**Phuong Ho**

**011056693**

**CECS277**

**Lab 6**

**Question 0.**

**public** **class** BankAccount {

**private** **double** balance;

BankAccount(){

balance = 0.0;

}

BankAccount(**double** bl){

balance = bl;

}

**public** **void** setBalance(**double** bl){

balance = bl;

}

**public** **double** getBalance(){

**return** balance;

}

**public** **void** deposit(**double** amount){

balance += amount;

}

**public** **void** withdraw(**double** amount){

balance -= amount;

}

}

import java.awt.Component;

import java.awt.Dimension;

import java.awt.Toolkit;

import java.awt.Window;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.AbstractButton;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JTextField;

public class BankAccountF extends JFrame{

static BankAccount a = new BankAccount(400.0);

private JLabel amountLabel;

private JTextField amountField;

private JLabel outputLabel;

BankAccountF(BankAccount ba){

setTitle("Title");

setSize(400, 100);

centerWindow(this);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

a = ba;

JPanel controlPanel = new JPanel();

amountLabel = new JLabel("Amount");

controlPanel.add(amountLabel);

amountField = new JTextField(7);

controlPanel.add(amountField);

controlPanel.add(createWButton());

controlPanel.add(createDButton());

outputLabel = new JLabel("Balane = " + ba.getBalance());

controlPanel.add(outputLabel);

this.add(controlPanel);

}

private JButton createWButton() {

JButton wButton = new JButton("Withdraw");

class WListener implements ActionListener

{

@Override

public void actionPerformed(ActionEvent e) {

double amount = Double.parseDouble(amountField.getText());

a.withdraw(amount);

outputLabel.setText("Balance = $ " + a.getBalance());

}

}

ActionListener l1 = new WListener();

wButton.addActionListener(l1);

return wButton;

}

private JButton createDButton() {

JButton dButton = new JButton("Deposit");

class DListener implements ActionListener{

@Override

public void actionPerformed(ActionEvent e) {

double amount = Double.parseDouble(amountField.getText());

a.deposit(amount);

outputLabel.setText("Balance = $ " + a.getBalance());

}

}

ActionListener l1 = new DListener();

dButton.addActionListener(l1);

return dButton;

}

private void centerWindow(Window w) {

Toolkit tk = Toolkit.getDefaultToolkit();

Dimension d = tk.getScreenSize();

setLocation((d.width - w.getWidth()) / 2, (d.height - w.getHeight()) / 2);

}

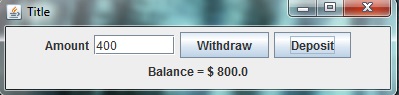
public static void main(String[] args) {

JFrame f = new BankAccountF(a);

f.setVisible(true);

}

}



**Question 1.**

**public** **class** SavingAccount {

**private** **double** balance;

SavingAccount(){

balance = 0.00;

}

SavingAccount(**double** bl){

balance = bl;

}

**public** **void** setBalance(**double** bl){

balance = bl;

}

**public** **double** getBalance(){

**return** balance;

}

**public** **double** yearlyBalance(**double** initialBalance, **double** anualRate){

**double** interest = initialBalance \* anualRate / 100;

**double** result = initialBalance += interest;

**return** result;

}

}

import java.awt.BorderLayout;

import java.awt.Component;

import java.awt.Dimension;

import java.awt.FlowLayout;

import java.awt.Toolkit;

import java.awt.Window;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.text.DecimalFormat;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JTextArea;

import javax.swing.JTextField;

import javax.swing.ScrollPaneConstants;

public class SavingAccountF extends JFrame{

private JLabel initBalanceLabel, anualRateLabel, numberYearsLabel;

private static JTextField initBalanceField;

private JTextField anualRateField;

private JTextField numberYearsField;

private static SavingAccount sa = new SavingAccount();

private JTextArea balanceArea;

SavingAccountF(SavingAccount a){

setTitle("SavingsCalculator");

setSize(260, 400);

centerWindow(this);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

sa = a;

initBalanceLabel = new JLabel("Initial Balance");

this.add(initBalanceLabel, BorderLayout.WEST);

initBalanceField = new JTextField(9);

this.add(initBalanceField, BorderLayout.EAST);

anualRateLabel = new JLabel("Anual Rate");

this.add(anualRateLabel, BorderLayout.WEST);

anualRateField = new JTextField(9);

this.add(anualRateField, BorderLayout.EAST);

numberYearsLabel = new JLabel("Number of Years");

this.add(numberYearsLabel, BorderLayout.WEST);

numberYearsField = new JTextField(9);

this.add(numberYearsField, BorderLayout.EAST);

