B.M.S. College of Engineering

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)
Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of
Computer Science & Engineering (CSE)

Report Programs Observation

Course Title: Object Oriented Java Programming

Course Code: 23CS3PCOOJ

BY
Bhavya Hanwitha(1BM22CS095)

B.M.S. College of Engineering

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)
Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of
Computer Science & Engineering (CSE)

CERTIFICATE

This is to certify that the report on **"Java Lab Programs"** has been carried out by **Bhavya Hanwitha** bearing USN **1BM22CS095** as a part of AAT for the course **Object Oriented Java Programming** with course code **23CS3PCOOJ**, Computer Science and Engineering from Visvesvaraya Technological University, Belgaum during the year 2023–24. It is certified that all corrections/suggestions indicated for Internal Assessments have been incorporated in the report.

Bhavya Hanwitha
1BM22CS095

Shravya AR
Department of CSE
BMSCE, Bengaluru-19

```
1  import java.util.scanner;
   ~~int~~
   class Books
   {
     string name;
     String author;
     int price;
     int numpages;

   Books(){ }
   Books(String name, String autor, int price, int
     numpages)
   {
     this.name = name;
     this.author = author;
     this.price = price;
     this.numpages = numpages;
   }
   public String toString()
   {
     String name, author;
     String price, numpages;
     name = "Book NAME:"+ this.name+ "\n";
     author = "Author name:"+ this.author +"\n";
     price = "PRICE:" + this.price +"\n";
     numpages = "Number OF PAGES:"+this.numpages;
     return name+author + price +numpages;
   }
  }
  class main
  {
  public static void main(String args[])
   {
```

Date ___/___/___

Page ______

```
Scanner s = new scanner (system.in);
int n;
string name;
String author;
int price;
int numpages;
system.out.println ("enter no.:");
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 name:");
 name = s.next();
 Syslem.out.printlu("enter Autor name:");
 author = s.next();
 system.out.println("enter price:");
 price = s.nextInt ();
 b[i] = new Books(name, author, price, nump);
}
for(i=0;i<n;i++)
System.out.println("Book "+(i+1)+ ":\n"+
                                        b[i]);
}
}
```

O/P

```
enter the number of books:
1
enter name of book:
HP
enter author name:
JKR
enter price:
540
enter page no:
400
```

2.

```
import   java.util.scanner;
class   Student {
        String   usn;'
        String   name;
         int   marks [] = new  int [6];

      void    Details()
      {
   Scanner    s = new   scanner (system.in)'
   System. out. printen ("Enter  USD")-
   usn = s.next();
   System.out.println ("enter  name");
   name = s.nent();
   system.out.print ln ("enter marks for
    6 students: ");
   for (int i=0; i<6; i++)
    {
    system.out.print ("subject " + (i+1)+":
    marks[i] = s.nentInt();
  }
   }
    double percentage ()
    {
 int  total = 0;
for (int  i=0; i<6; i++)
{
```

```
total += marks [i];
}
 double p = total / 6;
return p;
  }
void display()
{
 system.out.println("\n student det");
     system.out.println("USN:" + usn);
     system.out.println("Name: "+name);
     system.out.println("Marks:");

 for (int i=0; i<6; i++)
   {
 system.out.println("percentage:"+
       percentage() + "%");
 }
}
class Labstudent
{
  public static void main(String args[]){
  scanner s = new scanner(system.in);
  System.out.print("enter number");
  int n= s.nextInt()
  for (int i=0; i<n; i++)
 {
  students[i] = new student();
  students[i].Details();
 }
 for(Student student : students)
{
  Student.display();
}
}
```

O/P

```
enter no. of students : 2
enter details for student 1:
enter usn
095
Name:
hanwitha
enter marks for 6 subjects:
 subject 1:90
         2:90
         3:80
         4:80
         5:90
         6:60
percentage : 85%
```

