

CS108 Assignment 4: Single-Window GUI

Due Date: 31 March 2005 @ Midnight

Learning Objective

Learn how to build a simple graphical user interface for an application. The GUI for this application will be entirely contained in a single window. In addition, you will practice the design pattern known as “model-view-controller”, by developing a user interface (the GUI), which is separate from the implementation (the application class that does the “real work”). The back-end implementation code will be provided to you.

Problem Statement and Requirements

Most application developed for the Windows and Macintosh operating systems have graphical user interfaces. In this assignment, you will create a simple GUI interface for an application. The GUI you will develop should be built using the NetBeans graphical-user-interface drawing palette, and will take the structure of a single window, using labels, buttons, and text boxes.

Your application will provide a user interface, which will interact with one of the following back-end applications (your choice) that are available in the newest *cs108.jar* file*:

- Retirement Calculator (similar to assignment 2)
- Loan Payment Calculator
- Temperature Converter (Celsius/Fahrenheit/Kelvin)
- Contact Manager (similar to assignment 3)

Your GUI should:

- Create a window at start up.
- Display the application name in the title bar.
- Provided intuitively named fields and buttons.
- Take input from the user as required by the application you choose.
- Interact with the existing application class (calling methods to perform the “real work”).
- Present results to the user in the GUI window.

Alternative Project Option

You are encouraged to come up with another application idea of your own to fulfill the requirements of this project (above). If you choose this, you should be prepared to write the implementation code yourself, as well as the GUI. If you choose to do so, you must send me your idea by email no later than Friday, March 24, and obtain approval.

If you choose to do this, and you do a good and clever job, some bonus points will be awarded. However, if you choose this and cannot complete the assignment, you do so at your own risk.

*Get the new *cs108.jar* file from <http://people.bu.edu/azs/academics/cs108/cs108.jar>, and replace any previous version on your computer with this one. The documentation for the classes is available at <http://people.bu.edu/azs/academics/cs108/javadoc/>.

Programming hints

- Getting Started: after you choose which application you want to do, start by drawing the GUI form – don't worry about code yet, just work on the interface (look/feel).
- Use the properties dialog to give each GUI widget a meaningful name (e.g. call a button by a name reflecting it's purpose, "exitButton" to exit the application, not "jButton1").
- Try building and running the application. It won't do anything yet, but it should look right. If not, adjust this.
- Now add your application's implementation class (e.g., FinancialCalculator, LoanPaymentCalculator, etc.). You will create a member variable in your GUI class for the implementation class, and initialize it as needed (constructor or first use).
- NetBeans will do most of the "code generation" to enable the events. You will just need to do a bit of code to interact with your application class. Implement the methods needed to collect data from the form and call the methods on your implementation class, as well as displaying the result.

Deliverables

- The java code file(s) (*.java files).
- The NetBeans GUIs (*.form files).
- A short executive summary, which will serve to introduce the application.

Grading Criteria

- Correct syntax, no compile errors, good code formatting, follow naming conventions, appropriate comments in the code
- GUI window is aesthetically pleasing: items lined up, font appropriate and readable, etc.
- Interaction with the back-end application works correctly: passing arguments to methods, retrieving return values, displaying results in the GUI window.
- Executive summary document: briefly explain the motivation for creating such an application, what the application does, how it works, why you used the programming elements you did for this type of project, as well as any lessons learned or recommendations for a revised implementation of this application.

Questions?

Bring them to class, or email me, azs@bu.edu.

NOTE: I will take all questions up until 6 hours before the assignment deadline. After that time, you're on your own. Plan your time wisely and start early.