JPanel calcPanel = new JPanel();

calcPanel.setLayout(new FlowLayout(FlowLayout.CENTER));

calcPanel.add(createButton());

balanceArea = new JTextArea(10,8);

balanceArea.setLineWrap(true);

balanceArea.setWrapStyleWord(true);

JScrollPane balanceScroll = new JScrollPane(balanceArea,

ScrollPaneConstants.VERTICAL\_SCROLLBAR\_ALWAYS,

ScrollPaneConstants.HORIZONTAL\_SCROLLBAR\_NEVER);

calcPanel.add(balanceScroll);

this.setLayout(new FlowLayout(FlowLayout.CENTER));

this.add(calcPanel);

}

private JButton createButton() {

JButton cButton = new JButton("Calculate");

class Listener implements ActionListener{

@Override

public void actionPerformed(ActionEvent e) {

double initBalance = Double.parseDouble(initBalanceField.getText());

double anualRate = Double.parseDouble(anualRateField.getText());

double numberYears = Double.parseDouble(numberYearsField.getText());

for (int year = 1; year < numberYears; year++){

double balance = sa.yearlyBalance(initBalance, anualRate);

DecimalFormat df = new DecimalFormat("#,###,###,##0.00");

String bal = df.format(balance);

balanceArea.append("$" + bal + "\n");

initBalance = balance;

}

}

}

ActionListener l = new Listener();

cButton.addActionListener(l);

return cButton;

}

private void centerWindow(Window w) {

Toolkit tk = Toolkit.getDefaultToolkit();

Dimension d = tk.getScreenSize();

setLocation((d.width - w.getWidth()) / 2, (d.height - w.getHeight()) / 2);

}

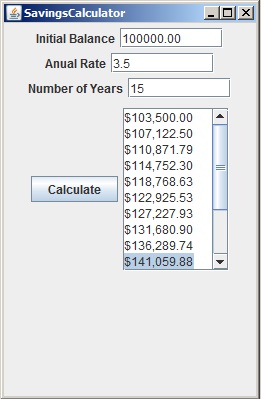
public static void main(String[] args) {

JFrame f = new SavingAccountF(sa);

f.setVisible(true);

}

}



**Question 1:**

import java.awt.BorderLayout;

import java.awt.Dimension;

import java.awt.FlowLayout;

import java.awt.GridLayout;

import java.awt.Toolkit;

import java.awt.Window;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JTextField;

public class Calculator extends JFrame implements ActionListener{

private JButton button1, button2, button3, button4, button5, button6, button7, button8, button9, button0;

private JButton buttonAdd, buttonSub, buttonMul, buttonDiv, buttonEqual, buttonDec,buttonClear;

private JTextField screenField;

private int clearField, add = 0, sub = 0, mul = 0, div = 0;

double number1, number2, temp;

public Calculator(){

setTitle("Calculator");

setResizable(false);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(250,200);

JPanel p2 = new JPanel();

p2.setLayout(new FlowLayout());

p2.add(screenField = new JTextField(21));

screenField.setEditable(false);

JPanel p1 = new JPanel();

p1.setLayout(new GridLayout(4, 4));

p1.add(button7 = new JButton("7"));

p1.add(button8 = new JButton("8"));

p1.add(button9 = new JButton("9"));

p1.add(buttonDiv = new JButton("/"));

p1.add(button4 = new JButton("4"));

p1.add(button5 = new JButton("5"));

p1.add(button6 = new JButton("6"));

p1.add(buttonMul = new JButton("\*"));

p1.add(button1 = new JButton("1"));

p1.add(button2 = new JButton("2"));

p1.add(button3 = new JButton("3"));

p1.add(buttonSub = new JButton("-"));

p1.add(button0 = new JButton("0"));

p1.add(buttonDec = new JButton("."));

p1.add(buttonEqual = new JButton("="));

p1.add(buttonAdd = new JButton("+"));

JPanel p3 = new JPanel();

p3.setLayout(new FlowLayout());

p3.add(buttonClear = new JButton("C"));