8/1/2024

3. Quadratic equation

```
import java.util.scanner;
import java.lang.math;
class quadractic
{  public static void main(String args[])
     {
         int a,b,c;
         System.out.println ("enter values
                   of a,b,c resp");
          scanner sc = new scanner (System.in);
     a = sc.next int()
      b = sc.next int();
      c = sc.next int();
     Equation eq = New equation (a,b,c);
      eq.quad();
          }


class equation
{
      int a,b,c;
      Equation (int x, int y, int z)
     {
          a = x;
          b = y;
          c = z;
        }
    quad()
        {
            if(a=0)
              SOP ("not quadratic");
           else if (a>0)
    {
      SOP ln ("Two real and different
                solution")'
   double r1 = -b+ ~~math~~ )
          sqrt ((d)/(2*a));
     double r2 = -b - Math.)
     SOPln (r1);
     SOP ln (r2);
    }
else if (d== 0)
 {
    SOPln ("real & same roots");
    r1 =(-b + Math.sqrt(d))/(2*a);
    r2 = (-b - Math.sqrt(d))|(2*a);
    system.out.println("r1)
      SOPln (r2);
 }
else if (d = = 0)
    {
      sopln ("Real & sam
else ~~if (d~~
    {
      SOPln ("Imaginary sol");
     double  real = -b/(2*a);

 }
```

Output:

```
    enter the values of a,b,c
2
3
2
imaginary solutions
```

Date ___/___/___

1. Develop a java program to create abitact clan Shape two integers and an empty method printArea() provide three claues rectangle, triangle, circle,

```
abstract clau Shape {
        protected int dimension1;
        protected int dimension2;

    public shape(int d1, int d2) {
            this.d1 = d1;
            this.d2 = d2;
    }
    public abstract void printArea();
}
clau Rectangle extends shape {
        public Rectangle (int length, int width)
    {
        super(length, int width);
    }
    public void print Area() {
            int area = d1 * d2;
            System.out.printen ("Arear = " + area);
    }
}
clau Triangle extends shape {
        public Triangle ( int bace, int height)
        { super(bace, heint );

        }
    public void print Area() {
            double area = 0.5 * d1 * d2;
        SOPln("Area" + area);
    }
```

```
class circle extends shape{
        public ~~Triangle~~ circle(int radius){
          super(radius, 0);
     }
    public void printArea(){
          double area = Math.PI * d1 * d1;
          SOPln ("Area" + area);
     }
  }
public class Main{
      public static void main(String[] args){
   Rectangle rectangle = new Rectangle(4,5);
   rectangle.printArea();

   Triangle triangle = new triangle(3,6);
   triangle.printArea();

     circle circle = new Circle(7);
     circle.printArea();
  }
}
```

O/P

```
Area of Rectangle = 20
Area of triangle = 9.0
Area of circle = 153.9386
```

Date__/__/__
Page____

```
clau Bank {
    public static main (String[] args) {
        savingsAccount. savingsAccount = new
savingsAccount ("John Doe", "SA1001");
CurrentAccount currentAccount = new
currentAccount ("Janey Smith", "CA2002");

savingAccount.deposit (500);
savingsAccount.display Balance();
savings Account.computer Interest ();
savingsAccount.display Balance();
savings Account.withdraw (200);
savings Account. display Balance();

currentAccount.deposit (8000);
current Account display Balance ();
}
}
class Account {

protected String customerName;
protected String account Name;
protected double balance;
public Account (String customerName
String accountNumber) {
            this.balance = 0;
    }
public void deposit(double amount) {
    balance += amount;
    SOPl("Deposit of "+amount +"
        succesful");
  }

public void displayBalance(){
    System.out.println("Acc NO"+account
  number + "In Balance : $"+ balance);
  }
}

class SavingsAccount extends Account{
    public SavingsAccount (String customer
  Name ) {
        super (CustomerName, account Name);
  }

  public void withdraw( double amount)
  {
    if(balance >= amount){
        balance -= amount;
    System.out.println("Withdrawal of
  "+amount + "succesful");
    }
    else{
    SOPln(" Insufficient funds");
    }
  }
}
```

```
class CurrentAccount extends Account {
        private double minBalance = 1000;
    public Current Acc( String customer
    Name, string accountNumber){
        super(customerName, accountNumber
  }

public void withdraw(double amt ){
        if(balance-amount >=minimum
                                balance)

  {
   balance -= amount;
  }
   System.out.println(" Withdrawal"
   +amount+" successful");
   }
   else{
   SOPln("Insufficient funds"))
   imposeServiceCharge();
   }
  }
private void imposeservice charge()
    {
     double servicecharge=20;
     balance -= service charge
    SOPln("service charge"+
    service charge "+imposed");
```

