**Z SHARIK ANWAR**

**9918004110**

**JAVA PROGRAMMES**

**EXERCISE 10:**

**1.Write a temperature conversion applet that converts from Fahrenheit to Celsius. The Fahrenheit temperature should be entered from the keyboard (via a JTextField). A JLabel should be used to display the converted temperature. Use the following formula for the conversion: Celcius = ((5/9)\*(Ferenheit‐32)). Enhance the temperature conversion applet of Q1 by adding the Kelvin temperature scale. The applet should also allow the user to make conversions between any two scales. Use the following formula for the conversion between Kelvin and Celsius (in addition to the formulain Q1): Kelvin = Celcius + 273.15**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.text.\*;

public class TempCon extends JApplet implements ActionListener {

JTextField txtInput;

JLabel lblResult;

JRadioButton rbCelcius, rbKelvin;

public void init(){

Container conpane = getContentPane();

conpane.setLayout (new FlowLayout());

txtInput = new JTextField("",10);

conpane.add(txtInput);

rbCelcius= new JRadioButton ("to Celcius", true);

conpane.add(rbCelcius);

rbKelvin = new JRadioButton("to Kelvin", false);

conpane.add(rbKelvin);

ButtonGroup selection = new ButtonGroup();

selection.add(rbCelcius);

selection.add(rbKelvin);

JButton button1 = new JButton ("Show Result");

button1.addActionListener(this);

conpane.add(button1);

lblResult= new JLabel ("Enter Ferenheit, Choose an option to convert and Click Show Result");

conpane.add(lblResult);

}

public void actionPerformed(ActionEvent e) {

DecimalFormat df = new DecimalFormat ("#.##");

double ferenheit = Double.parseDouble(txtInput.getText());

double answer = 0.0;

answer = ((5.0/9.0)\*(ferenheit - 32.0));

if (rbKelvin.isSelected())

answer += 273.15;

lblResult.setText(String.valueOf(df.format(answer)));

}

}

**2. Calculator using AWT controls**

import java.awt.\*;

import java.awt.event.\*;

public class calculator implements ActionListener

{

int c,n;

String s1,s2,s3,s4,s5;

Frame f;

Button b1,b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,b14,b15,b16,b17;

Panel p;

TextField tf;

GridLayout g;

calculator()

{

f = new Frame("My calculator");

p = new Panel();

f.setLayout(new FlowLayout());

b1 = new Button("0");

b1.addActionListener(this);

b2 = new Button("1");

b2.addActionListener(this);

b3 = new Button("2");

b3.addActionListener(this);

b4 = new Button("3");

b4.addActionListener(this);

b5 = new Button("4");

b5.addActionListener(this);

b6 = new Button("5");

b6.addActionListener(this);

b7 = new Button("6");

b7.addActionListener(this);

b8 = new Button("7");

b8.addActionListener(this);

b9 = new Button("8");

b9.addActionListener(this);

b10 = new Button("9");

b10.addActionListener(this);

b11 = new Button("+");

b11.addActionListener(this);

b12 = new Button("-");

b12.addActionListener(this);

b13 = new Button("\*");

b13.addActionListener(this);

b14 = new Button("/");

b14.addActionListener(this);

b15 = new Button("%");

b15.addActionListener(this);

b16 = new Button("=");

b16.addActionListener(this);

b17 = new Button("C");

b17.addActionListener(this);

tf = new TextField(20);

f.add(tf);

g = new GridLayout(4,4,10,20);

p.setLayout(g);

p.add(b1);p.add(b2);p.add(b3);p.add(b4);p.add(b5);p.add(b6);p.add(b7);p.add(b8);p.add(b9);

p.add(b10);p.add(b11);p.add(b12);p.add(b13);p.add(b14);p.add(b15);p.add(b16);p.add(b17);

f.add(p);

f.setSize(300,300);

f.setVisible(true);

}

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==b1)

{

s3 = tf.getText();

s4 = "0";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b2)

{

s3 = tf.getText();

s4 = "1";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b3)

{

s3 = tf.getText();

s4 = "2";

s5 = s3+s4;

tf.setText(s5);

}if(e.getSource()==b4)

{

s3 = tf.getText();

s4 = "3";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b5)

{

s3 = tf.getText();

s4 = "4";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b6)

{

s3 = tf.getText();

s4 = "5";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b7)

{

s3 = tf.getText();

s4 = "6";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b8)

{

s3 = tf.getText();

s4 = "7";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b9)

