Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-01

Practical Name:- Write a Java program that demonstrate program structure of java with use of Arithmetical and Logical implementation.

import java.util.\*;

class Logical

{

public static void main(String arge[])

{

int phy,chem,math,avg;

Scanner sc=new Scanner(System.in);

System.out.println("Enter the marks of phy");

phy=sc.nextInt();

System.out.println("Enter the marks of chem");

chem=sc.nextInt();

System.out.println("Enter the marks of math");

math=sc.nextInt();

avg=(phy+chem+math)/3;

if(avg>=60 && math>=60)

System.out.println("Eligible for engeneering");

else

System.out.println("Not Eligible for engeneering");

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-2.1

Practical Name:- Write a Java program that demonstrate string operations using String class.

class StringDemo

{

public static void main(String args[])

{

String s=new String("Java");

System.out.println(s);

System.out.println(s.length());

System.out.println(s.toUpperCase());

System.out.println(s.toLowerCase());

System.out.println(s.indexOf("v"));

System.out.println(s.charAt(3));

System.out.println(s.concat(" Programing "));

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-2.2

Practical Name:- Write a Java program that demonstrate string operations using StringBuffer class.

class StringBufferDemo

{

public static void main(String args[])

{

StringBuffer s1=new StringBuffer("good");

System.out.println(s1);

System.out.println(s1.append(" Morning"));

System.out.println(s1.reverse());

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-2.3

Practical Name:- Write a Java program to check whether string is palindrome or not.

import java.util.\*;

class StringPalindrome

{

public static void main(String args[])

{

String str,reverse="";

Scanner sc=new Scanner(System.in);

System.out.println("Enter a String");

str=sc.nextLine();

int length=str.length();

for(int i=length-1;i>=0;i--)

{

reverse=reverse+str.charAt(i);

}

if(str.equals(reverse))

System.out.println("String is palindrome");

else

System.out.println("String is not palindrome");

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-3.2

Practical Name:- Write a Java program that demonstrate static fields.

import java.util.\*;

class Players

{

int run;

String name;

static int score;

Scanner sc=new Scanner(System.in);

void get()

{

System.out.println("Enter run and name");

run=sc.nextInt();

name=sc.next();

}

void put()

{

score=score+run;

}

static void show()

{

System.out.println("score is="+score);

}

}

class Static

{

public static void main(String args[])

{

Players p1=new Players();

Players p2=new Players();

Players p3=new Players();

p1.get();

p2.get();

p3.get();

p1.put();

p2.put();

p3.put();

Players.show();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-3.3

Practical Name:- Write a Java program that demonstrate method overloading.

class Addition

{

void sum(int x,int y)

{

System.out.println("Addition is "+(x+y));

}

void sum(int x,float y)

{

System.out.println("Addition is "+(x+y));

}

void sum(float x,int y)

{

System.out.println("Addition is "+(x+y));

}

void sum(float x,float y)

{

System.out.println("Addition is "+(x+y));

}

}

class MethodOverload

{

public static void main(String args[])

{

Addition a1=new Addition();

a1.sum(10,20);

a1.sum(10,6.1f);

a1.sum(5.6f,20);

a1.sum(2.1f,6.9f);

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-4.1

Practical Name:- Write a Java program that demonstrate single level inheritance.

import java.util.\*;

class Emp

{

int sal;

String ename;

Scanner sc=new Scanner(System.in);

void get()

{

System.out.println("Enter employee name and salary");

ename=sc.next();

sal=sc.nextInt();

}

}

class Dept extends Emp

{

String dname;

void put()

{

System.out.println("Enter the dept name");

dname=sc.next();

System.out.println("----------------------------------");

System.out.println("emp name is" +ename);

System.out.println("Salary is" +sal);

System.out.println("dept name is" +dname);

}

}

class SingleDemo

{

public static void main(String args[])

{

Dept d=new Dept();

d.get();

d.put();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-4.2

Practical Name:- Write a Java program that demonstrate multi-level inheritance.

import java.util.\*;

class Stud

{

int rno;

String sname;

Scanner sc=new Scanner(System.in);

}

class Course extends Stud

{

int cfees;

String cname;

void get()

{

System.out.println("Enter the name and rollno ");

sname=sc.next();

rno=sc.nextInt();

System.out.println("Enter the course and total fees");

cname=sc.next();

cfees=sc.nextInt();

}

}

class Fees extends Course

{

int fp,fr;

void calc()

{

System.out.println("Enter the paid fees ");

fp=sc.nextInt();

fr=cfees-fp;

System.out.println("remaining fees"+fr);

}

}

class MultiLevel

{

public static void main(String args[])

{

Fees f=new Fees();

f.get();

f.calc();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-4.4

Practical Name:- Write a Java program that demonstrate method overriding.

class A

{

void put()

{

System.out.println("put method of A class");

}

}

class B extends A

{

void put()

{

super.put();

System.out.println("put method of B class");

}

}

class MethodOverride

{

public static void main(String args[])

{

B b1=new B();

b1.put();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-4.5

Practical Name:- Write a Java program that demonstrate abstract method and abstract class.