JPanel p = new JPanel();

p.setLayout(new BorderLayout());

p.add(p2, BorderLayout.NORTH);

p.add(p1, BorderLayout.CENTER);

p.add(p3, BorderLayout.EAST);

this.add(p);

button0.addActionListener(this);

button1.addActionListener(this);

button2.addActionListener(this);

button3.addActionListener(this);

button4.addActionListener(this);

button5.addActionListener(this);

button6.addActionListener(this);

button7.addActionListener(this);

button8.addActionListener(this);

button9.addActionListener(this);

buttonAdd.addActionListener(this);

buttonSub.addActionListener(this);

buttonMul.addActionListener(this);

buttonDiv.addActionListener(this);

buttonDec.addActionListener(this);

buttonClear.addActionListener(this);

buttonEqual.addActionListener(this);

}

@Override

public void actionPerformed(ActionEvent e) {

if (e.getSource() == button0){

if(clearField == 1){

screenField.setText("0");

clearField = 0;

}else screenField.setText(screenField.getText() + "0");

}

if (e.getSource() == button1){

if(clearField == 1){

screenField.setText("1");

clearField = 0;

}else screenField.setText(screenField.getText() + "1");

}

if (e.getSource() == button2){

if(clearField == 1){

screenField.setText("2");

clearField = 0;

}else screenField.setText(screenField.getText() + "2");

}

if (e.getSource() == button3){

if(clearField == 1){

screenField.setText("3");

clearField = 0;

}else screenField.setText(screenField.getText() + "3");

}

if (e.getSource() == button4){

if(clearField == 1){

screenField.setText("4");

clearField = 0;

}else screenField.setText(screenField.getText() + "4");

}

if (e.getSource() == button5){

if(clearField == 1){

screenField.setText("5");

clearField = 0;

}else screenField.setText(screenField.getText() + "5");

}

if (e.getSource() == button6){

if(clearField == 1){

screenField.setText("6");

clearField = 0;

}else screenField.setText(screenField.getText() + "6");

}

if (e.getSource() == button7){

if(clearField == 1){

screenField.setText("7");

clearField = 0;

}else screenField.setText(screenField.getText() + "7");

}

if (e.getSource() == button8){

if(clearField == 1){

screenField.setText("8");

clearField = 0;

}else screenField.setText(screenField.getText() + "8");

}

if (e.getSource() == button9){

if(clearField == 1){

screenField.setText("9");

clearField = 0;

}else screenField.setText(screenField.getText() + "9");

}

if (e.getSource() == buttonAdd)

{

if(screenField.getText() != null)

{

number1 = Double.parseDouble(String.valueOf(screenField.getText()));

add = 1;

clearField = 1;

}

}

if(e.getSource() == buttonSub){

if(screenField.getText() != null){

number1 = Double.parseDouble(String.valueOf(screenField.getText()));

sub = 1;

clearField = 1;

}

}

if(e.getSource() == buttonMul){

if(screenField.getText() != null){

number1 = Double.parseDouble(String.valueOf(screenField.getText()));

mul = 1;

clearField = 1;

}

}

if(e.getSource() == buttonDiv){

if(screenField.getText() != null){

number1 = Double.parseDouble(String.valueOf(screenField.getText()));

div = 1;

clearField = 1;

}

}

if(e.getSource() == buttonClear){

screenField.setText("");

number1 = 0;

number2 = 0;

add = 0;

sub = 0;

mul = 0;

div = 0;

}

if(e.getSource() == buttonEqual){

number2 = Double.parseDouble(String.valueOf(screenField.getText()));

screenField.setText(screenField.getText());

if(add == 1){

temp = number1 + number2;

screenField.setText(String.valueOf(temp));

add = 0;

}

if(sub == 1){

temp = number1 - number2;

screenField.setText(String.valueOf(temp));

sub = 0;

}

if(mul == 1){

temp = number1 \* number2;

screenField.setText(String.valueOf(temp));

mul = 0;

}

if(div == 1){

temp = number1 / number2;

screenField.setText(String.valueOf(temp));

div = 0;

}

}

if (e.getSource() == buttonDec){

if (screenField.getText().contains("."))

screenField.setText(screenField.getText());

screenField.setText(screenField.getText() + ".");

}

}

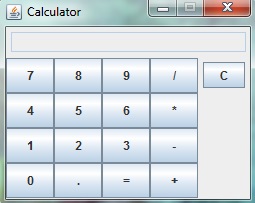
public static void main(String[] args){

JFrame f = new Calculator();

f.setVisible(true);

}

}



**Question 2:**

**import** javax.swing.JFrame;

**public** **class** Main {

**public** **static** **void** main(String[] args){

JFrame f = **new** TableFrame();

f.setVisible(**true**);