{

s3 = tf.getText();

s4 = "8";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b10)

{

s3 = tf.getText();

s4 = "9";

s5 = s3+s4;

tf.setText(s5);

}

if(e.getSource()==b11)

{

s1 = tf.getText();

tf.setText("");

c=1;

}

if(e.getSource()==b12)

{

s1 = tf.getText();

tf.setText("");

c=2;

}

if(e.getSource()==b13)

{

s1 = tf.getText();

tf.setText("");

c=3;

}

if(e.getSource()==b14)

{

s1 = tf.getText();

tf.setText("");

c=4;

}

if(e.getSource()==b15)

{

s1 = tf.getText();

tf.setText("");

c=5;

}

if(e.getSource()==b16)

{

s2 = tf.getText();

if(c==1)

{

n = Integer.parseInt(s1)+Integer.parseInt(s2);

tf.setText(String.valueOf(n));

}

else

if(c==2)

{

n = Integer.parseInt(s1)-Integer.parseInt(s2);

tf.setText(String.valueOf(n));

}

else

if(c==3)

{

n = Integer.parseInt(s1)\*Integer.parseInt(s2);

tf.setText(String.valueOf(n));

}

if(c==4)

{

try

{

int p=Integer.parseInt(s2);

if(p!=0)

{

n = Integer.parseInt(s1)/Integer.parseInt(s2);

tf.setText(String.valueOf(n));

}

else

tf.setText("infinite");

}

catch(Exception i){}

}

if(c==5)

{

n = Integer.parseInt(s1)%Integer.parseInt(s2);

tf.setText(String.valueOf(n));

}

}

if(e.getSource()==b17)

{

tf.setText("");

}

}

public static void main(String[] abc)

{

calculator v = new calculator();

}

**3. Authentication check using AWT controls**

|  |
| --- |
| import java.awt.\*;  import java.awt.event.\*;  class MyLoginWindow extends Frame  {  TextField name,pass;  Button b1,b2;  MyLoginWindow()  {  setLayout(new FlowLayout());  this.setLayout(null);  Label n=new Label("Name:",Label.CENTER);  Label p=new Label("password:",Label.CENTER);  name=new TextField(20);  pass=new TextField(20);  pass.setEchoChar('#');  b1=new Button("submit");  b2=new Button("cancel");  this.add(n);  this.add(name);  this.add(p);  this.add(pass);  this.add(b1);  this.add(b2);  n.setBounds(70,90,90,60);  p.setBounds(70,130,90,60);  name.setBounds(200,100,90,20);  pass.setBounds(200,140,90,20);  b1.setBounds(100,260,70,40);  b2.setBounds(180,260,70,40);  }  public static void main(String args[])  {  MyLoginWindow ml=new MyLoginWindow();  ml.setVisible(true);  ml.setSize(400,400);  ml.setTitle("my login window");  }  } |

**4. Design a calculator using event-driven programming paradigm of Java with the following options. a) Decimal manipulations b) Scientific manipulations**

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

import javax.swing.event.\*;

public class ScientificCalculator extends JFrame implements ActionListener

{

JTextField tfield;

double temp, temp1, result, a;

static double m1, m2;

int k = 1, x = 0, y = 0, z = 0;

char ch;

JButton b1, b2, b3, b4, b5, b6, b7, b8, b9, zero, clr, pow2, pow3, exp,

fac, plus, min, div, log, rec, mul, eq, addSub, dot, mr, mc, mp,

mm, sqrt, sin, cos, tan;

Container cont;

JPanel textPanel, buttonpanel;

ScientificCalculator()

{

cont = getContentPane();

cont.setLayout(new BorderLayout());

JPanel textpanel = new JPanel();

tfield = new JTextField(25);

tfield.setHorizontalAlignment(SwingConstants.RIGHT);

tfield.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent keyevent) {

char c = keyevent.getKeyChar();

if (c >= ‘0’ && c <= ‘9’) {

}

else

{

keyevent.consume();

