CS161: Introduction to Computer Science
Lab Assignment 5

This week we are looking at more sophisticated examples of classes. In particular, we're looking at different ways in which classes can interact.

In-class we saw the `BankAccount` and `Bank` class. The `Bank` class contains methods that (1) take objects of type `BankAccount` as input parameters and (2) return objects of type `BankAccount` as return values.

In lab, we're going to explore another way in which two classes can interact: a class can have instance variables whose types are themselves classes.

To illustrate this concept, you're going to be creating primitive organisms which consist of a brain, a stomach, and a name. Download the starter code for this lab from the course webpage. You can use your `Brain` and `Stomach` classes from last week's lab or use the ones in the starter code. Your job is to *use* these classes to create your `Organism` class.

Organism Class

1. Begin by opening the `Brain` and `Stomach` classes and reminding yourself of their respective documentations. Make sure you're familiar with the methods in each class.

2. Open the `Organism` class. Your job is to finish writing this class. An organism should have a brain, a stomach, and a name. Fill-in the instance variables and the constructor for this class.

3. Next, fill-in the methods for the `Organism` class. The methods in this class are truly just *wrapper* methods – i.e., methods that call other methods to do the hard work. As a result, none of these methods should be longer than a few lines of code.

Tester Class

Once your `Organism` class is written and compiles, you can start creating actual organisms. Open up the `Tester` class and in the `main` method, create an organism. Have your organism perform at least 5 different actions. Print the state of your organism after each action.

Once you finish, call me or the student assistant over so we can see your work.

Extensions

There are lots of ways that you can extend this primitive organism:

1. Add asexual reproduction to the `Organism` class. In reality, if our organism could not reproduce, it would quickly become extinct. Create a method inside the `Organism` class called `reproduce()` that returns a new organism. The name of the new organism should somehow be related to the name of the original organism. For example, if your organism is named "ozzy" then the new organism might be named "ozzy 1".

2. Continuing with the previous idea, how could you modify your `Organism` class so that it keeps track of the number of children produced and numbers them accordingly? I.e., the first time the `reproduce` method is called, the new organism has the name "ozzy 1". The next time the `reproduce` method is called, the new organism has the name "ozzy 2". Then "ozzy 3", "ozzy 4", etc.

3. In your `Tester` class, create children and grandchildren for your original organism.

4. Continuing with this idea, how could you implement sexual reproduction? That is, how could you modify the `reproduce()` method in the `Organism` class so that it now requires two organisms to create a new organism?

▬▬▬ Submitting your lab assignment ▬▬▬

You should submit your `lab5` folder with the following classes: `Brain`, `Stomach`, `Organism`, and `Tester`. Rename the folder using both of your names. Then zip and submit via Moodle.