main(String [] args);
    Internals internal = new Internals ('cs294', Suhas, 3,
                                        [35, 34, 33, 22, 2];
    Externals external = new Externals ('cs294', Suhas,
                                        [78, 94, 83, 5;
    System.out.println("The Student " + internal.name +
                       with usn " + internal.usn +
                       the semester" + internal.sem +
                       obtained the mark" + internal.cie
                       "in his internals and " +
                       external.seeMarks + " in his
                       externals");
}
```

Q 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 construct which takes the age and throws the exception WrongAge() when the input age < 0. In Son class, implement a constructor that takes both father and son's age and throws an exception if son's age >= father's age.

Program:-

```
import java.util.Scanner;

class Father {
    private int age;

    public Father(int age) throws WrongAgeException {
        if (age < 0) {
            throw WrongAgeException();
        }
    }
    this.age = age;
}
}

Class Son extends Father {
    private in sonage;

    public Son(int SonAge, int fatherAge) throws WrongAgeException {
        super(fatherAge);
        if (SonAge >= fatherAge) {
            throw WrongAgeException
        }
        this.SonAge = sonAge;
    }
}
```

```
public class Main {
    public static void main(String[] args) {
        try {
            Scanner s = new Scanner(System.in);
            System.out.println("Enter Father's age:");
            int fatherage = s.nextInt();
            System.out.println("Enter Son's age:");
            int sonage = s.nextInt();

            Son son = new Son(fatherage, sonage);
            System.out.println("Both ages are valid");
        }
    }
}
```

Q) Write a program which creates two threads, one displaying "BMS College of Engineering" one every ten seconds the other displaying "CSE" once every 2 seconds

Program:-

```
class DisplayThread extends Thread {
    private String message;
    private int interval

    public DisplayThread(String message, int interval) {
        this.message = message;
        This.interval = interval;
    }

    @Override
    public void run() {
        while (true) {
            System.out.println(message);
```

```
        try {
            Thread.sleep (interval * 1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

public class Main {
    public static void main (String[] args) {
        DisplayThread bms = new DisplayThread ("BMS College of Engg", 10);
        DisplayThread cse = new DisplayThread ("CSE", 2);

        bms.start();
        cse.start();
    }
}
```

Lab - 5

1) Creating label, button and Textfield in a frame using AWT

```
import java.awt.*; import java.awt.event.*;
public class AWTExample extends WindowAdapter {
    Frame f;
    AWTExample() {
        f = new Frame();
        f.addWindowListener(this);

        Label l = new Label("Employee_id:");
        Button b = new Button("submit");
        TextField t = new TextField();

        l.setBounds(20, 80, 80, 30);
        t.setBounds(20, 100, 80, 30);
        b.setBounds(100, 100, 80, 30);

        f.add(l);
        f.add(t)
        f.add(b)
        f.setSize(400, 300);
        f.setTitle("Employee info:");
        f.setLayout(null);
        f.setVisible(true);
    }

    public void windowClosing(WindowEvent e) {
        System.exit(0);
    }

    public static void main(String[] args) {
        AWTExample awt_obj = new AWTExample();
    }
}
```

Date :
Page No :

2) Create a button and add an action listener for mouse click.

Prgm:

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

public class EventHandling extends WindowAdapter
    implements ActionListener {
    Frame f;
    TextField t;
    EventHandling() {
        f = new Frame();
        f.addEventListener(this);
        t = new TextField();
        t.setBounds(60,50,170,20);
        Button b = new Button("Click me");
        b.setBounds(100,120,80,30);

        b.addActionListener(this);

        f.add(b);
        f.add(t);
        f.setSize(300,300)
        f.setLayout(null)
        f.setVisible(true)
    }
    public static void actionPerformed(ActionEvent e
        t.setText("Welcome");
    }
    public void windowClosing(WindowEvent
        System.exit(0);
    }
```

```
public static void main(String[] args){
    new EventHandling();
  }
}
```

Programs on IO

3)
```
import java.io.*;
public class ByteArrayInput {
    public static void main(String[] args) throws IOException {
        byte[] buf = {35, 36, 37, 38};

        ByteArrayInputStream byt = new ByteArrayInputStream
                                   (buf);

        int k = 0;
        while ((k = byt.read()) != -1) {
            char ch = (char) k;
            System.out.println("ASCII value of
              Character is: " + k + "; Special Character
              is:" + ch + ");
        }
    }
}
```

4)
```
import java.io.*;

public class ByteArray_ex1
    public static void main (String[] args)
        throws Exception {
        FileOutputStream fout = new FileOutputStream
                                ("Example.txt");

        FileOutputStream fout = new FileOutputStream("Example2.txt");
        ByteArrayOutputStream bout = new ByteArrayOutputStream();
        bout.write(65);
        bout.writeTo(fout1);
        bout2.writeTo(fout2);
        bout.flush();
        bout.close();
        System.out.println("Success...");
    }
}
```

5)
```
import java.io.*;
public class FileEx1{
    public static void main (String a[]) throws IOException {
        FileInputStream fin = new FileInputStream ("Example.txt");
        byte[] bytes = new byte[20]
        int i;
        char c;
        i= fin.read(bytes);
        System.out.println("Number of bytes read :"+i);
        System.out.println("Bytes Read:");
        for (byte b:bytes){
            c= (char)b;
            System.out.print(c);
        }
    }
}
```