**Course Outcome 3(CO3):**

1. **Area of different shapes using overloaded functions**

import java.util.Scanner;

public class OverloadShapeArea

{

void calculateArea(float x)

{

System.out.println("Area of the square :"+x\*x+"sq\_units");

}

void calculateArea(float x,float y)

{

System.out.println("Area of the rectangle :"+x\*y+"sq\_units");

}

void calculateArea(double r)

{

double area = 3.14\*r\*r;

System.out.println("Area of the circle :"+area+"sq\_units");

}

public static void main(String args[])

{

OverloadShapeArea obj = new OverloadShapeArea();

System.out.println("Enter the sides of a square :");

Scanner sc = new Scanner(System.in);

float side = sc.nextFloat();

obj.calculateArea(side);

System.out.println("Enter the sides of the rectangle :");

float side1 = sc.nextFloat();

float side2 = sc.nextFloat();

obj.calculateArea(side1,side2);

System.out.println("enter the radius of the circle : ");

double rad = sc.nextDouble();

obj.calculateArea(rad);

}

}

1. **Create a class ‘Employee’ with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class ‘Teacher’ that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.**

import java.util.Scanner;

class EmployeeT

{

int empid;

String name;

float salary;

String address;

EmployeeT(){}

EmployeeT(int empid , String name , float salary , String address)

{

this.empid = empid;

this.name = name;

this.salary = salary;

this.address = address;

}

}

class Teacher extends EmployeeT

{

String department,subject;

Teacher(int empid , String name , float salary , String address , String department , String subject)

{

super(empid , name , salary , address);

this.department = department;

this.subject = subject;

}

public void display()

{

System.out.println("Teacher\_id : "+empid);

System.out.println("Teacher\_name : "+name);

System.out.println("Teacher\_salary : "+salary);

System.out.println("Teacher\_address : "+address);

System.out.println("Teacher\_department : "+department);

System.out.println("Teacher\_subject : "+subject);

}

}

public class TeacherArrObjects

{

public static void main(String[] args)

{

System.out.println("Enter no.of Teachers : ");

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

System.out.println("Enter Teacher details : ");

Teacher teacher[] = new Teacher[n];

Scanner scT = new Scanner(System.in);

int tid;

String name;

float salary;

String address;

String department;

String subject;

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

{

System.out.println("Enter " + (i+1) + " teacher details : ");

System.out.println("Enter Teacher\_id(integer): ");

tid = scT.nextInt();

System.out.println("Enter Teacher\_name(String): ");

name = scT.next();

System.out.println("Enter Teacher\_salary(float): ");

salary = scT.nextFloat();

System.out.println("Enter Teacher\_address(String): ");

address = scT.next();

System.out.println("Enter Teacher\_department(String): ");

department = scT.next();

System.out.println("Enter Teacher\_subject(String): ");

subject = scT.next();

Teacher t = new Teacher(tid , name , salary , address , department , subject);

teacher[i] = t;

}

System.out.println("Teachers are : \n ");

for(Teacher x : teacher)

{

x.display();

System.out.println("\n");

}

}

}

**3. Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class ‘Employee’ that inherits the properties of class Person and also contains its own data members like Empid, Company\_name, Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.**

import java . util . Scanner ;

class Person

{

String Name, Gender , Address ;

protected int Age;

public Person ()

{}

public Person ( String n, String g , String addr , int a)

{

this .Name = n;

this . Gender = g ;

this . Address =addr ;

this .Age = a ;

}

public void displayPerson()

{

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

System.out.println("Gender : " +Gender );

System.out.println("Address : " +Address );

System.out.println("Age: " +Age );

}

}

class Employee extends Person

{

int Empid, Salary ;

String Companyname , Qualification ;

public Employee()

{ }

public Employee( String n, String g , String addr , int a ,

int eid , String cname , String qual , int sal )

{

super(n, g , addr , a );

Empid = eid ;

Companyname = cname ;

Qualification = qual ;

Salary = sal ;

}

public void displayEmployee ()

{

super . displayPerson ();

System . out . println ("Empid: "+Empid);

System . out . println ("Company name: " + Companyname );

System . out . println (" Qualification :" + Qualification );

System . out . println ("Salary :" + Salary );

