CSGE601021 Foundations of Programming 2
(Dasar-Dasar Pemrograman 2)
Lab 07: GUI with Events and Listeners; Objects as parameters

Build a *Java project* for each part of this lab.

Task Description:

A) Voting with Buttons

Files *VoteCounter.java* and *VoteCounterPanel.java* contain the program that counts the number of times a button is pushed. Each push is a vote for Joe so the button and variables have been named appropriately.

1. Compile the program, then run it to see how it works.

2. Modify the program so that there are three candidates to vote for—Joe, Sam and Mary. To do this you need to do the following:
   a. Add variables for Sam and Mary respectively—a vote counter, a button, and a label.
   b. Add a new inner class named *SamButtonListener* to listen for clicks on the button for Sam. Instantiate an instance of the class when adding the ActionListener to the button for Sam. Do likewise for Mary.
   c. Add the buttons and labels for Sam and Mary to the panel.

3. Compile and run your program.

```
//********************************************************
// VoteCounter.java
//
// Demonstrates a graphical user interface and event listeners to
// tally votes
//********************************************************
import javax.swing.JFrame;

public class VoteCounter
{
    //-----------------------------------------
    // Creates the main program frame.
    //-----------------------------------------
    public static void main(String[] args)
    {
       JFrame frame = new JFrame("Vote Counter");
       frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

       frame.getContentPane().add(new VoteCounterPanel());

       frame.pack();
       frame.setVisible(true);
    }
}
```

```
//********************************************************
// VoteCounterPanel.java
//
// Demonstrates a graphical user interface and event listeners to
// tally votes
//********************************************************

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class VoteCounterPanel extends JPanel
{
    private int votesForJoe;
    private JButton joe;
    private JLabel labelJoe;

    //-----------------------------------------------
    // Constructor: Sets up the GUI.
    //-----------------------------------------------
    public VoteCounterPanel()
    {
       votesForJoe = 0;

       joe = new JButton("Vote for Joe");
       joe.addActionListener(new JoeButtonListener());

       labelJoe = new JLabel("Votes for Joe: " + votesForJoe);

       add(joe);
       add(labelJoe);

       setPreferredSize(new Dimension(400, 40));
       setBackground(Color.cyan);
    }

    //*************************************************
    // Represents a listener for button push (action) events
    //*************************************************
    private class JoeButtonListener implements ActionListener
    {
       //-----------------------------------------------
       // Updates the counter and label when Vote for Joe
       // button is pushed
       //-----------------------------------------------
       public void actionPerformed(ActionEvent event)
       {
           votesForJoe++;
           labelJoe.setText("Votes for Joe: " + votesForJoe);
       }
    }
}
```

B) Objects as parameters

The file *ChangingPeople.java* contains a program that illustrates parameter passing. The program uses *Person* objects defined in the file *Person.java*. Do the following:

1. On a piece of paper, trace the execution of the program using diagrams and show what is printed by the program.
2. Compile and run the program to see if your trace was correct.
3. Modify the *changePeople* method minimally so that it does what the documentation says it does, that is, the two Person objects passed in as actual parameters are actually changed.

```
// ********************************************************************
//   ChangingPeople.java
//
//   Demonstrates parameter passing -- contains a method that should
//   change to Person objects.
// ********************************************************************

public class ChangingPeople
{
    // -----------------------------------------------------------
    //  Sets up two person objects, one integer, and one String
    //  object.  These are sent to a method that should make
    //  some changes.
    // -----------------------------------------------------------
    public static void main (String[] args)
    {
      Person person1 = new Person ("Sally", 13);
      Person person2 = new Person ("Sam", 15);

      int age = 21;
      String name = "Jill";

      System.out.println ("\nParameter Passing... Original values...");
      System.out.println ("person1: " + person1);
      System.out.println ("person2: " + person2);
      System.out.println ("age: " + age + "\tname: " + name + "\n");

      ChangingPeople ob = new ChangingPeople();
      ob.changePeople (person1, person2, age, name);

      System.out.println ("\nValues after calling changePeople...");
      System.out.println ("person1: " + person1);
      System.out.println ("person2: " + person2);
      System.out.println ("age: " + age + "\tname: " + name + "\n");
    }
```

```
    // -----------------------------------------------------------------
    //  Change the first actual parameter to "Jack - Age 101" and change
    //  the second actual parameter to be a person with the age and
    //  name given in the third and fourth parameters.
    // -----------------------------------------------------------------
    public void changePeople (Person p1, Person p2, int age,
                                    String name)
    {
      System.out.println
         ("\nInside changePeople... Original parameters...");
      System.out.println ("person1: " + p1);
      System.out.println ("person2: " + p2);
      System.out.println ("age: " + age + "\tname: " + name + "\n");

      // Make changes
      Person p3 = new Person (name, age);
      p2 = p3;
      name = "Jack";
      age = 101;
      p1.changeName (name);
      p1.changeAge (age);

      // Print changes
      System.out.println ("\nInside changePeople... Changed values...");
      System.out.println ("person1: " + p1);
      System.out.println ("person2: " + p2);
      System.out.println ("age: " + age + "\tname: " + name + "\n");
    }
}
```

```
// ********************************************************************
//   Person.java
//
//   A simple class representing a person.
// ********************************************************************
public class Person
{
    private String name;
    private int age;

    // -----------------------------------------------------------------
    //  Sets up a Person object with the given name and age.
    // -----------------------------------------------------------------
    public Person (String name, int age)
    {
      this.name = name;
      this.age = age;
    }

    // -----------------------------------------------------------------
    //   Changes the name of the Person to the parameter newName.
    // -----------------------------------------------------------------
    public void changeName(String newName)
    {
      name = newName;
    }

    // -----------------------------------------------------------------
    //   Changes the age of the Person to the parameter newAge.
    // -----------------------------------------------------------------
    public void changeAge (int newAge)
    {
      age = newAge;
    }

    // -----------------------------------------------------------------
    //   Returns the person's name and age as a string.
    // -----------------------------------------------------------------
    public String toString()
    {
      return name + " - Age " + age;
    }
}
```

Marking components:

Code correctness	90%
Clear comments	10%

Through the link at SCeLE, submit all your project files (2 project folders), zipped into a file: lab07_<class>_<TACode>_<YourName>_<YourNPM>.zip

Selamat Mengerjakan!
'Met Ngoding! ☺
L.Y.Stefanus & the Asdos Team

☺Happy Programming☺