**CSC 232 – Software Development**

**Project Assignment #2 – Parsing Basic Commands**

# Due Friday, October 27, by the beginning of class

This project assignment will involve somewhat more programming than previous assignments, so do not waste time getting started. You will begin work on your project by producing a class which represents a location where items may be found and a class that can process simple user commands.

Work with your group members to complete this assignment. The submission will be handled by committing the requested changes to your GitHub repository. Check the directions in the following sections for specific instructions. Many of the steps in this assignment can be completed independently. It is recommended that you split these tasks among team members.

***Part One – Prepare your Item Class***

If you have not yet chosen an Item class for your group from among those you developed for Homework #3, you should do so now. Make sure that the Item.java file has been added to your project.

***Part Two – Develop Your Location Class***

Your Location class should have, at a minimum, a name, a description, and a collection of Items (consider using an ArrayList of Items), which may be queried in several ways. In addition to getters and setters for the name and description, you should at least have methods to add an item to the location, retrieve an item given its short name, and retrieve a count of the number of items in the location. You can add additional methods if you wish.

***Part Three – Develop Your Simple Driver Class***

This class will have a main method and probably some helper methods. You will improve the driver to be more sophisticated in later projects, so this version is basically just for testing purposes. It should create a Location and add some Items to it, and assume that the player is in the Location created. Then it should go into an “infinite” loop where it prompts the user for a command on the console (using System.in and System.out), reads a line from the user, and processes the command that the user typed:

* If the command is “quit”, the loop should exit and the program ends.
* If the command is “look”, print out the description of the location, as well as the names of the items found there.
* If the command is “examine <*name>*”, get the item with the given name from the location and print its description. So, if the user types “examine sword”, and there is an item with the name of “sword in the Location, print the description of that item. If there is no item with that name in the Location, print some sort of error message.
* If the command is something else, print a message such as “I don’t know how to do that.”

You may choose to implement more commands, but these are the minimum.