}

}

class Teacher1 extends Employee

{

String Subject , Department ;

int Teacherid ;

public Teacher1( String n, String g , String addr , int a , int eid , String cname , String qual , int sal ,

String sub , String dept , int tid )

{

super(n, g , addr , a , eid , cname , qual , sal );

Subject = sub ;

Department = dept ;

Teacherid = tid ;

}

public void displayTeacher ()

{

super . displayEmployee ();

System . out . println ("Teacher id : "+Teacherid );

System . out . println ("Subject : " + Subject );

System . out . println ("Department : " + Department );

}

}

public class InheritencePersonExample

{

public static void main( String args [])

{

System . out . println ("Enter number of teachers" );

Scanner sc = new Scanner (System . in );

int N = sc . nextInt ();

Teacher1 [ ] teacher1s = new Teacher1 [N] ;

Scanner scs = new Scanner (System . in );

for ( int i = 0; i<N; i++){

System . out . println ("Enter name of the teacher . " );

String name = scs . nextLine ();

System . out . println ("Enter gender of the teacher . " );

String gen = scs . next ();

System . out . println ("Enter address of the teacher . " );

String addr = scs . nextLine ();

System . out . println ("Enter age of the teacher . " );

int ag = sc . nextInt ();

System . out . println ("EnterEmpid of the teacher . " );

int eid = sc . nextInt ();

System . out . println ("EnterCompany name. " );

String cn = scs . nextLine ();

System . out . println ("Enterqualification of the teacher . " );

String quali = scs . nextLine ();

System . out . println ("Entersalary the teacher . " );

int sal = sc . nextInt ();

System . out . println ("EnterTeacher id" );

int tid = sc . nextInt ();

System . out . println ("EnterSubject the teacher . " );

String sub = scs . nextLine ();

System . out . println ("EnterDepartment of the teacher . ");

String dept = scs . nextLine ();

Teacher1 t = new Teacher1(name, gen , addr , ag , eid , cn , quali , sub , dept , tid );

teacher1s [ i ] = t ;

}

for (Teacher1 t : teacher1s )

{

t . displayTeacher ();

}

}

}

**4. Write a program has class Publisher, Book, Literature and Fiction. Read the information and print the details of books from either the category, using inheritance.**

**/\***

**Write a program has class Publisher, Book, Literature and Fiction. Read the information**

**and print the details of books from either the category, using inheritance**

**\*/**

import java . util . Scanner ;

class Publisher

{

String publisher;

Publisher(String pubi)

{

this.publisher=pubi;

}

}

class book

{

String name;

Publisher publisher;

book()

{

}

public book(String name,Publisher publisher)

{

this.name=name;

this.publisher=publisher;

}

}

class liter extends book

{

String lit\_type="literature";

liter(String name,Publisher publisher)

{

super(name,publisher);

}

void display()

{

System.out.println("name;"+super.name);

System.out.println("type;"+this.lit\_type);

System.out.println("publisher;"+this.publisher.publisher);

}

}

class fiction extends book

{

String lit\_type="Fiction";

fiction(String name,Publisher publisher)

{

super(name,publisher);

}

void display()

{

System.out.println("name;"+super.name);

System.out.println("type;"+this.lit\_type);

System.out.println("publisher;"+this.publisher.publisher);

}

}

public class InheritanceBookExample

{

public static void main(String[] args)

{

Publisher lp=new Publisher("s chand");

liter l=new liter("as you like",lp);

l.display();

Publisher fp=new Publisher("tata");

fiction f=new fiction ("tempest",fp);

f.display();

}

}

**5. Create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student.**

/\*

Create classes Student and Sports. Create another class Result inherited from Student and

Sports. Display the academic and sports score of a student

\*/

interface student

{

int score=10;

void displayscore();

}

interface sports

{

int score=25;

void displaysportscore();

}

class result implements student,sports

{

public void displayscore()

{

System.out.println("acadamic score"+student.score);

}

public void displaysportscore()

{

System.out.println("sport score"+sports.score);

}

}

public class SportStudentResult

{

public static void main(String[]args)

{

result r=new result();

r.displayscore();

r.displaysportscore();

}

}

**6. Create an interface having prototypes of functions area() and perimeter(). Create two classes Circle and Rectangle which implements the above interface. Create a menu driven program to find area and perimeter of objects.**

