CSCI 3353 Object Oriented Design
Homework Assignment 9, part 1
Due Monday, November 28, 5pm
NO LATE SUBMISSIONS

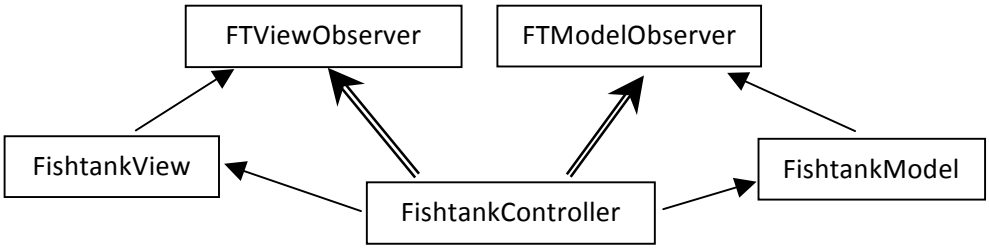
Download my HW6 fishtank solution. Although the code is written pretty well, it does not follow the Model-View-Controller pattern. The goal of this assignment is to rewrite it using MVC, with some added features.

The assignment has two parts. In this part, I want you to analyze my existing code and propose a re-design. The second part (to be assigned later) will have you do some implementation.

1. The fishtank code consists of 16 files. Draw a class diagram that depicts the interrelationships between the classes and interfaces declared in these files.

2. Of these classes/interfaces, some are exclusively view-centric, some are exclusively model-centric, and some contain a mix of view and model concepts.
a) Which ones are exclusively view-centric?
b) Which ones are exclusively model-centric?
c) For the ones that have a mix of view and model concepts, explain what part of their code is view-specific, and what part is model-specific.

3. Suppose you want to redesign the code to follow the MVC pattern. The general structure of this pattern gives rise to the following class diagram:



Of course, the view and model actually consist of several classes each. This diagram is pretending that *FishtankView* and *FishTankModel* are single classes, for simplicity.

We need to assign functionality to each class/interface in this diagram. Please answer the following questions:

a) What information does *FishTankModel* need to keep track of?
b) What information does *FishTankView* need to keep track of?
c) What methods should be in the *FTModelObserver* interface?
d) What methods should be in the *FTViewObserver* interface?
e) What methods will the *FishtankController* object need to call on the view?
f) What methods will the *FishtankController* object need to call on the model?
g) How do you propose to handle the animation timer? In particular, which class should be the action listener?

The answers to these questions are not easy. You first need to look carefully at the existing code and determine the functionality of each class. You then need to consider how this functionality relates to the MVC pattern.

I don't want you to implement anything yet. I will post part 2 of this assignment on November 28 at 5pm, describing what I want implemented. That part of the assignment will be due on Friday December 2.

WHAT TO SUBMIT: Create a document containing answers to the above problems, and submit it to Canvas by 5pm on November 28. I will post the solution at that time, because it will be needed for part 2. Therefore, late submissions will **NOT** be accepted.