Labs

Lab 7.1: Actions Class

What is the purpose?

In this lab, you will write a program that contains the menu items and toolbar buttons that can be used to select flags to be displayed in an ImageViewer. Use the Action interface to centralize the processing for the action.

Note: The flag images can be found in the image folder in the *Programming in Java II Student CD*.

What are the steps?

- **Task 1**

  Procedure:

  1. Create a Java GUI file called Actions.java that contains the menu items and toolbar buttons to display the selected images in ImageViewer.
  2. Complete the following code or create your own code:

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

public class Actions extends JApplet {
  // Create image icons
  private ImageIcon caImageIcon =
    new ImageIcon(getClass().getResource("image/caIcon.gif"));
  private ImageIcon usImageIcon =
    new ImageIcon(getClass().getResource("image/usIcon.gif"));

  private JLabel jlblImage = new JLabel(caImageIcon,
JLabel.CENTER);
  private FlowLayout flowLayout = new FlowLayout();

  public Actions() {
    // Create actions


    // Create menus


    // Add actions to the menu


    // Add actions to the toolbar


    // Add tool bar to the east and panel to the center

  }
```

```
  private class MyAction extends AbstractAction {
    String name;

    MyAction(String name, Icon icon) {
      super(name, icon);
      putValue(Action.SHORT_DESCRIPTION, " Select the " + name +
        " flag to display");
      this.name = name;
    }

    public void actionPerformed(ActionEvent e) {
      if (name.equals("CA"))
        jlblImage.setIcon(caImageIcon);
      else if (name.equals("US"))
        jlblImage.setIcon(usImageIcon);
    }
  }

  public static void main(String[] args) {
    Actions applet = new Actions();
    JFrame frame = new JFrame();
    //EXIT_ON_CLOSE == 3
    frame.setDefaultCloseOperation(3);
    frame.setTitle("Using Actions");
    frame.getContentPane().add(applet, BorderLayout.CENTER);
    applet.init();
    applet.start();
    frame.setSize(400,320);
    Dimension d = Toolkit.getDefaultToolkit().getScreenSize();
    frame.setLocation((d.width - frame.getSize().width) / 2,
      (d.height - frame.getSize().height) / 2);
    frame.setVisible(true);
  }
}
```

Figure 7-1-1

Here is a sample of the output:

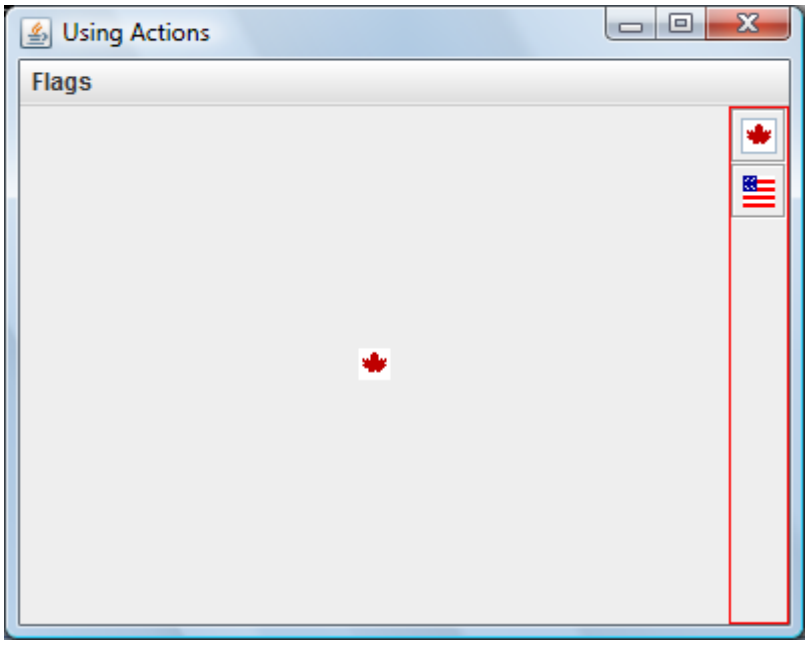

Figure 7-1-2: Sample Output 1

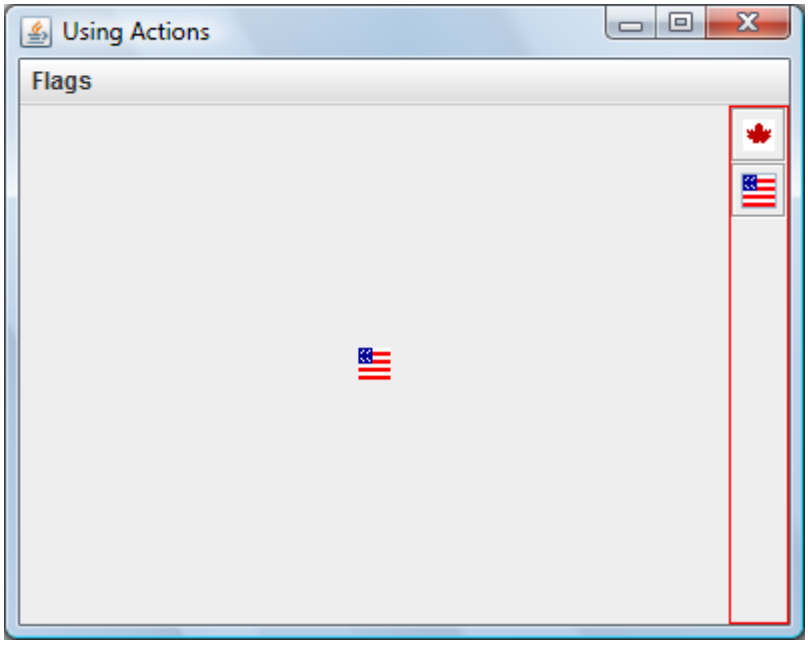

Figure 7-1-3: Sample Output 2

Did it work?

Were you able to select a different flag:

- With the menu items?
- With the toolbar buttons?

Lab 7.2: JFileChooser Class

What is the purpose?

In this lab, you will write a program that enables the user to select a file from a file-open dialog box. This dialog box will be displayed when the user clicks the Browse button. The file is displayed in the text area. The file name is displayed in the text field when a user clicks the OK button in the file-open dialog box. The user can also enter the file name in the text field and press the Enter key to display the file in the text area.

What are the steps?

- **Task 1**

  Procedure:

  1. Create a Java GUI file called displayFiles.java that enables the user to view a file by selecting it from a file-open dialog box.
  2. Complete the following code or create your own code:

