Міністерство освіти і науки України

Черкаський державний технологічний університет

Кафедра програмного забезпечення автоматизованих систем

Звіт

З лабораторної роботи №7

З предмету «Об’єктно орієнтоване програмування»

Виконав

Студент ІІ курсу

Група ПЗ-154

Линник В.Ю.

Перевірив

Викладач кафедри

Програмного забезпечення автоматизованих систем

Крайовий В. М.

Виконав

Студент І курсу

Група ПЗ-154

Кравченко Артем Олегович

ЧЕРКАСИ 2016

**Тема:** Колекції. RTTI.

**Мета:** Зрозуміти призначення колекцій та навчитись їх використовувати. Зрозуміти призначення RTTI та навчитись його використовувати.

**Завдання.**

В даній роботі будуть використовуватись два класи, розроблені в попередніх лабораторних роботах.

Розробити програму, яка:

1. Створює колекцію об’єктів (JPanel та двох власних класів попередніх лабораторних робіт), причому об’єкти вносяться в колекцію у випадковій послідовності (використати генератор випадкових чисел).
2. Відображає колекцію на екрані з можливістю вибору поточного елементу.
3. Відображає тип поточного об’єкту, беручи інформацію з об’єкту типу Class для даного об’єкту.
4. Забезпечує можливість виконання самостійно розроблених методів поточного об’єкту.

**Індивідуальне завдання:**

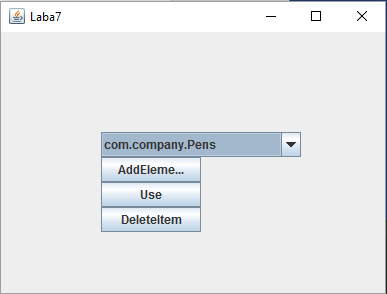
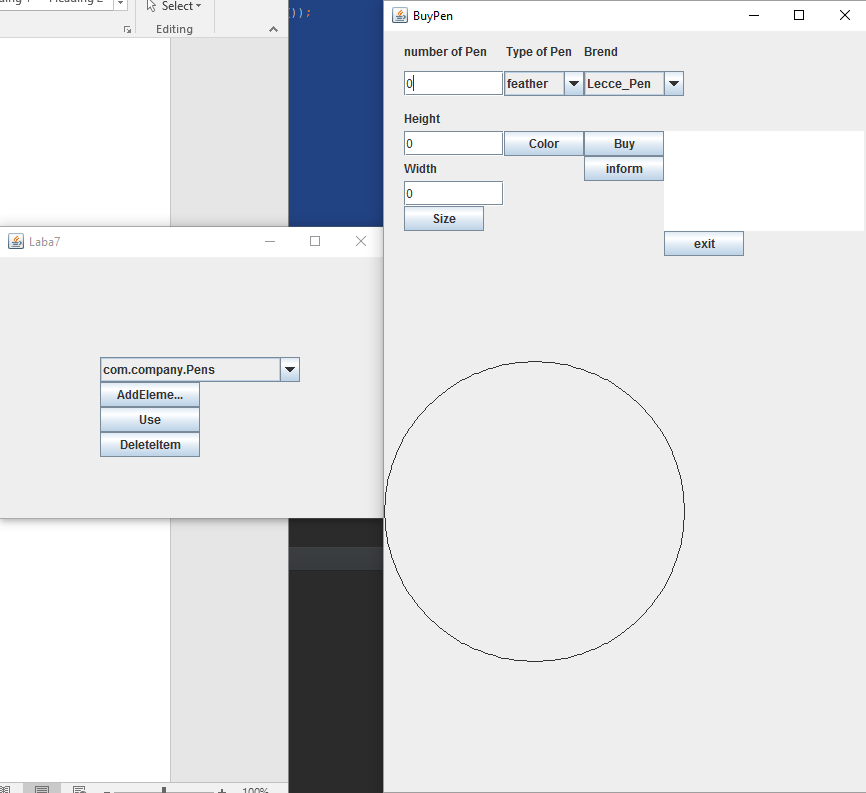
**Варіант 12 .**

-Пристрій для писання та ручка.