abstract class A

{

abstract void put();

}

class B extends A

{

void put()

{

System.out.println("Hello World");

}

}

class AbstractDemo

{

public static void main(String args[])

{

B b1=new B();

b1.put();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-4.6

Practical Name:- Write a Java program that demonstrate multiple inheritance using interface.

import java.util.\*;

class Student

{

int rno;

String name;

Scanner sc = new Scanner(System.in);

void get()

{

System.out.println("Enter rollno&name");;

rno=sc.nextInt();

name=sc.next();

}

}

class Test extends Student

{

int t1,t2;

void input()

{

System.out.println("Enter marks of two subject");

t1=sc.nextInt();

t2=sc.nextInt();

}

}

interface Sports

{

int score=25;

void show();

}

class Result extends Test implements Sports

{

int total;

public void show()

{

total=t1+t2+score;

System.out.println("Total marks is="+total);

}

}

class MultipleDemo

{

public static void main(String args[])

{

Result R1=new Result();

R1.get();

R1.input();

R1.show();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-5.1

Practical Name:- Write a Java program that demonstrate shapes on frames.

import java.awt.\*;

class MyFrame1 extends Frame

{

MyFrame1()

{

setVisible(true);

setSize(500,500);

setBackground(Color.cyan);

}

public void paint(Graphics g)

{

g.drawLine(50,50,50,150);

g.drawRect(100,50,250,150);

g.drawOval(150,250,50,150);

}

public static void main(String args[])

{

MyFrame1 f1=new MyFrame1();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No:77

Practical No:-6

Practical Name:- Write a Java program that demonstrate color and fonts.

import java.awt.\*;

import javax.swing.\*;

class FontDemo extends JFrame

{

FontDemo()

{

setVisible(true);

setSize(500,500);

add(new MyPanel());

}

public static void main(String args[])

{

FontDemo f1=new FontDemo();

}

}

class MyPanel extends Panel

{

Font f1,f2,f3;

MyPanel()

{

setBackground(Color.red);

f1=new Font("Times new Roman",Font.BOLD,50);

f2=new Font("Algerian",Font.ITALIC,20);

f3=new Font("Times new Roman",Font.ITALIC,30);

}

public void paint(Graphics g)

{

g.setFont(f1);

g.drawString("Hello",100,100);

g.setFont(f2);

g.drawString("Good Morning",200,200);

g.setFont(f3);

g.drawString("------------",150,150);

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No-77

Practical No:-8.2

Practical Name:- Write a Java program to illustrate use of various swing components.(JChechBox and JPanel)

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

class MyFrame extends JFrame implements ActionListener

{

JPanel p1;

JCheckBox c1,c2;

JLabel j1;

MyFrame()

{

setVisible(true);

setSize(600,600);

j1=new JLabel("Java Programing", SwingConstants.CENTER);

c1=new JCheckBox("Bold");

c2=new JCheckBox("Italic");

p1=new JPanel();

p1.add(c1);

p1.add(c2);

add (p1,"South");

add (j1);

c1.addActionListener(this);

c2.addActionListener(this);

}

public void actionPerformed(ActionEvent e)

{

int model=0;

if(c1.isSelected())

model+=Font.BOLD;

if(c2.isSelected())

model+=Font.ITALIC;

j1.setFont(new Font("Times New Roman",model,30));

}

}

class CheckBoxDemo

{

public static void main(String args[])

{

MyFrame f1=new MyFrame();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No-77

Practical No:-8.3

Practical Name:- Write a Java program to illustrate use of various swing components.(JRadioButton)

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

class MyFrame extends JFrame implements ActionListener

{

JLabel j1;

JPanel p1;

JRadioButton r1,r2,r3;

ButtonGroup g;

MyFrame()

{

setVisible(true);

setSize(500,500);

j1=new JLabel ("Hello", SwingConstants.CENTER);

add(j1);

p1=new JPanel();

r1=new JRadioButton("small");

r2=new JRadioButton("medium");

r3=new JRadioButton("large");

g=new ButtonGroup();

g.add(r1);

g.add(r2);

g.add(r3);

p1.add(r1);

p1.add(r2);

p1.add(r3);

add (p1,"South");

r1.addActionListener(this);

r2.addActionListener(this);

r3.addActionListener(this);

}

public void actionPerformed(ActionEvent e)

{

int size=0;

if(r1.isSelected())

size=20;

if(r2.isSelected())

size=40;

if(r3.isSelected())

size=60;

j1.setFont(new Font("Times New Roman",Font.BOLD,size));

}

}

class RadioButtonDemo

{

public static void main(String args[])

{

MyFrame f1=new MyFrame();

}

}

Name:- Amruta Nagesh Kulkarni.

Roll No-77

Practical No:-11

Practical Name:- Write a Java program to illustrate exception handling.

class ExceptionDemo

{

public static void main(String args[])

{

int x=10, y=0;

try

{

System.out.println(x/y);

}

catch(Exception e1)

{

System.out.println("can not divided by zero");

}

}

}