```
// Display file in a text area. File is chosen from a dialog box
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import javax.swing.*;

public class displayFiles extends JFrame implements ActionListener {
  private Jbutton jbtBrowse = new Jbutton("Browse");

  // Text field to receive file name
  private JtextField jtfFile = new JtextField();

  // Text area to display file
  private JtextArea jtaFileContent = new JtextArea();

  // Create jFileChooser
  private JfileChooser jFileChooser = new JfileChooser();

  public static void main(String[] args) {
    displayFiles frame = new displayFiles();
    frame.setSize(400, 300);
    frame.setTitle("Using JfileChooser");
    frame.setDefaultCloseOperation(Jframe.EXIT_ON_CLOSE);
    frame.setVisible(true);
  }

  public displayFiles() {
    // Create a Panel to hold a label, a text field, and a button
    Jpanel p = new Jpanel();
    p.setLayout(new BorderLayout());
    p.add(new Jlabel("Filename"), BorderLayout.WEST);
    p.add(jtfFile, BorderLayout.CENTER);
    jtfFile.setBackground(Color.white);
    jtfFile.setForeground(Color.black);
    p.add(jbtBrowse, BorderLayout.EAST);
```

```
    // Create a 120oolean120le text area
    JscrollPane jsp = new JscrollPane(jtaFileContent);

    // Set default directory to the current directory


    // Use BorderLayout for the frame
    getContentPane().setLayout(new BorderLayout());
    getContentPane().add(jsp, BorderLayout.CENTER);
    getContentPane().add(p, BorderLayout.SOUTH);
    jtaFileContent.setBackground(Color.white);
    jtaFileContent.setForeground(Color.black);

    // Register listener

  }

  //handling the "View" button
  public void actionPerformed(ActionEvent e) {
    if (e.getSource() == jbtBrowse) {
      browse();
    }
    else if (e.getSource() == jtfFile) {
      showFile(new File(jtfFile.getText().trim()));
    }
  }

  private void browse() {


  }

  private void showFile(File file) {
    BufferedReader infile = null;  //declare buffered stream
    //get file name from the text field
    String 120oolea;

    jtfFile.setText(file.getName());

    try {
      //create a buffered stream

      //read a line


      120oolean firstLine = true;

