**IMPLEMENTING PACKAGE, INHERITANCE AND INTERFACE**

import java.io.\*;

import java.util.\*;

public interface ex1

{

static float pi=3.14F;

float area(float rad);

float perim(float rad);

}

class circle implements ex1

{

public float area(float rad)

{

return(pi\*rad\*rad);

}

public float perim(float rad)

{

return(2\*pi\*rad);

}

}

class testinterface

{

public static void main(String args[])

{

circle c=new circle();

System.out.print("get radius");

Scanner in=new Scanner(System.in);

int r;

r=in.nextInt();

System.out.println("area of circle="+c.area(r));

System.out.println("perimeter of circle="+c.perim(r));

}

}

**OUTPUT**

GET RADIUS : 10

AREA OF CIRCLE : 314.0

PERIMETER OF CIRCLE: 62.800003

**IMPLEMENTING FLOW, BORDER, & GRID LAYOUT**

import javax.swing.JApplet;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class layout extends JApplet {

public static void main(String[]args){

JFrame frame=new JFrame();

JPanel radioPanel=new JPanel(new FlowLayout(FlowLayout.CENTER));

JPanel GridPanel=new JPanel(new GridLayout(2,2));

JPanel BorderPanel=new JPanel(new BorderLayout());

JRadioButton button1=new JRadioButton("Button 1",true);

JRadioButton button2=new JRadioButton("Button 2",false);

JRadioButton button3=new JRadioButton("Button 3",false);

radioPanel.add(button1);

radioPanel.add(button2);

radioPanel.add(button3);

GridPanel.add(new JButton("Button1"));

GridPanel.add(new JButton("Button2"));

GridPanel.add(new JButton("Button3"));

GridPanel.add(new JButton("Button4"));

BorderPanel.add(new JButton("Button1"));

frame.add(radioPanel,BorderLayout.NORTH);

frame.add(GridPanel,BorderLayout.CENTER);

frame.add(BorderPanel,BorderLayout.SOUTH);

frame.pack();

frame.setVisible(true);}

}

**OUTPUT**

****

**IMPLEMENTING DIALOGS, MENU & FRAME**

import javax.swing.JApplet;

import java.applet.Applet;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Dialog extends JApplet implements ActionListener {

Frame f;

MenuItem mi1;

public void init(){

f=new Frame("demo Frame");

f.setSize(200,200);

MenuBar mb=new MenuBar();

f.setMenuBar(mb);

Menu file=new Menu("file");

mi1=new MenuItem("new");

file.add(mi1);

mi1.addActionListener(this);

file.add(new MenuItem("open"));

file.add(new MenuItem("close"));

file.add(new MenuItem("quit"));

mb.add(file);

Menu Edit=new Menu("edit");

Edit.add(new MenuItem("cut"));

Edit.add(new MenuItem("copy"));

Edit.add(new MenuItem("paste"));

mb.add(Edit);

f.show();}

**OUTPUT**

 ** **

**IMPLEMENTING USER DEFINED EXCEPTION HANDLING**

import java.io.\*;

import java.util.\*;

class OwnException extends Exception

{

OwnException()

{

}}

public class UserException

{

public static void main(String[] args)

{

System.out.print("Get voter's age");

Scanner in=new Scanner(System.in);

int age;

age=in.nextInt();

try

{

if(age<18)

throw new OwnException();

else

System.out.println("voter's Age is accepted");

}

catch(OwnException exp)

{

System.out.println("voter's age should be above 10");

System.out.println(exp);

}}**}**

**OUTPUT**

GET VOTER’S AGE

13

VOTER’S AGE SHOULD BE ABOVE 18

OWN EXCEPTION

GET VOTER’S AGE

45

VOTER’S AGE IS ACCEPTED

**IMPLEMENTING MULTI-THREADING**

class share extends Thread

{

static String msg[]={"this","is","a","synchronized","variable"};

share(String threadname)

{

super(threadname);

}

public void run()

{

display(getName());

}

public synchronized void display(String threadN)

{

for(int i=0;i<=4;i++)

System.out.println(threadN+msg[i]);

try

{

this.sleep(8000);

}

catch(Exception e)

{

}

}

}

public class multithreading

{

public static void main(String[]args)

{

share t1=new share("ThreadOne:");

t1.start();

share t2=new share("ThreadTwo:");

t2.start();

share t3=new share("ThreadThree:");

t3.start();

}

}

**OUTPUT**

Thread One :this

Thread One:is

Thread One:a

Thread One:synchronized

Thread One:Variable

Thread Three:this

Thread Two:this

Thread Three:is

Thread Three:a

Thread Three:synchronized

Thread Two:is

Thread Three:Variable

Thread Two:a

Thread Two:synchronized

Thread Two;Variable

**IMPLEMENTING I/O STREAM FILE HANDLING**

import java.io.\*;

import java.util.\*;

public class ex9

{

static Scanner in=new Scanner(System.in);

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

{

DataOutputStream dos=new DataOutputStream(new FileOutputStream("test.dat"));

System.out.println("how many students?");

int no=in.nextInt();

for(int i=1;i<=no;i++)

{

System.out.print("enter name :");

