**Assignment #3 – CSC 260 – Jenkins**

This assignment is based on the ideas from Chapter 3 from “Learning Object Oriented Programming”.

There are two deliverables here: the write-up and the .zip file of your Visual Studio files and .cs code files ( or GitHub / version control link ).

**Part 1 (10 points):** Write approximately 200 words on the topic of “What is the difference between encapsulation, information hiding, and abstraction?”. Please review the following source before proceeding. Cite this and at least one other source in your answer.

<http://www.tonymarston.co.uk/php-mysql/abstraction.txt>

**Part 2 (20 points):** Write a C# .NET Framework Console Application to demonstrate use cases of properties in C#.

Assumptions & Requirements:

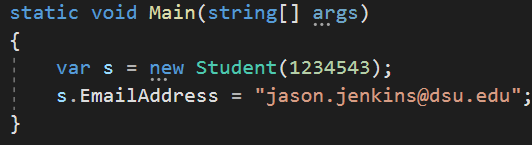
* You will need a Student class to accomplish goals of this assignment, as well as a main for object instantiation and information output (such as Console.Writeline).
* Your Student class should contain at least these fields: ID, Name, Major, StartDate, Anticipated Graduation Date, State/Province/Territory, Country, Email, Phone #, Mailing Address.
* The Student class should have at least two constructors.
* StudentID is read only and should be set by the end of the constructor.

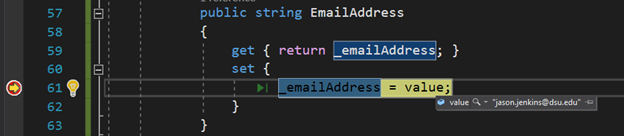
The application will do the following to accomplish this goal:

1) Log every instance of writing of a private field within a property setter. Before the program exits, output the activity log to the screen. Set the value of each property of each student to facilitate testing and operation of this logging functionality.

2) Choose at least one property for students, and make sure its setter has logic in it other than the default auto property setter code. Logging is not counted for these requirements. See the WatchDogAbility example from Chapter 3 slides/text. Historically, systems like WebAdvisor and Banner have allowed students to restrict the viewing of their personal information. Implement this system here, by ensuring the getters of properties check for this value before returning student information. Student ID and Name are exempt from needing to be checked.

3) Submit a screenshot of you stopping your program in debug mode in Visual Studio. Set one of the Student object's properties. Pause on the line where the new value is written to the private field. Make sure the new value is shown on the screen in your image (see example below, using Alt + Print Screen). Output all data for all students, let’s say a minimum of 10 students, with at least one with restricted personal information.





If you hover on “value”, you’ll see the value while the breakpoint is on that line.