}

}

});

textpanel.add(tfield);

buttonpanel = new JPanel();

buttonpanel.setLayout(new GridLayout(8, 4, 2, 2));

boolean t = true;

mr = new JButton(“MR”);

buttonpanel.add(mr);

mr.addActionListener(this);

mc = new JButton(“MC”);

buttonpanel.add(mc);

mc.addActionListener(this);

mp = new JButton(“M+”);

buttonpanel.add(mp);

mp.addActionListener(this);

mm = new JButton(“M-“);

buttonpanel.add(mm);

mm.addActionListener(this);

b1 = new JButton(“1”);

buttonpanel.add(b1);

b1.addActionListener(this);

b2 = new JButton(“2”);

buttonpanel.add(b2);

b2.addActionListener(this);

b3 = new JButton(“3”);

buttonpanel.add(b3);

b3.addActionListener(this);

b4 = new JButton(“4”);

buttonpanel.add(b4);

b4.addActionListener(this);

b5 = new JButton(“5”);

buttonpanel.add(b5);

b5.addActionListener(this);

b6 = new JButton(“6”);

buttonpanel.add(b6);

b6.addActionListener(this);

b7 = new JButton(“7”);

buttonpanel.add(b7);

b7.addActionListener(this);

b8 = new JButton(“8”);

buttonpanel.add(b8);

b8.addActionListener(this);

b9 = new JButton(“9”);

buttonpanel.add(b9);

b9.addActionListener(this);

zero = new JButton(“0”);

buttonpanel.add(zero);

zero.addActionListener(this);

plus = new JButton(“+”);

buttonpanel.add(plus);

plus.addActionListener(this);

min = new JButton(“-“);

buttonpanel.add(min);

min.addActionListener(this);

mul = new JButton(“\*”);

buttonpanel.add(mul);

mul.addActionListener(this);

div = new JButton(“/”);

div.addActionListener(this);

buttonpanel.add(div);

addSub = new JButton(“+/-“);

buttonpanel.add(addSub);

addSub.addActionListener(this);

dot = new JButton(“.”);

buttonpanel.add(dot);

dot.addActionListener(this);

eq = new JButton(“=”);

buttonpanel.add(eq);

eq.addActionListener(this);

rec = new JButton(“1/x”);

buttonpanel.add(rec);

rec.addActionListener(this);

sqrt = new JButton(“Sqrt”);

buttonpanel.add(sqrt);

sqrt.addActionListener(this);

log = new JButton(“log”);

buttonpanel.add(log);

log.addActionListener(this);

sin = new JButton(“SIN”);

buttonpanel.add(sin);

sin.addActionListener(this);

cos = new JButton(“COS”);

buttonpanel.add(cos);

cos.addActionListener(this);

tan = new JButton(“TAN”);

buttonpanel.add(tan);

tan.addActionListener(this);

pow2 = new JButton(“x^2”);

buttonpanel.add(pow2);

pow2.addActionListener(this);

pow3 = new JButton(“x^3”);

buttonpanel.add(pow3);

pow3.addActionListener(this);

exp = new JButton(“Exp”);

exp.addActionListener(this);

buttonpanel.add(exp);

fac = new JButton(“n!”);

fac.addActionListener(this);

buttonpanel.add(fac);

clr = new JButton(“AC”);

buttonpanel.add(clr);

clr.addActionListener(this);

cont.add(“Center”, buttonpanel);

cont.add(“North”, textpanel);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

public void actionPerformed(ActionEvent e)

{

String s = e.getActionCommand();

if (s.equals(“1”))

{

if (z == 0)

{

tfield.setText(tfield.getText() + “1”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “1”);

z = 0;

}

}

if (s.equals(“2”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “2”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “2”);

z = 0;

}

}

if (s.equals(“3”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “3”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “3”);

z = 0;

}

}

if (s.equals(“4”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “4”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “4”);

z = 0;

}

}

if (s.equals(“5”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “5”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “5”);

z = 0;

}

}

if (s.equals(“6”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “6”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “6”);

z = 0;

}

}

if (s.equals(“7”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “7”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “7”);

z = 0;

}

}

if (s.equals(“8”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “8”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “8”);

z = 0;

}

}

if (s.equals(“9”)) {

if (z == 0) {

tfield.setText(tfield.getText() + “9”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “9”);

z = 0;

}

}

if (s.equals(“0”))

{

if (z == 0) {

tfield.setText(tfield.getText() + “0”);

}

else

{

tfield.setText(“”);

tfield.setText(tfield.getText() + “0”);

z = 0;

}

}

if (s.equals(“AC”)) {

tfield.setText(“”);

x = 0;

y = 0;

z = 0;

}

if (s.equals(“log”))

{

if (tfield.getText().equals(“”)) {

tfield.setText(“”);

}

else

{

a = Math.log(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“1/x”)) {

if (tfield.getText().equals(“”)) {

tfield.setText(“”);