**7. Prepare bill with the given format using calculate method from interface. Order No. Date : Product Id Name Quantity unit price Total 101 A 2 25 50 102 B 1 100 100 Net. Amount 150**

**Course Outcome 4 (CO4):**

1. **Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle, Square and Circle. Test the package by finding the area of these figures.**

package p1;

public class circle

{

private int radius;

public circle(int r)

{

radius=r;

}

public double area()

{

return (3.14\*radius\*radius);

}

}

package p1;

public class rectangle

{

private int l,d;

public rectangle(int a,int b)

{

l=a;

d=b;

}

public int area()

{

return (l\*d);

}

}

package p1;

public class square

{

private int side;

public square(int s)

{

side=s;

}

public int area()

{

return (side\*side);

}

}

package p1;

public class triangle

{

private int side1,side2,side3;

public triangle(int s1,int s2,int s3)

{

side1=s1;

side2=s2;

side3=s3;

}

public double area()

{

double s=(side1+side2+side3)/2;

double a=Math.sqrt((s-side1)+(s-side2)+(s-side3));

return a;

}

}

import p1.\*;

import java.util.\*;

class graphics

{

public static void main(String []args)

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter The side of the Square : ");

int s=sc.nextInt();

square sq=new square(s);

System.out.println("Area of Square is " + sq.area());

System.out.println("Enter The radius of the Circle : ");

int r=sc.nextInt();

circle ci=new circle(s);

System.out.println("Area of Circle is " + ci.area());

System.out.println("Enter The length of rectangle ");

int a=sc.nextInt();

System.out.println("Enter The breath of rectangle ");

int b=sc.nextInt();

rectangle q=new rectangle(a,b);

System.out.println("Area of Triangle is " + q.area());

System.out.println("Enter The Side1 of the Triangle : ");

int s1=sc.nextInt();

System.out.println("Enter The Side2 of the Triangle : ");

int s2=sc.nextInt();

System.out.println("Enter The Side3 of the Triangle : ");

int s3=sc.nextInt();

triangle t=new triangle(s1,s2,s3);

System.out.println("Area of Triangle is " + t.area());

}

}

1. **Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic operations. Test the package by implementing all operations on two given numbers**

package p2;

public class arithmetic

{

private int a,b;

public arithmetic(int c,int d)

{

a=c;

b=d;

}

public double add()

{

return (a+b);

}

public double div()

{

return (a/b);

}

public double sub()

{

return (a-b);

}

public double mul()

{

return (a\*b);

}

}

import p2.\*;

import java.util.\*;

class operation

{

public static void main(String []args)

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter The 1st number ");

int c=sc.nextInt();

System.out.println("Enter The 2nd number ");

int d=sc.nextInt();

arithmetic m=new arithmetic(c,d);

System.out.println("Addition of 2 numbers" + m.add());

System.out.println("Division of 2 numbers" + m.div());

System.out.println("sub of 2 numbers" + m.sub());

System.out.println("Multiplecation of 2 numbers" + m.mul());

}

}

1. **Write a user defined exception class to authenticate the user name and password.**

import java.util.Scanner;

class UsernameException extends Exception {

public UsernameException(String msg) {

super(msg);

}

}

class PasswordException extends Exception {

public PasswordException(String msg) {

super(msg);

}

}

public class CheckLoginCredential {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

String username, password;

System.out.print("Enter username :: ");

username = s.nextLine();

System.out.print("Enter password :: ");

password = s.nextLine();

int length = username.length();

try {

if(length < 6)

throw new UsernameException("Username must be greater than 6 characters ???");

else if(!password.equals("hello"))

throw new PasswordException("Incorrect password\nType correct password ???");

else

System.out.println("Login Successful !!!");

}

catch (UsernameException u) {

u.printStackTrace();

}

catch (PasswordException p) {

p.printStackTrace();

}

}

}

1. **Find the average of N positive integers, raising a user defined exception for each negative input.**

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.Scanner;

class MyException extends Exception

{

public MyException(String str)

{

System.out.println(str);

}

}

public class SignException {

public static void main(String[] args)throws IOException {

int n,i,total=0;

float p = 0;

System.out.print("Enter the range :: ");