**Хід роботи :**

package com.company;  
import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.\*;  
import java.util.\*;  
import java.util.List;  
import java.lang.String;  
import java.util.Random;  
  
class SupJPanel extends JPanel  
{  
 void draw\_w()  
 {  
 JFrame windowTwo = new JFrame("close");  
 windowTwo.setSize(500, 500);  
  
 windowTwo.setLocationRelativeTo(null);  
 windowTwo.setLayout(null);  
 windowTwo.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
  
  
 //btExit  
 JButton btExit = new JButton("EXIT");  
 btExit.setBounds(200, 100, 80, 25);  
 windowTwo.add( btExit);  
 windowTwo.setVisible(true);  
  
 btExit.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 windowTwo.dispose();  
 //wind.setVisible(true);  
  
 }  
 });  
 }  
}  
class Pens extends SupJPanel {  
  
 int PensNumber;  
 Color color;  
 String Brend;  
 String TypeOfPen;  
 int numbOfPen;  
 int Width;  
 int Height;  
  
 Pens() {  
 numbOfPen = 0;  
 Brend = "Lecce\_Pen";  
 TypeOfPen = "feather";  
 Height = 300;  
 Width = 300;  
  
 }  
  
  
 public void id() {  
 System.*out*.println("Cat #" + PensNumber);  
 }  
  
 public Pens(int i) { PensNumber = i; }  
 public String toString() {  
 return " Pen : " + "\n" +  
 "Number of Pen = " + numbOfPen + "\n" + " Type = " + TypeOfPen + "\n" + "Brend = " + Brend + "\n";  
 }  
  
 void setPenWidth(int PenWingt) {  
 this.Width = PenWingt;  
 }  
  
 void setPenHeight(int PenHeight) {  
 this.Height = PenHeight;  
 }  
  
 void setColor(Color col) {  
 this.color = col;  
 }  
  
 void setNumbOfPen(int numb) {  
 this.numbOfPen = numb;  
 }  
  
 void setBrend(String bre) {  
 this.Brend = bre;  
 }  
  
 void setTypeOfPen(String type) {  
 this.TypeOfPen = type;  
 }  
  
 public void paint(Graphics g) {  
 int h = Height;  
 int w = Width;  
 g.setColor(color);  
 g.drawOval(0, 30, h, w);  
  
  
 }  
  
  
  
  
 void setAll(int Numb , String Bren , String Type)  
 {  
 numbOfPen = Numb ;  
 TypeOfPen = Type;  
 Brend = Bren;  
 }  
  
 String information()  
 {  
 String i = "Number of Pen = " + numbOfPen + "\n" + " Type = " + TypeOfPen + "\n" + "Brend = " + Brend + "\n";  
 return i ;  
 }  
  
 public void draw\_window(Pens pen ) {  
 String[] Brend = {  
 "Lecce\_Pen",  
 "Parker\_Pen",  
 "AURORA"  
 };  
  
  
 String[] Type = {  
 "feather",  
 "rollers",  
 "ball-point"  
 };  
  
  
 JFrame windowTwo = new JFrame("BuyPen");  
 windowTwo.setSize(500, 800);  
  
 windowTwo.setLocationRelativeTo(null);  
 windowTwo.setLayout(null);  
 windowTwo.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
  
  
 //btBuy  
 JButton btBuy = new JButton("Buy");  
 btBuy.setBounds(200, 100, 80, 25);  
 windowTwo.add(btBuy);  
  
 //inform  
 JButton btInf = new JButton("inform");  
 btInf.setBounds(200, 125, 80, 25);  
 windowTwo.add(btInf);  
  
 // btSize  
 JButton btSize = new JButton("Size");  
 btSize.setBounds(20, 175, 80, 25);  
 windowTwo.add(btSize);  
  
 //draw  
 pen.setBounds(0, 300, 800, 800);  
 windowTwo.add(pen);  
  
 //Height  
 JLabel HLabel = new JLabel("Height");  
 HLabel.setBounds(20, 75, 100, 25);  
 windowTwo.add(HLabel);  
  
 JTextField TxtHeight = new JTextField("0");  
 TxtHeight.setBounds(20, 100, 100, 25);  
 windowTwo.add(TxtHeight);  
 //Width  
 JLabel WLabel = new JLabel("Width");  
 WLabel.setBounds(20, 125, 100, 25);  
 windowTwo.add(WLabel);  
  
 JTextField TxtWidth = new JTextField("0");  
 TxtWidth.setBounds(20, 150, 100, 25);  
 windowTwo.add(TxtWidth);  
  
 //color  
  
 JButton btColor = new JButton("Color");  
 btColor.setBounds(120, 100, 80, 25);  
 windowTwo.add(btColor);  
  
  
 JLabel LPen = new JLabel("number of Pen");  
 LPen.setBounds(20, 5, 100, 30);  
 windowTwo.add(LPen);  
  
