CS 1632 - DELIVERABLE 2: Unit Testing and Code Coverage

Brendan Murray and Max Benson

Code Available at: <https://github.com/BrendanMurray/cs1632>

Summary

The first issue we faced when working on this project was writing the code in such a way as to make it actually testable; we had to refactor our code a couple times after we had implemented all the functionality necessary just to make it possible to write unit tests for it. This included adding return values to functions that originally would just print a message directly to the console, and restructuring the way game-state was represented by the CoffeeMakerQuest class. Writing the tests themselves helped us determine how to best do this refactoring.

We also had some trouble applying the concepts of doubling and stubbing. For the most part the methods in the CoffeeMakerQuest class operate independently of one another, so there was little opportunity to stub methods since none really relied on any other functioning in a certain way. While the selectChoice() method does call other methods depending on its input, most of those calls do not alter the state of the function, so we didn’t think stubbing those methods would be appropriate. The two test cases where we were able to apply doubled objects and stubbed methods were the unit tests for the input value “D” on the selectChoice() method, since that method given that input will call the drink() method and function differently based on what drink() returns.