O/P

```
Deposit of 5000 succesful
 Acc no: SA1001
  Balance : 5000.0
 Interest of 250.0 computed & added
  Acc no: SA1001
 Balance : 5250.0
 withdrawal of $2000.0 succesful
  Acc no: SA1001
 Balance : 3250.0
  Deposit of 8000.0 succesful
 Acc no: CA2002
  Balance: 8000.0
  withdrawal of 5000.0 succesful
   Acc no: CA2002
 Balance: 3000.0
```

8/10
23/1/29

6.

Packages

create a package CIE with 2-class - student internals with USN, name,

Student

```
package cie;
 public class student {
          public string name;
           public string USN;
           public int sem;
       }
```

Internals

```
 package cie;
 public class Internals extends student {
     public int[] marks = new int [5];
   }
```

external.java →

```
package see;
import cie.student;
public class External extends student [
public int[] seemarks = new int [5];
}
```

Date __/__/__
Page - ______

```
Main.java →
import  cie.internals;
import  see.External;
import  java.util.scanner;

public class main {
    pvsm (String[] args) {
        sopln ("enter no. of students");
        scanner input = new scanner (sys.in)
        int  n =  input.nextInt();

        Internals[] s1  = new   Internals[n];
        Externals [] s2  = new   Externals[n];
        int []    finalcie = new   int [n];
        int[]     final see = new   int[n];

for ( int  i=0;   i<n; i++)
    {
        s1[i] =   new   External( );
        SOP ln ("Enter name");
        s2[i].name = input.next()
        SOP ln ( "USN");
        s1 i 2.USN = input. next();
        SOP ln ("enter  sem");
          s1[i]. sem =  input. nextInt();

SOP ln ("enter  the  marks of  5 sub");
   for ( int  j=0; j<5; j++)
        { s2[i]  .seemarks [j] = input.nextInt
        finalsea[i] += s2[i].seemarks[j]
   }
```

```
SOP ln(" Final marks");
for ( int i=0; i<5; i++)
{
        SOP ln ("name:" + S1[i].name +
              " usn " + S1[i].ausn +
              "Sem:" + s1[i].sem);
   SOP ln (" Internal marks :" + final.ie[i
   SOP ln (" External marks:" + Finalsee[i]
}
}
```

Output

```
enter no. of students
1
enter name
Jack
 USN
1BM05
enter the sem
 3
marks of 5 sub
10
20
10
20
10
Final :
name: jack  usn: 01-   sem: 3
 internal :100
external :70
  total : 70
```

Program: 7

Write a program that demonstrates handling of exceptions in inheritance tree. create a bases class called "Father" & derived class called "son" which extends the base class. In Farther class, implement a constructor which takes the age & throws the exception wrongAge() when the input age<0.

In son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >= father's age

```
class wrongAge extends Exception{
        public wrongAge (String message){
                super (message);
            }
        }
class Father{
        int age;
        public Father(int age)  thraws wrongAge{
            if(age<0){
                throw new WrongAge("Age cant be
                                        negetive");
            }
        this.age = age;
        }
        public int getAge(){
                return age;
            }
        }
```

```
clas   Son  extends   Father {
            int  sonAge;
       public son (int fatherAge ,int SonAge)
  throws  wrongAge {
              super (fatherAge );
         if (sonAge >= fatherAge) {
            throw new   wrongAge
     ("Sons  age  cannot be  greater  than or
          equal " );
      }
        this.sonAge = sonAge;
    }

     public   int  getSonAge ( ) {
             return  sonAge;
        }
    }


public    clas  han {
          public    static   void  main (String[] arg
   {
        try {
         Father    father =  new   Father (35) ;
    System.out.println ("Father's  age " + father.getAg

        Son   son  = new   Son (35, 10 ) ;
        System.out.println ("Son's age : " +
    }                                        son.getSonAg
   catch (wrongAge  e) {
           System.out.println( "Exc caught" +
                                          e.getMsg()
      }
   }
}
```