 JTextField numbPen = new JTextField("0");  
 numbPen.setBounds(20, 40, 100, 25);  
 windowTwo.add(numbPen);  
  
  
 //type  
 JLabel LType = new JLabel("Type of Pen");  
 LType.setBounds(122, 5, 80, 30);  
 windowTwo.add(LType);  
  
 JComboBox BoxTypeVar = new JComboBox(Type);  
 BoxTypeVar.setBounds(120, 40, 80, 25);  
 windowTwo.add(BoxTypeVar);  
  
 //brend  
 JLabel LBrend = new JLabel("Brend");  
 LBrend.setBounds(200, 5, 100, 30);  
 windowTwo.add(LBrend);  
  
 JComboBox BoxBreVar = new JComboBox(Brend);  
 BoxBreVar.setBounds(200, 40, 100, 25);  
 windowTwo.add(BoxBreVar);  
  
 //inform  
 JTextArea Information = new JTextArea();  
 Information.setBounds(280, 100, 200, 100);  
 windowTwo.add(Information);  
  
 //btExit  
 JButton btExit = new JButton("exit");  
 btExit.setBounds(280, 200, 80, 25);  
 windowTwo.add(btExit);  
  
  
 btExit.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 windowTwo.dispose();  
 //wind.setVisible(true);  
  
 }  
 });  
  
  
 btColor.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 Color choosenColor = JColorChooser.*showDialog*(null, "Choose color", Color.*white*);  
 if (choosenColor != null) {  
 pen.setColor(choosenColor);  
 pen.setOpaque(true);  
  
 windowTwo.repaint();  
  
 }  
 }  
 });  
  
 btSize.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
 pen.setPenHeight(Integer.*valueOf*(TxtHeight.getText().toString()));  
 pen.setPenWidth(Integer.*valueOf*(TxtWidth.getText().toString()));  
 windowTwo.repaint();  
 }  
 });  
  
  
 btBuy.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 // pen.setBrend(BoxBreVar.getSelectedItem().toString());  
 // pen.setTypeOfPen(BoxTypeVar.getSelectedItem().toString());  
 // pen.setNumbOfPen(Integer.valueOf(numbPen.getText().toString()));  
 // Information.setText(null);  
 // Information.append(pen + "");  
  
 pen.setAll(Integer.*valueOf*(numbPen.getText().toString()),BoxBreVar.getSelectedItem().toString(),BoxTypeVar.getSelectedItem().toString());  
 }  
 });  
  
 btInf.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 Information.setText(null);  
  
 Information.append( pen.information() + "");  
 }  
 });  
 windowTwo.setVisible(true);  
 }  
  
  
  
  
  
}  
  
  
class Felt\_tip\_pen extends Pens {  
 int Felt\_PensNumber;  
 String Packing\_type;  
 String Form;  
 String Type\_caps;  
  
 Felt\_tip\_pen() {  
 Packing\_type = "cardboard box";  
 Form = "round";  
 Type\_caps = "with a ventilated";  
 }  
  
  
 public void id() {  
 System.*out*.println("Cat #" + Felt\_PensNumber);  
 }  
  
 public Felt\_tip\_pen(int i) { Felt\_PensNumber = i; }  
 void setPacking\_type(String Packing) {  
 this.Packing\_type = Packing;  
 }  
  
 void setForm(String F) {  
 this.Form = F;  
 }  
  
 void setType\_caps(String Caps) {  
 this.Type\_caps = Caps;  
 }  
  
  
 public void paint(Graphics g) {  
 int w = Width;  
 int h = Height;  
 g.setColor(color);  
 g.drawRect(0, 30, w, h);  
  
  
 }  
  
  
  
 void setAll(int Numb , String form , String Pac\_t)  
 {  
 numbOfPen = Numb ;  
 Packing\_type = Pac\_t;  
 Form = form;  
 }  
  
 String information()  
 {  
 String i = " Felt\_Pen : " + "\n" +  
 "Number of Felt = " + numbOfPen + "\n" + " Type = " + TypeOfPen + "\n" + "Brend = " + Brend + "\n" + "Packing\_type = " + Packing\_type + "\n"  
 + "Form of Felt\_Pen = " + Form + "\n" + "Type caps = " + Type\_caps;  
 return i ;  
 }  
  
 public void draw\_Felt\_window(Felt\_tip\_pen felt ) {  
 String[] Brend = {  
 " KOH-I-NOOR",  
 "Faber Castell",  
 "Staedler",  
 "Lyra"  
 };  
  
