Software Engineering ESOF 322 HW3 (25 pts)

- Hand in a print out of your work either during class on **10/16** (midterm exam day), or the day after on **10/17**. If I am not in my office on Friday 10/17 you can leave it in my mail cubby in the main CS office before 5pm.
- Hand in your code, and a test program that clearly shows the registration of a sub-classed Singleton, and the lookup of a sub-classed Singleton.
- Use software to draw UML. Hand-drawn diagrams will receive 0. Hand in your complete class diagram.
- Use a partner. Clearly label every page with all names.
- Staple the pages together.

**Question 1 (25 pts)**

We are all familiar with the Singleton pattern. We learned the necessity of using this particular design pattern as a way to protect a single global resource in a system. In this assignment you will investigate this further and exercise your design skills by creating an architecture to manage a collection of Singleton patterns.

The global resource will be a set of `<name, Singleton>` pairs, where the name is a string that identifies a Singleton pattern, and Singleton is the actual instance of the Singleton pattern.

Since this architecture guards our global resource (in this case the collection of Singleton patterns), why not use the Singleton pattern itself as well (as discussed in class) to guard said collection?

You should use a generalized version of the Singleton pattern that guards the collection, but add the following to this class:

- A method to register a Singleton (i.e., add to the collection)
  
  ```java
  static void Register(String name, Singleton s);
  ```

- A method to lookup a Singleton by name:
  ```java
  static Singleton Lookup(String name);
  ```

- A static private collection that holds other Singleton patterns:
  ```java
  private static Collection<name, Singleton> registry;
  ```

Your UML class diagram should use generalization and look something like this: