CSC 122: Calculator

Basic Graphical User Interface

1 Introduction

In this lab, you will be required to take a pre-existing GUI framework for a calculator and add the proper fuctionality to it, such that it behaves as a calculator should. At this point, you should have all the necessary knowledge to implement this calculator.

The calculator must work like a normal calculator, in that it displays the digits and operations that have been entered. It must also correctly perform the mathematical operations specified in the Activity section of this document. A good example of this can be seen in the existing Android calculator.

2 Objective

In this lab, you will learn how to utilize and implement a Graphical User Interface or GUI. Most all applications use a GUI because they are more intuitive and sometimes easier to use than a command line interface. You will be supplied with a non operational GUI application to which you must add the necessary functionality to each button. You are highly encouraged, and will possibly be graded, to enhace the visual aesthetics of this application to make it more user friendly and appealing. As is, it uses a basic design layout and does not rotate to landscape mode. Any modifications to the GUI must be done via the text version and not the drag and drop option. By the end of this lab, you will be able to implement your own GUI application without any assistance.

3 Activity

These functions must be implemented manually:

1. Numbers 0 through 9
2. Addition
3. Subtraction
4. Multiplication
5. Division

6. Equals/Evaluate
7. Negative
8. Decimal
9. Backspace (delete the last added digit)
10. Clear All

These functions should be done with the use of a Java or Android specific API:

1. Square Root
2. Inverse
3. Percent

To make the buttons of the calculator function you must implement an anonymous inner class that listens for button presses. Anonymous inner classes are classes that are declared and instantiated in the middle of an expression. Because they are only called when triggered by a certain event, ex. the number 8 key is pressed, they do not need a named reference. By doing this the programmer can link specific code to specific GUI elements, such that each GUI element has its own function. More information about anonymous inner classes can be readily found on the web and Oracle's Java documentation.

4 Conclusion

In essence, you will be learning how to modify and improve and existing GUI application. This is meant as an introduction to GUIs to get you familiar with how GUIs are generated in the Android OS. This will reinforce design and attention to detail while making an application that is intended to be easy to use for a general, non-technology inclined audience.

5 Deliverables

You are required to turn in your entire project source code as a zip file through the means specified by the teacher. The project must successfully compile into an Android application. If the application does not compile, your project will not be graded. If the application does not run on a phone, it will receive minimal marks. Any function that does not work will result in a loss of points. A force close (crash) will also result in a deduction of points. Any known bugs should be documented in a readme file included with the project. This may allow for a smaller deduction of points depending on how serious the bugs are.