  
 String[] Type = {  
 "watercolor",  
 "changeColor",  
 "withEffects"  
 };  
  
 String[] Pucking\_Type = {  
 "blister",  
 "carton",  
 "plastic packaging"  
 };  
  
 String[] Form\_Type = {  
 "round",  
 "hexagons",  
 "triquetrous"  
 };  
  
 String[] Type\_caps = {  
 "with ventilated ",  
 "non-ventilated cap"  
 };  
  
 JFrame windowTwo = new JFrame("BuyFeltPen");  
 windowTwo.setSize(500, 800);  
  
 windowTwo.setLocationRelativeTo(null);  
 windowTwo.setLayout(null);  
 windowTwo.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
  
 //inform  
 JButton btInf = new JButton("inform");  
 btInf.setBounds(200, 125, 80, 25);  
 windowTwo.add(btInf);  
  
 //btBuy  
 JButton btBuy = new JButton("Buy");  
 btBuy.setBounds(200, 100, 80, 25);  
 windowTwo.add(btBuy);  
  
 // btSize  
 JButton btSize = new JButton("Size");  
 btSize.setBounds(20, 175, 80, 25);  
 windowTwo.add(btSize);  
  
 //draw  
 felt.setBounds(0, 300, 800, 800);  
 windowTwo.add(felt);  
  
 //Height  
 JLabel HLabel = new JLabel("Height");  
 HLabel.setBounds(20, 75, 100, 25);  
 windowTwo.add(HLabel);  
  
 JTextField TxtHeight = new JTextField("0");  
 TxtHeight.setBounds(20, 100, 100, 25);  
 windowTwo.add(TxtHeight);  
 //Width  
 JLabel WLabel = new JLabel("Width");  
 WLabel.setBounds(20, 125, 100, 25);  
 windowTwo.add(WLabel);  
  
 JTextField TxtWidth = new JTextField("0");  
 TxtWidth.setBounds(20, 150, 100, 25);  
 windowTwo.add(TxtWidth);  
  
 //color  
  
 JButton btColor = new JButton("Color");  
 btColor.setBounds(120, 100, 80, 25);  
 windowTwo.add(btColor);  
  
  
 JLabel LPen = new JLabel("number of Felt\_Pen");  
 LPen.setBounds(20, 5, 100, 30);  
 windowTwo.add(LPen);  
  
 JTextField numbPen = new JTextField("0");  
 numbPen.setBounds(20, 40, 100, 25);  
 windowTwo.add(numbPen);  
  
  
 //type  
 JLabel LType = new JLabel("Type of Felt\_Pen");  
 LType.setBounds(122, 5, 80, 30);  
 windowTwo.add(LType);  
  
 JComboBox BoxTypeVar = new JComboBox(Type);  
 BoxTypeVar.setBounds(120, 40, 100, 25);  
 windowTwo.add(BoxTypeVar);  
  
 //brend  
 JLabel LBrend = new JLabel("Brend");  
 LBrend.setBounds(230, 5, 100, 30);  
 windowTwo.add(LBrend);  
  
 JComboBox BoxBreVar = new JComboBox(Brend);  
 BoxBreVar.setBounds(220, 40, 100, 25);  
 windowTwo.add(BoxBreVar);  
  
  
 //setPacking\_type  
 JComboBox BoxPackingVar = new JComboBox(Pucking\_Type);  
 BoxPackingVar.setBounds(320, 40, 70, 25);  
 windowTwo.add(BoxPackingVar);  
 //setForm  
 JComboBox BoxFormVar = new JComboBox(Form\_Type);  
 BoxFormVar.setBounds(390, 40, 70, 25);  
 windowTwo.add(BoxFormVar);  
 //SetCaps  
 JComboBox BoxCapsVar = new JComboBox(Type\_caps);  
 BoxCapsVar.setBounds(320, 65, 110, 25);  
 windowTwo.add(BoxCapsVar);  
  
  
 //inform  
 JTextArea Information = new JTextArea();  
 Information.setBounds(280, 100, 200, 120);  
 windowTwo.add(Information);  
  
 //btExit  
 JButton btExit = new JButton("exit");  
 btExit.setBounds(280, 220, 80, 25);  
 windowTwo.add(btExit);  
  
  
 btExit.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 windowTwo.dispose();  
 // wind.setVisible(true);  
  
 }  
 });  
  
  
 btColor.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 Color choosenColor = JColorChooser.*showDialog*(null, "Choose color", Color.*white*);  
 if (choosenColor != null) {  
 felt.setColor(choosenColor);  
 felt.setOpaque(true);  
  
 windowTwo.repaint();  
  