String name=in.next();

System.out.print("enter role no :");

int role =in.nextInt();

System.out.print("enter the mark1 :");

int m1=in.nextInt();

System.out.print("enter the mark2 :");

int m2=in.nextInt();

dos.writeUTF(name);

dos.writeInt(role);

dos.writeInt(m1);

dos.writeInt(m2);

}

dos.close();

DataInputStream dis=new DataInputStream(new FileInputStream("test.dat"));

for(int i=1;i<=no;i++)

{

String n=dis.readUTF();

int role =dis.readInt();

int mark1=dis.readInt();

int mark2=dis.readInt();

int mark3=mark1+mark2;

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

System.out.println("name :"+n);

System.out.println("role no :"+role);

System.out.println("mark1 :"+mark1);

System.out.println("mark2 :"+mark2);

System.out.println("total :"+mark3);

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

}

dis.close();

}

}

**OUTPUT**

HOW MANY STUDENTS?

3

ENTER NAME : VISHNU

ENTER ROLL NUMBER : 1001

ENTER MARK1 : 98

ENTER MARK2 : 98

ENTER NAME : SOWNDHARYA

ENTER ROLL NUMBER : 1002

ENTER MARK1 : 99

ENTER MARK2 : 99

ENTER NAME : THAMOTHARAN

ENTER ROLL NUMBER : 1003

ENTER MARK1 : 100

ENTER MARK2 : 99

NAME : VISHNU

ROLL NUMBER : 1001

MARK 1 : 98

MARK 2 : 98

TOTAL : 196

=======================================================

NAME : SOWNDHARYA

ROLL NUMBER : 1002

MARK 1 : 99

MARK 2 : 99

TOTAL : 198

========================================================

NAME : THAMOTHARAN

ROLL NUMBER : 1003

MARK 1 :100

MARK 2 : 99

TOTAL : 199

=========================================================

**IMPLEMENTING CALCULATOR USING SWING**

import javax.swing.JApplet;

import javax.swing.\*;

import javax.swing.border.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.script.ScriptEngineManager;

import javax.script.ScriptEngine;

import javax.script.ScriptException;

public class calcul extends JFrame implements ActionListener {

private JPanel buttonPanel;

private JPanel inputOutputPanel;

private JTextField info;

private boolean editable=true;

public calcul()

{

Container contentPane=getContentPane();

contentPane.setLayout(new FlowLayout());

buttonPanel=new JPanel();

buttonPanel.setLayout(new GridLayout(4,4,5,5));

inputOutputPanel=new JPanel();

inputOutputPanel.setLayout(new FlowLayout());

setSize(300,300);

setResizable(false);

info=new JTextField();

info.setPreferredSize(new Dimension(270,35));

info.addActionListener(this);

inputOutputPanel.add(info);

contentPane .add(inputOutputPanel);

String buttons[]={"7","8","9","/","4","5","6","\*","1","2","3","-","0",".","=","+",};

for(String i:buttons)

{

JButton button =new JButton(i);

button.addActionListener(this);

buttonPanel.add(button);

}

contentPane.add(buttonPanel);

}

public void actionPerformed(ActionEvent event)

{

if(event.getSource()instanceof JButton)

{

JButton clickedButton=(JButton)event.getSource();

if(clickedButton.getText().equals("="))

{

addOutput();

}

else

{

addInput(clickedButton.getText());

}

}

}

public void addInput(String line)

{

editable=true;

info.setText(info.getText()+line);

}

public void addOutput()

{

double output=0;

try

{

ScriptEngineManager mgr=new ScriptEngineManager();

ScriptEngine engine=mgr.getEngineByName("javaScript");

info.setText(engine.eval(info.getText()).toString());

}

catch(ScriptException e)

{

info.setText("Syntax error");

}

editable=false;

}

public static void main(String[]args)

{

calcul calculator=new calcul();

calculator.setVisible(true);

}

}

**OUTPUT**

****

**CRUD OPERATION USING JDBC**

import java.sql.\*;

public class CrudEx{

public static void main(String args[]){

try{

String url="jdbc:odbc:coll";

Class.forName("sun.jdbc.odbc.jdbcOdbcDriver");

Connection c=DriveManager.getConnection(url);

Statement st=c.createStatement();

ResultSet rs=st.executeQuery("select\*from stud");

while(rs.next()){

System.out.println(rs.getString(1));

System.out.println(rs.getString(2));

System.out.println(rs.getString(3));

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

}

st.executeUpdate("insert into stud values(14,'sowndharya,467)");

st.executeUpdate("Delete from stud where stName='vishnu'");

st.executeUpdate("update stud set stName='malar'where i9d=11");

rs=st.executeOuery("select\*from stud");

while(rs.next()){

System.out.println(rs.getString(1));

System.out.println(rs.getString(2));

System.out,println(rs.getString(3));

}}

catch(Exception ee){

System.out.println(ee);

}}}

**OUTPUT**

Init:

Run-single:

6

Sowndharya

567

11

Vishnu

- - - - - - - -- - - - - - - - - -- - - -------- --- - --- -- -

11

VISHNU

467

14

Malar

467