O/P

```
Age cannot be negetive
```

Program: 8.

Threads

```
class   DispMessage   extends   Threads {
            String   message;
            int   interval;

      public   DispMessage (String   message , int interval
               this . message =   message;
               this . interval =   interval;
      }
     public   void   run() {
            while ( true ){
              System.out.println( message );
        try {
              Thread . sleep ( interval );
        }
        catch ( InterruptedException   e ){
                     e.printStackTrace ();
               }
           }
     }
}
public   class   Hi {
      public   static   void   main (String[] args
    {
          Disp message   bmsThread  =
              new     DispMessage ("BMS   college
    of   Engineering",  10000);

            bmsThread . start();
            cseThread . start();
    }
}
```

O/P

```
BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
BMS
```

19/2/24

program

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

public class AWTExample extends windowAdapter{
    frame f;
    AWTExample(){
        f = new frame();
        f.addwindow (this);
        Label l = new Label("Emp id:");
        Button b = new Botton ("Submit");
        TextField t = new TextField();

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

    f.add (b);
    f.add (l);
    f.add (t);

     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();
    }
  }
```

2. Create a button & add a action listener for Mouce click.

```
import java.awt.*;
import java.awt.event.*;
public class EventHandling extends
WindowAdapter implements ActionListener {
          Frame f;
          TextField tF;
          EventHandling () {

     f = new Frame ();
     f.addWindowListener (this);
     tF. setBound (60,50,170,20);
     Button b = new Button ("clink me");
     b.setBounds (100, 120, 80, 30);

     b.addActionListener (this);
     f.add(b); f.add(tF);
     f.setSize (300,300);
     f.setLayout (null);
     f.setVisible (true);
   }
```

```
    public void  action Performed (ActionEvente
   {
     tf set Text ("welcome");
  }
 }
   public  void  windowClosing (windows
   {
       systm.exit(0);
   }

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

3. Program on IO

4.

```
import  java.io.*;
public clau  ByteArrayImport{
  public  static  void  main(String[] args)
  throws  IOException{
        byte[]  buf  = {35,36, 37,38};
       ByteArrayInputStream by = new  Byt[buf]
       int  k = 0;
       while (k= byt.read()  != -1){
          char  ch = (char) k;
     SOP ("K"+&"+ch");
    }
   }
 }
```

Example 2

```
    import java.io;
    public class ByteArrayInput {
            public static v m(S a) throws IO Exception
      byte[] buf = [35,36,37,38];
    ByteArrayInputStream byt =
              new ByteArrayInputStream(buf);
      int k = 0;
    while(k = byt.read() != -1){
         char ch = (char) k;
      SOP ("Ascii "+k+ "special" +ch);
    }
   }
}
```

Example 3

```
    public class FileEx {
        p s v m (String a[] throws IOExcep {
  FileInputStream fin = new fin("Example.txt");
  int content;
  System.out.println(" remaining bytes "+fin.
                                     available());

   content = fin.read();
       SOP('fin.available());
   {
  }
```

Example 4

```
        import java.io.FileInputStream;
        import java.io.IOException;

     public class FileEx2{
           public static void main (String a[])
     throws IOException {
           FileInputStream fin= new FIS("Example");
        byte[] bytes = new byte[20];
        int i;
        char c;
         i= fin.read(bytes);
         SOP ("number"+i);
         SOP( "Bytes");
         for(byte b: bytes){
              c=(char)b;
         SOP(c);
        }
     }
```