Scanner s = new Scanner(System.in);

n=s.nextInt();

int num[]=new int[100];

System.out.print("Input number :: ");

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

{

num[i]=s.nextInt();

total=total+num[i];

p=total/n;

}

System.out.println("average is:: " +p);

try {

if(num[i] < 0)

throw new MyException("Number is negative");

else

throw new MyException("Number is positive");

}

catch (MyException m) {

System.out.println(m);

}

}

}

1. **Define 2 classes; one for generating multiplication table of 5 and other for displaying first N prime numbers. Implement using threads. (Thread class)**

**6. Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface)**

**7. Producer/Consumer using ITC**

**8. Program to create a generic stack and do the Push and Pop operations.**

**9. Using generic method perform Bubble sort.**

**10. Maintain a list of Strings using ArrayList from collection framework, perform built-in operations. 11. Program to remove all the elements from a linked list**

**12. Program to remove an object from the Stack when the position is passed as parameter**

**13. Program to demonstrate the creation of queue object using the PriorityQueue class**

**14. Program to demonstrate the addition and deletion of elements in deque**

**15. Program to demonstrate the creation of Set object using the LinkedHashset class**

**16. Write a Java program to compare two hash set**

**17. Program to demonstrate the working of Map interface by adding, changing and removing elements.**

**18. Program to Convert HashMap to TreeMap**

**Course Outcome 5 (CO5):**

1. **Program to draw Circle, Rectangle, Line in Applet.**

import java.applet.Applet;

import java.awt.\*;

public class GraphicsDemo extends Applet{

public void paint(Graphics g){

g.setColor(Color.red);

g.drawLine(20,30,20,300);

g.drawRect(70,100,30,30);

g.drawOval(70,200,30,30);

}

**2. Program to find maximum of three numbers using AWT.**

private void button1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

int a,b,c;

a=Integer.parseInt(textField1.getText());

b=Integer.parseInt(textField5.getText());

c=Integer.parseInt(textField6.getText());

if ( (a > b) && (a > c))

{

label5.setText(" "+a);

}

else if ((b > a) && (b > c))

{

label5.setText(" "+b);

}

else

{

label5.setText(" "+c);

}

}

**3. Find the percentage of marks obtained by a student in 5 subjects. Display a happy face if he secures above 50% or a sad face if otherwise.**

private void button1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

float a,b,c,d,e,f;

a=Float.parseFloat(textField1.getText());

b=Float.parseFloat(textField2.getText());

c=Float.parseFloat(textField3.getText());

d=Float.parseFloat(textField4.getText());

e=Float.parseFloat(textField5.getText());

f=(((a+b+c+d+e)/500) \* 100 );

label6.setText(" " + f);

ImageIcon image1;

if (f > 50 )

{

image1 = new ImageIcon("C:\\Users\\student.MCALAB\\Documents\\NetBeansProjects\\awt\\src\\awt\\happy.jpg");

}

else

{

image1 = new ImageIcon("C:\\Users\\student.MCALAB\\Documents\\NetBeansProjects\\awt\\src\\awt\\sad.jpg");

}

jLabel1.setIcon(image1);

}

**4. Using 2D graphics commands in an Applet, construct a house. On mouse click event, change the color of the door from blue to red.**

import java.applet.\*;

import java.awt.\*;

import java.awt.event.\*;

public class housecon extends Applet implements MouseListener {

Color color = Color.blue;

public void init()

{

addMouseListener(this);

}

public void paint(Graphics g)

{

int [] xCoords = { 40, 250, 460 };

int [] yCoords = { 200, 50, 200 };

super.paint(g);

g.drawRect(80, 200, 330, 260);

g.drawPolygon(xCoords , yCoords , 3);

g.setColor(this.color);

g.fillRect(190, 330, 100, 130);

}

public void mouseClicked(MouseEvent e)

{

this.color = color.red;

this.repaint();

}

public void mouseEntered(MouseEvent e) {}

public void mouseExited(MouseEvent e) {}

public void mousePressed(MouseEvent e) {}

public void mouseReleased(MouseEvent e) {}

}

**5. Implement a simple calculator using AWT components.**

private void button1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

float a,b,f;

a=Float.parseFloat(textField1.getText());

b=Float.parseFloat(textField2.getText());

f=(a+b);

label1.setText(" " + f);

