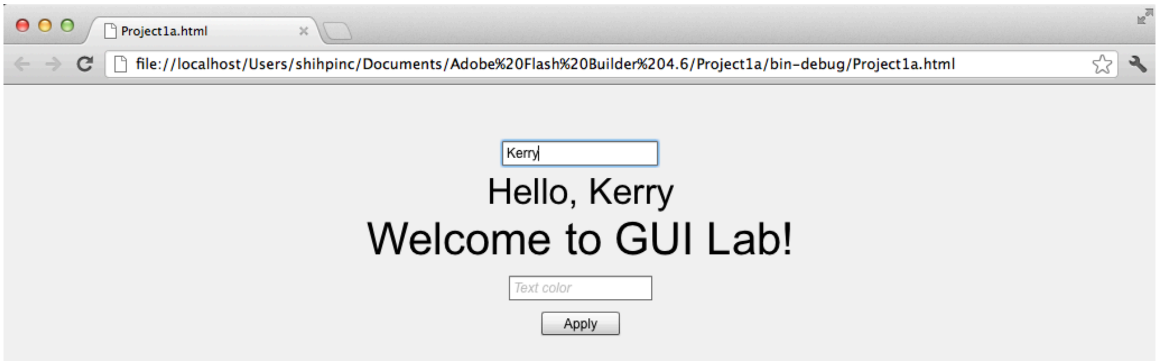
05433-B/05633-B User Interface Lab (Section B - GUI)

Project 1A: Hello {name}

due 9:00am Wednesday, September 11, 2013.

Goal

The goal of part A is to get Flash Builder up and running and to build a basic application that will help you get familiarized with MXML and Flash Builder's interface builder.



Project1a.html

file://localhost/Users/shihpinc/Documents/Adobe%20Flash%20Builder%204.6/Project1a/bin-debug/Project1a.html

Kerry

Hello, Kerry

Welcome to GUI Lab!

Your task is to replicate the interface shown above. When the user first starts the program, both text inputs are empty and there is gray text in them to suggest what the user should enter – "Your name" in the first text input and "Text color" in the second text input. Whenever the user types something into the first text input (in this case, the value is Kerry), the value in the "Hello, {name}" label changes as the user types. The user can enter 6-digit hexadecimal colors (starting with 0x, for example, 0x0000ff means blue) in the second text input, and press the "Apply" button to change the color of the greeting text.

For every project, you will be required to leave your name in a comment. To create a comment in your Hello {name} file, switch over to the source view and below the first line (the one that reads `<?xml version="1.0" encoding="utf-8"?>`), type in: `<!-- (your name) -->`

Layout

In your MXML file, there should be two text fields, two text inputs and one text button.

There should be five graphical elements specified:

1. A **text input** that shows gray hint text "Your name" when it is empty.
2. A **label** with font size 32pt, whose value is "Hello, {name}" Where {name} is the value of the text input above
3. A **label** with font size 40pt, with a secondary greeting ("Welcome to GUI Lab!" in the example above)
4. A **text input** that shows gray hint text "Text color" when it is empty.
5. A **button** that has text "Apply" on it. When it is clicked, the color of the two labels becomes the value of the text input above.

All of these objects should be stacked vertically and center-aligned. Note that the third graphical object (the label saying “Hello, {name}” should be centered **regardless of how long {name} is**. There should be a 10px gap between each object, and a 50px gap between the top of the browser window and the first text input. Finally, the background color should be gray (0xcccccc).

Hints

- Use the setStyle function to set the color of the text.
    - For example, text1.setStyle(“color”, “0x000000”) sets the color of the element that has id “`text1`” to black.

Tentative Rubric

Turned in & compiles:	5 points
All elements (labels, text inputs and button) are present:	5 points
Both text inputs have the correct hint text:	5 points
The first label updates with the first text input:	5 points
The text color updates correctly:	5 points
Items are always centered:	5 points
Gaps between the elements and the browser are set up correctly:	5 points
Background color is gray (0xcccccc):	5 points
Your name is in a comment:	5 points

Critical Thinking

You are not required to turn in answers to any of the questions in this section, but we recommend that you explore and think about some of the questions.

1. Our application only stays centered within the screen until we resize it to be smaller than 955x600. Why? How can we prevent this?
2. Part A consisted of only declarative programming. How would we implement this in an imperative way?
3. The second text input only recognizes color in the 6-digit hexadecimal format. If the user enters the color in other formats (for example, (0, 0, 256)), the color would not update correctly. How could we design the UI to prevent the user accidentally enter something that does not match what the system expects?

Turning it in

Project 1a is due by 9:00am September 11th, 2013 as an FXP file (Instructions for how to create an FXP file are below). Email your file to Kelly at krivers@andrew.cmu.edu. Late entries will be penalized -5% for every late day.