CS 3667 -- Observer Pattern Problem, Workout 2020-08-28

Introduction

Imagine you are building a two-player computer game named “Goat Simulator”. Each player starts with a score of 0 and manipulates a goat across a terrain. The object of the game is to eat as much as possible in the given terrain. There are no point penalties; each goat only accumulates points by eating stuff.

To submit the assignment, zip up your code and submit the zip file on ASULearn.

Concept for Our Part of the Game Code

When a goat’s score changes, it triggers several changes:

1. The graphic image of the goat gets bigger.
2. The point display for the goat is changed.
3. The screen color scheme changes as each goat gets closer to a winning score of 1000 using the following table.

Score	Color Scheme
<500	Green
>=500 and <900	Yellow
>=900	Red

4. A check is made to see if the game has been won.

The Observer pattern is well-suited for this problem. Each player is a Source and each of the 4 changes is an Observer.

Program Design

- You are required to Java class Observerable and the Observer interface using the “pull” strategy for communicating the score.
- Have a Goat Class that extends Observable. In Goat, have a name field (String) and a score field (int). Write a getter and setter for the name. Initialize the score to 0 when a Goat is instantiated. Goat needs a method named addPoints(int) that adds points to the Goat’s score. addPoints(int) calls setchanged() and notifyObservers() to signal that the score has changed (in addition to changing the score for a Goat) for an int > 0 (do nothing if the parameter is <= 0).
- Create four classes that implement Observer, named ImageDisplay, PointDisplay, ColorDisplay, and WinnerDisplay.
  - update in ImageDisplay will output “<nameOfGoat> is made bigger.”
  - update in PointDisplay will output “<nameOfGoat> ate something and now has <score> points.”
  - update in ColorDisplay will have a field for the current color (initially Green). ColorDisplay’s update method will only produce output if the color scheme is changed: “The new color scheme is <color>.” (Note: the first goat to get to 500 or 900 will change the color scheme. Do NOT ever go back to a lower level.)

  - WinnerDisplay’s update method will only produce output when the first Goat breaks 1000 points. When a Goat breaks 1000 points, WinnerDisplay will output, “<nameOfGoat> wins.”
- The GoatSimulator class contains main. It declares, instantiates, and registers the Observers. Then make a series of <Goat>.addPoints (<n>) calls until one of the Goats wins the game. (Have at least 10 calls to addPoints before one of the goats acquires 1000 points.)