      //append the line to the text area

    }
    catch (FileNotFoundException ex) {
      System.out.println("File not found: " + file.getName());
    }
    catch (IOException ex) {
      System.out.println(ex.getMessage());
    }
    finally {
```

```
      try {

      }
      catch (IOException ex) {
      }
    }
  }
}
```

Figure 7-2-1

Here is a sample of the output:

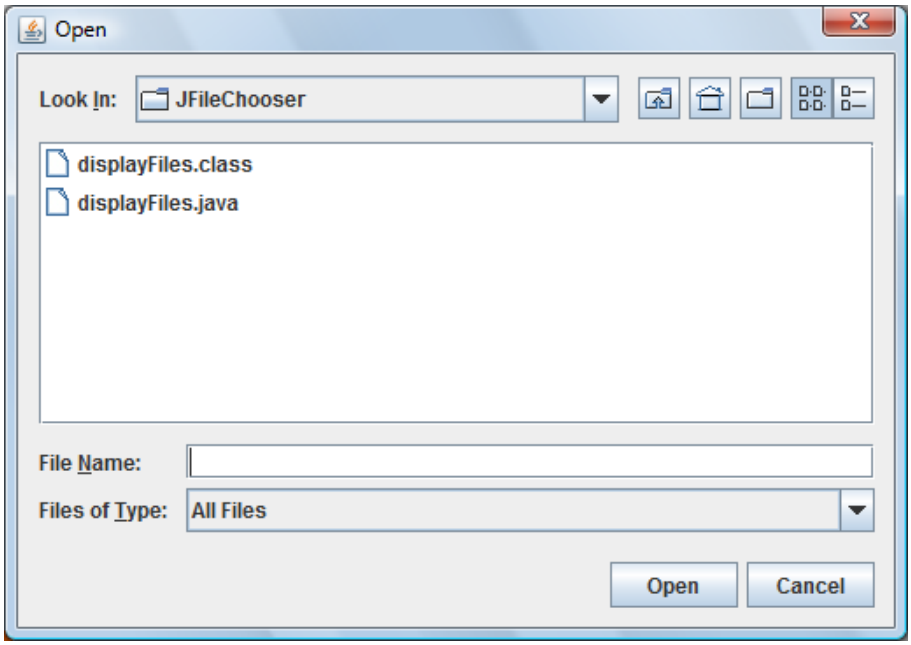

Figure 7-2-2

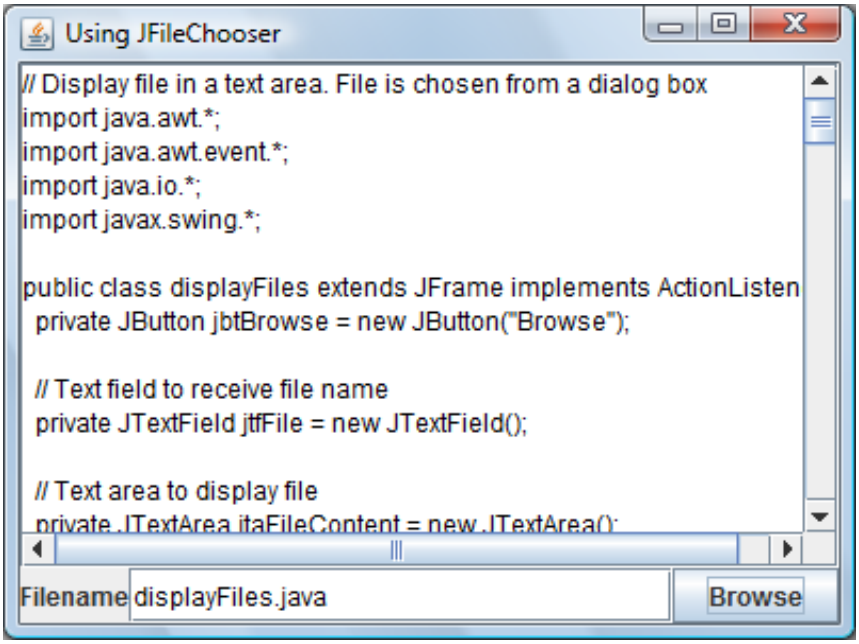

Figure 7-2-3

3. Submit your Java code and sample output to your instructor.

Did it work?

Were you able to use the GUI to:

- Select a file from a file-open dialog box?
- Display the contents of a file in a text area?