

CSCI 6901: Foundations of Programming

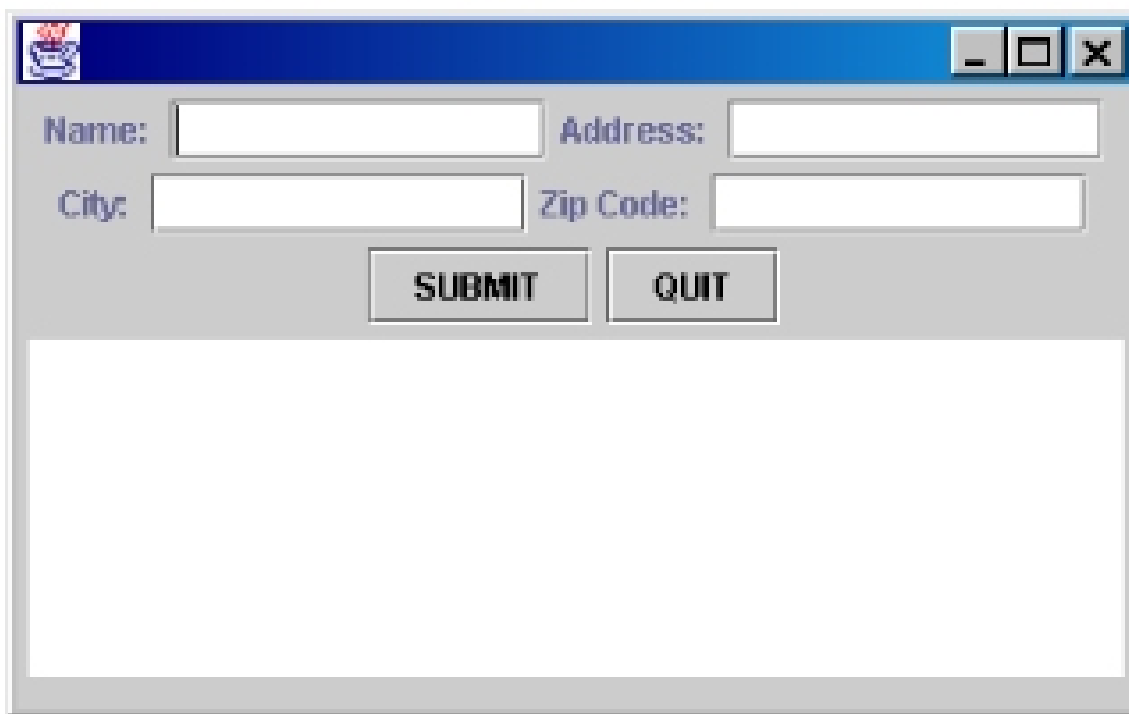
Assignment 5: Vectors and Visual Applications

Due Monday, November 21

Programming Project 1, Chapter 7 (page 514).

Some more specific instructions and hints:

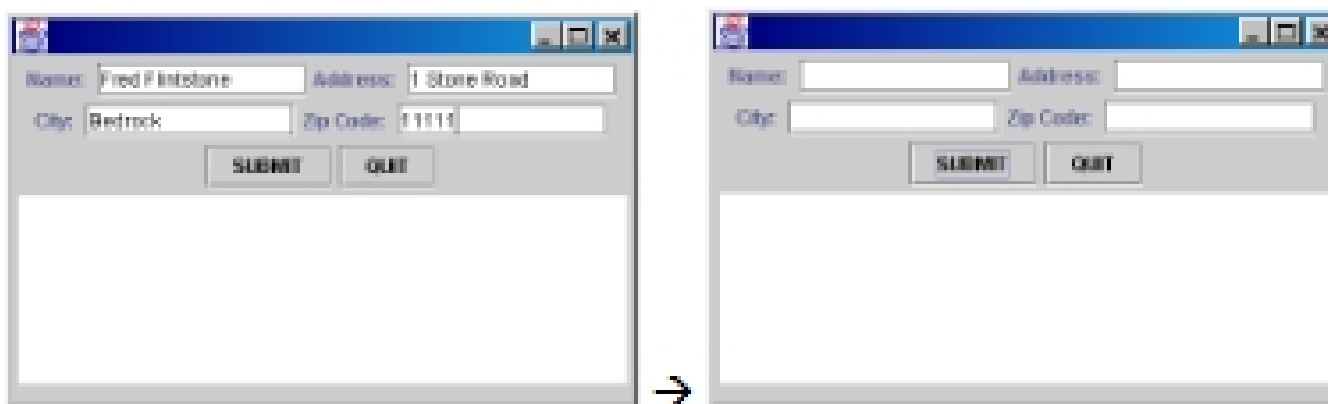
- You are to create an application (that extends `JFrame`) rather than an applet. It should look roughly like the one below:



A screenshot of a Java Swing window. The window has a blue title bar with a small icon on the left and standard window controls (minimize, maximize, close) on the right. The main content area is light gray and contains four text input fields arranged in a 2x2 grid. The top-left field is labeled "Name:", the top-right "Address:", the bottom-left "City:", and the bottom-right "Zip Code:". Below the input fields are two buttons: "SUBMIT" and "QUIT". The bottom half of the window is a large, empty white rectangular area.

Do not worry too much about coming up with a great layout for this – we will improve the appearance of the application in the next assignment.

- When the user types in the information and presses `SUBMIT`, the information should no longer be displayed in the text fields (it should be added to the `Vector`).



Two side-by-side screenshots of the application window, connected by a right-pointing arrow. The left screenshot shows the window with the text fields filled with the following data: Name: Fred Flintstone, Address: 1 Stone Road, City: Bedrock, Zip Code: 51111. The right screenshot shows the same window after the SUBMIT button has been clicked; the text fields are now empty, and the information has been removed from the display.

- When the QUIT button is pressed, the address information for each person is to be displayed on a separate line in the text area, with each part separated by a tab.

The screenshot shows a Java Swing window with a blue title bar containing the text 'JW'. The window has a standard Mac OS-style title bar with minimize, maximize, and close buttons. The main content area is a light gray panel. At the top, there are four text input fields arranged in two rows. The first row has 'Name:' followed by an empty text box, and 'Address:' followed by an empty text box. The second row has 'City:' followed by an empty text box, and 'Zip Code:' followed by an empty text box. Below these fields are two buttons: 'SUBMIT' and 'QUIT'. At the bottom of the window is a text area containing two lines of address information: 'Fred Flintstone 1 Stone Road Bedrock 11111' and 'George Jetson 10 Skyway Ln Future City 99999'.

Note that after the QUIT button is pressed, both buttons are **disabled**. You will need to research the **AbstractButton** class to find out how to do this.

- You are to create a very simple **support class** called **Address**, which stores the name, address, city, and zip of each person in your address book. It will need, at a minimum, the following:
 - A **constructor**, which takes a name, address, city, and zip (all Strings) as parameters, and stores them as state variables.
 - `getName()`, `getAddress()`, `getCity()`, and `getZip()` methods which return that information.
- Your application will contain a single **Vector** object containing each **Address** object.
 - When a new address is submitted, an **Address** object is constructed and added to the **Vector**.
 - When the QUIT button is pressed, your application will need to iterate through the **Vector**, extracting each **Address** object in turn. Don't forget that you will need to cast the object back to the **Address** type! Your application will then need to use the `getName()`, `getAddress()`, `getCity()`, and `getZip()` methods to extract each string, and append that information to the text area.