}

}

**import** java.awt.\*;

**import** javax.swing.\*;

**public** **class** TableMessage **extends** JPanel{

**int** x,y;

**public** **void** paintComponent(Graphics g){

FontMetrics fm = g.getFontMetrics();

**int** sw = fm.stringWidth("Phuong Ho");

x = getWidth()/2-sw/2;

y = getHeight()/2-fm.getAscent()/2;

g.drawString("Phuong Ho",x,y);

}

}

**import** javax.swing.JFrame;

**import** javax.swing.JPanel;

**public** **class** TableFrame **extends** JFrame{

TableFrame(){

setTitle("Table");

setSize(500, 200);

setResizable(**false**);

setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

JPanel panel = **new** TablePanel();

**this**.add(panel);

}

}

import java.awt.BorderLayout;

import java.awt.Color;

import java.awt.Font;

import java.awt.FontMetrics;

import java.awt.Graphics;

import java.awt.GraphicsEnvironment;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.ButtonGroup;

import javax.swing.JCheckBox;

import javax.swing.JComboBox;

import javax.swing.JPanel;

import javax.swing.JRadioButton;

public class TablePanel extends JPanel implements ActionListener{

private int x,y;

private int size;

private String itemS;

private String strS;

private String c;

private boolean styleB;

private boolean styleI;

private boolean styleBI;

private boolean center;

private JComboBox fontCombBox, sizeCombBox;

private JCheckBox boldCheckBut, italicCheckBut, centerCheckBut;

private JRadioButton blueRadBut, redRadBut, greenRadBut;

//Get font

GraphicsEnvironment e = GraphicsEnvironment.getLocalGraphicsEnvironment();

String f[] = e.getAvailableFontFamilyNames();

String s[] = {"8","10","12","14","16","18","20", "22", "24"};

TablePanel(){

//JPanel

JPanel colorPanel = new JPanel();

JPanel stylePanel = new JPanel();

JPanel font = new JPanel();

JPanel size = new JPanel();

//Create Components

blueRadBut = new JRadioButton("Blue",false);

redRadBut = new JRadioButton("Red",false);

greenRadBut = new JRadioButton("Green",false);

boldCheckBut = new JCheckBox("Bold",false);

italicCheckBut = new JCheckBox("Italic",false);

centerCheckBut = new JCheckBox("Center",false);

fontCombBox = new JComboBox(f);

sizeCombBox = new JComboBox(s);

//Add

colorPanel.add(fontCombBox);

ButtonGroup bgColor = new ButtonGroup();

bgColor.add(redRadBut);

bgColor.add(greenRadBut);

bgColor.add(blueRadBut);

colorPanel.setLayout(new GridLayout(1,3));

colorPanel.add(redRadBut);

colorPanel.add(greenRadBut);

colorPanel.add(blueRadBut);

colorPanel.add(sizeCombBox);

stylePanel.setLayout(new GridLayout(1,3));

stylePanel.add(boldCheckBut);

stylePanel.add(italicCheckBut);

stylePanel.add(centerCheckBut);

//Add ActionListener

fontCombBox.addActionListener(this);

sizeCombBox.addActionListener(this);

blueRadBut.addActionListener(this);

redRadBut.addActionListener(this);

greenRadBut.addActionListener(this);

boldCheckBut.addActionListener(this);

italicCheckBut.addActionListener(this);

centerCheckBut.addActionListener(this);

//Layout

this.setLayout(new BorderLayout());

this.add(colorPanel, BorderLayout.NORTH);

this.add(stylePanel, BorderLayout.SOUTH);

}

public void paintComponent(Graphics g){

//Draw Message

super.paintComponent(g);

FontMetrics fm = g.getFontMetrics();

int sw = fm.stringWidth("Phuong Ho");

Color red = new Color(255,0,0);

Color green = new Color(0,255,0);

Color blue = new Color(0,0,255);

Font plain = new Font(itemS,Font.PLAIN,size);

Font bold = new Font(itemS,Font.BOLD,size);

Font italic = new Font(itemS,Font.ITALIC,size);

Font bi = new Font(itemS,Font.BOLD | Font.ITALIC,size);

if(c == "b")

g.setColor(blue);

else if(c == "g")

g.setColor(green);

else if(c == "r")

g.setColor(red);

if(styleB == true)

g.setFont(bold);

if(styleI == true)

g.setFont(italic);

if(styleBI == true)

g.setFont(bi);

if(styleB == false && styleI == false && styleBI == false)

g.setFont(plain);

if(center == true){

x = getWidth()/2-sw/2;

y = getHeight()/2-fm.getAscent()/2;