}

else

{

a = 1 / Double.parseDouble(tfield.getText());

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“Exp”)) {

if (tfield.getText().equals(“”)) {

tfield.setText(“”);

}

else

{

a = Math.exp(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“x^2”)) {

if (tfield.getText().equals(“”)) {

tfield.setText(“”);

}

else

{

a = Math.pow(Double.parseDouble(tfield.getText()), 2);

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“x^3”)) {

if (tfield.getText().equals(“”)) {

tfield.setText(“”);

}

else

{

a = Math.pow(Double.parseDouble(tfield.getText()), 3);

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“+/-“)) {

if (x == 0) {

tfield.setText(“-” + tfield.getText());

x = 1;

}

else

{

tfield.setText(tfield.getText());

}

}

if (s.equals(“.”)) {

if (y == 0) {

tfield.setText(tfield.getText() + “.”);

y = 1;

}

else

{

tfield.setText(tfield.getText());

}

}

if (s.equals(“+”))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

temp = 0;

ch = ‘+’;

}

else

{

temp = Double.parseDouble(tfield.getText());

tfield.setText(“”);

ch = ‘+’;

y = 0;

x = 0;

}

tfield.requestFocus();

}

if (s.equals(“-“))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

temp = 0;

ch = ‘-‘;

}

else

{

x = 0;

y = 0;

temp = Double.parseDouble(tfield.getText());

tfield.setText(“”);

ch = ‘-‘;

}

tfield.requestFocus();

}

if (s.equals(“/”)) {

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

temp = 1;

ch = ‘/’;

}

else

{

x = 0;

y = 0;

temp = Double.parseDouble(tfield.getText());

ch = ‘/’;

tfield.setText(“”);

}

tfield.requestFocus();

}

if (s.equals(“\*”)) {

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

temp = 1;

ch = ‘\*’;

}

else

{

x = 0;

y = 0;

temp = Double.parseDouble(tfield.getText());

ch = ‘\*’;

tfield.setText(“”);

}

tfield.requestFocus();

}

if (s.equals(“MC”))

{

m1 = 0;

tfield.setText(“”);

}

if (s.equals(“MR”))

{

tfield.setText(“”);

tfield.setText(tfield.getText() + m1);

}

if (s.equals(“M+”))

{

if (k == 1) {

m1 = Double.parseDouble(tfield.getText());

k++;

}

else

{

m1 += Double.parseDouble(tfield.getText());

tfield.setText(“” + m1);

}

}

if (s.equals(“M-“))

{

if (k == 1) {

m1 = Double.parseDouble(tfield.getText());

k++;

}

else

{

m1 -= Double.parseDouble(tfield.getText());

tfield.setText(“” + m1);

}

}

if (s.equals(“Sqrt”))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

}

else

{

a = Math.sqrt(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

field.setText(tfield.getText() + a);

}

}

if (s.equals(“SIN”))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

}

else

{

a = Math.sin(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“COS”))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

}

else

{

a = Math.cos(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“TAN”)) {

if (tfield.getText().equals(“”)) {

tfield.setText(“”);

}

else

{

a = Math.tan(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

if (s.equals(“=”))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

}

else

{

temp1 = Double.parseDouble(tfield.getText());

switch (ch)

{

case ‘+’:

result = temp + temp1;

break;

case ‘-‘:

result = temp – temp1;

break;

case ‘/’:

result = temp / temp1;

break;

case ‘\*’:

result = temp \* temp1;

break;

}

tfield.setText(“”);

tfield.setText(tfield.getText() + result);

z = 1;

}

}

if (s.equals(“n!”))

{

if (tfield.getText().equals(“”))

{

tfield.setText(“”);

}

else

{

a = fact(Double.parseDouble(tfield.getText()));

tfield.setText(“”);

tfield.setText(tfield.getText() + a);

}

}

tfield.requestFocus();

}

double fact(double x)

{

int er = 0;

if (x < 0)

{

er = 20;

return 0;

}

double i, s = 1;

for (i = 2; i <= x; i += 1.0)

s \*= i;

return s;

}

public static void main(String args[])

{

try

{

UIManager.setLookAndFeel(“com.sun.java.swing.plaf.windows.WindowsLookAndFeel”);

}

catch (Exception e)

{

}

ScientificCalculator f = new ScientificCalculator();

f.setTitle(“ScientificCalculator”);

f.pack();

f.setVisible(true);

} }