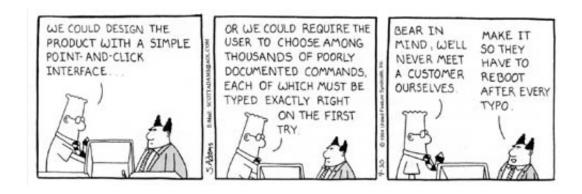# CS150 - Lab Prep 8
Design
Due: Friday Nov. 1, at the beginning of class

Feedback *(optional)*

We're a bit over halfway through the course and I wanted to see how everything is going for you guys. I've setup a quick, anonymous survey online that you can take at:

`https://docs.google.com/spreadsheet/viewform?formkey=dGFKMWxtelY5X3ZueE45TDlpbkhnQlE6MA`

It is optional, however, I would appreciate the feedback and it shouldn't take more than a minute or two.

I'll go over a summary of the results with you guys in class next week.

matplotlib

We will be using `matplotlib` for the lab this Friday. If you plan on working on your own computer, you'll need to install `matplotlib` before lab. In the "Resources" section of the course web page, I've put information on how to install `matplotlib`. Follow these instructions and then try out some of the simple examples from class on Wednesday to make sure it is installed properly.

If you have troubles with installing `matplotlib` please come see me.

Program design

I've posted the lab for this Friday on the course web page. Read through the lab and make sure you understand what the program is supposed to be doing.

Once you have a good feeling for the program, your lab prep for this week is to write a high-level design for your program. Your design *must* include the following information:

- the list of all of the functions you plan on implementing. For each function, you should include:

  - the list of what parameters it will take
  - what it will return
  - one sentence describing what the function will do (eventually something like this will end up in the docstring)

- in addition, include description of any other aspects of your program (e.g. functionality not in functions)

As you're putting together your design, think about the following:

- What is the flow of the program and the flow of information? What needs to happen first? Second?

- What information will each function generate and how will that information be used by other functions?

- Keep functions as simple as possible and as general as possible.

- If the function does some computation, don't also have it print out that information. Instead return it and print it out somewhere else. This allows you to reuse that function in other situations where printing isn't desired.

You will be graded based on the detail that you provide in addition to how well you've thought through the problem. The more time you spend thinking about the organization of your code, the easier actually generating the final code will be.