}

if(center == false){

x = 0;

y = getHeight()/2-fm.getAscent()/2;

}

g.drawString("Phuong Ho",x,y);

}

public void actionPerformed(ActionEvent e){

Object o = e.getSource();

//Update Message

if(o == fontCombBox){

Object item = fontCombBox.getSelectedItem();

itemS = item.toString();

repaint();

}

if(o == sizeCombBox){

Object str = sizeCombBox.getSelectedItem();

size = Integer.parseInt(str.toString());

repaint();

}

if(blueRadBut.isSelected()){

c = "b";

repaint();

}

if(redRadBut.isSelected()){

c = "r";

repaint();

}

if(greenRadBut.isSelected()){

c = "g";

repaint();

}

if(boldCheckBut.isSelected() && italicCheckBut.isSelected()){

styleBI = true;

styleB = false;

styleI = false;

repaint();

}

if(boldCheckBut.isSelected() == false && italicCheckBut.isSelected() == false){

styleBI = false;

styleB = false;

styleI = false;

repaint();

}

if(boldCheckBut.isSelected() == false && italicCheckBut.isSelected()){

styleBI = false;

styleB = false;

styleI = true;

repaint();

}

if(boldCheckBut.isSelected() && italicCheckBut.isSelected() == false){

styleBI = false;

styleB = true;

styleI = false;

repaint();

}

if(centerCheckBut.isSelected()){

center = true;

repaint();

}

if(centerCheckBut.isSelected() == false){

center = false;

repaint();

}

}

}



**Question 3:**

**import** javax.swing.JFrame;

**public** **class** MainClock {

**public** **static** **void** main(String[] args){

JFrame f = **new** ClockFrame();

f.setVisible(**true**);

}

}

import javax.swing.JFrame;

import javax.swing.JPanel;

public class ClockFrame extends JFrame{

ClockFrame(){

setTitle("Clock");

setSize(500,500);

setResizable(false);

setLocation(0,0);

JPanel panel = new clockPanel();

this.add(panel);

}

}

import java.awt.\*;

import java.awt.event.\*;

import java.awt.geom.\*;

import javax.swing.\*;

public class Clock\_Panel extends JPanel implements ActionListener{

private double r = 250;

private double hours;

private double minutes;

private double minuteLength = r\*.8;

private double hourLength = r\*.8;

private JLabel hL, mL;

private JTextField hTF, mTF;

private JButton dB;

Clock\_Panel(){

//setPreferredSize(new Dimension(2\*r+1,2\*r+1));

//JPanel

JPanel display = new JPanel();

//Create Components

hL = new JLabel("Hour = ");

mL = new JLabel("Minute = ");

hTF = new JTextField(10);

mTF = new JTextField(10);

dB = new JButton("Draw");

//Add Components

display.add(hL);

display.add(hTF);

display.add(mL);

display.add(mTF);

display.add(dB);

//ActionListener

dB.addActionListener(this);

this.setLayout(new BorderLayout());

this.add(display, BorderLayout.NORTH);

}

public void paintComponent(Graphics g){

//Circle

super.paintComponent(g);

Graphics2D g2 = (Graphics2D) g;

Ellipse2D.Double circle = new Ellipse2D.Double(50,50,1.6\*r,1.6\*r);

g2.draw(circle);

//draw hour

double hourAngle = Math.toRadians(90 - 360 \* minutes / (12 \* 60));

draw(g2, hourAngle, hourLength);

//draw minute

double minuteAngle = Math.toRadians(90 - 360 \* minutes / 60);

draw(g2, minuteAngle, minuteLength);

}

public void draw(Graphics2D g2, double angle, double length){

Point2D.Double center = new Point2D.Double(r,r);

Point2D.Double p = new Point2D.Double(r+length\*Math.cos(angle),r-length\*Math.sin(angle));

Line2D.Double line = new Line2D.Double(center, p);

g2.draw(line);

}

public void actionPerformed(ActionEvent e){

Object o = e.getSource();

if(o == dB){

hours = Integer.parseInt(hTF.getText());

minutes = Integer.parseInt(mTF.getText());

minutes = minutes + hours \* 60;

repaint();

}

}

}