}

private void button2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

float a,b,f;

a=Float.parseFloat(textField1.getText());

b=Float.parseFloat(textField2.getText());

f=(a-b);

label1.setText(" " + f);

}

private void button3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

float a,b,f;

a=Float.parseFloat(textField1.getText());

b=Float.parseFloat(textField2.getText());

f=(a\*b);

label1.setText(" " + f);

}

private void button4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

float a,b,f;

a=Float.parseFloat(textField1.getText());

b=Float.parseFloat(textField2.getText());

f=(a / b);

label1.setText(" " + f);

}

**6. Develop a program that has a Choice component which contains the names of shapes such as rectangle, triangle, square and circle. Draw the corresponding shapes for given parameters as per user’s choice.**

import java.awt.Graphics;

import java.applet.Applet;

public class circle extends javax.swing.JFrame {

public int a;

public circle() {

initComponents();

choice1.addItem("rectangle");

choice1.addItem("circle");

choice1.addItem("triangle");

choice1.addItem("square");

}

public void paint(Graphics g){

//System.out.println(a);

if ( a==0 )

{

g.drawRect(120,80,40,70);

}

else if ( a==1 )

{

g.drawOval(40,200,30,30);

}

else if ( a==2 )

{

g.drawLine(180,150,180,370);

g.drawLine(180,150,440,370);

g.drawLine(180,370,440,370);

}

else if ( a==3 )

{

g.drawRect(100,200,50,50);

}

}

private void choice1ItemStateChanged(java.awt.event.ItemEvent evt) {

a =choice1.getSelectedIndex();

// System.out.println(a);

/\*if (evt.getSource() == choice1 )

a = 1;\*/

repaint ();

}

**7. Develop a program to handle all mouse events and window events**

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class MouseDemo extends Applet implements MouseListener,MouseMotionListener

{

int mx=0;

int my=0;

String msg="";

public void init()

{

addMouseListener(this);

addMouseMotionListener(this);

}

public void mouseClicked(MouseEvent me)

{

mx=20;

my=40;

msg="Mouse Clicked";

repaint();

}

public void mousePressed(MouseEvent me)

{

mx=30;

my=60;

msg="Mouse Pressed";

repaint();

}

public void mouseReleased(MouseEvent me)

{

mx=30;

my=60;

msg="Mouse Released";

repaint();

}

public void mouseEntered(MouseEvent me)

{

mx=40;

my=80;

msg="Mouse Entered";

repaint();

}

public void mouseExited(MouseEvent me)

{

mx=40;

my=80;

msg="Mouse Exited";

repaint();

}

public void mouseDragged(MouseEvent me)

{

mx=me.getX();

my=me.getY();

showStatus("Currently mouse dragged"+mx+" "+my);

repaint(); }

public void mouseMoved(MouseEvent me)

{

mx=me.getX();

my=me.getY();

showStatus("Currently mouse is at"+mx+" "+my);

repaint();

}

public void paint(Graphics g)

{

g.drawString("Handling Mouse Events",30,20);

g.drawString(msg,60,40);

}

}

**8. Develop a program to handle Key events.**

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class keydemo extends Applet implements KeyListener {

String msg="";

char ms;

public void init()

{

addKeyListener(this);

requestFocus();

}

public void paint(Graphics g)

{

g.drawString(msg,100,50);

}

public void keyTyped(KeyEvent ke)

{

msg="Key Typed";

repaint();

}

public void keyReleased(KeyEvent ke)

{

msg="Key Released";

repaint();

}

public void keyPressed(KeyEvent ke)

{

int k=ke.getKeyCode();

switch(k)

{

case KeyEvent.VK\_F1:msg="F1";

break;

case KeyEvent.VK\_F2:msg="F2";

break;

case KeyEvent.VK\_F3:msg="F3";

break;

case KeyEvent.VK\_PAGE\_UP:msg="Pageup";

break;

case KeyEvent.VK\_PAGE\_DOWN:msg="Pagedown";

break;

case KeyEvent.VK\_LEFT:msg="Left arrow";

break;

case KeyEvent.VK\_RIGHT:msg="Right arrow";

break;

}

repaint();

}

}