 }  
 }  
 });  
  
 btSize.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
 felt.setPenHeight(Integer.*valueOf*(TxtHeight.getText().toString()));  
 felt.setPenWidth(Integer.*valueOf*(TxtWidth.getText().toString()));  
 windowTwo.repaint();  
 }  
 });  
  
  
 btBuy.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
 felt.setAll(Integer.*valueOf*(numbPen.getText().toString()),BoxFormVar.getSelectedItem().toString(),BoxPackingVar.getSelectedItem().toString());  
  
 felt.setBrend(BoxBreVar.getSelectedItem().toString());  
 felt.setTypeOfPen(BoxTypeVar.getSelectedItem().toString());  
  
  
 felt.setType\_caps(BoxCapsVar.getSelectedItem().toString());  
 // felt.setForm(BoxFormVar.getSelectedItem().toString());  
 // felt.setNumbOfPen(Integer.valueOf(numbPen.getText().toString()));  
  
 }  
 });  
  
 btInf.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 Information.setText(null);  
 Information.append( felt.information() + "");  
 }  
 });  
 windowTwo.setVisible(true);  
 }  
  
  
  
  
  
}  
  
  
  
  
  
  
  
  
  
public class Main {  
  
 public static void main(String[] args) {  
 JFrame window = new JFrame("Laba7");  
 window.setSize(400, 300);  
 window.setLocationRelativeTo(null);  
  
 window.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
  
  
 window.setLayout(null);  
  
 JButton btAddElement = new JButton("AddElement");  
 btAddElement.setBounds(100, 125, 100, 25);  
 window.add(btAddElement);  
  
  
 JComboBox ComboBoxV= new JComboBox();  
 ComboBoxV.setBounds(100, 100, 200, 25);  
 window.add(ComboBoxV);  
  
  
  
 JButton btDelItem = new JButton("DeleteItem");  
 btDelItem.setBounds(100, 175, 100, 25);  
 window.add(btDelItem);  
  
 JButton btUseItem = new JButton("Use");  
 btUseItem.setBounds(100, 150, 100, 25);  
 window.add(btUseItem);  
  
  
  
  
  
 window.setVisible(true);  
  
  
  
  
  
  
 List feltAndPen = new ArrayList();  
 int mas[] = new int[3];  
 for (int i = 0; i < 3; i++)  
 {  
 mas[i] =i;  
  
 }  
  
  
  
 btAddElement.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
  
  
 for (int i:mas )  
 {  
 System.*out*.println(mas[i]);  
 if(i == 0)  
 {  
 feltAndPen.add(new Pens());  
  
 }  
  
 else if(i == 1)  
 {  
 feltAndPen.add(new Felt\_tip\_pen());  
 }  
  
 else if(i == 2)  
 {  
 feltAndPen.add(new SupJPanel());  
 }  
  
 ComboBoxV.addItem(feltAndPen.get(i).getClass().getName());  
  
 }  
  
  
  
  
  
  
  
 }  
 });  
  
  
  
 btUseItem.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
  
 switch (ComboBoxV.getSelectedItem().toString()) {  
 case "com.company.Pens" :  
 {  
 feltAndPen.remove(ComboBoxV.getSelectedIndex());  
 Pens p = new Pens();  
 p.draw\_window(p);  
 feltAndPen.add(p);  
 }break;  
 case "com.company.Felt\_tip\_pen":  
 {  
 feltAndPen.remove(ComboBoxV.getSelectedIndex());  
 Felt\_tip\_pen f = new Felt\_tip\_pen();  
 f.draw\_Felt\_window(f);  
 feltAndPen.add(f);  
 }break;  
 case "com.company.SupJPanel":  
 {  
 feltAndPen.remove(ComboBoxV.getSelectedIndex());  
 SupJPanel Jpanel = new SupJPanel();  
 Jpanel.draw\_w();  
 feltAndPen.add(Jpanel);  
 }break;  
 }  
  
  
  
 }  
 });  
  
 btDelItem.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 ComboBoxV.removeItem(ComboBoxV.getSelectedItem());  
 feltAndPen.remove(ComboBoxV.getSelectedIndex());  
 }  
 });  
  
  
  
  
  
 }  
}

**Результат виконання :**



**Висновок** : під час виконання лабораторної роботи , я зрозумів призначення колекцій та навчився їх використовувати. Зрозумів призначення RTTI та навчився його використовувати.