# *Programming IV (420-B42-HR)*

# *Assignment 1 – Class and Test Libraries*

Date assigned: January 23, 2017

Date due: February 6, 2017

**Learning Objectives**

Upon successful completion of this assignment, the student will be able to:

* Create a C# class library for the classes needed for Happy Valley Kennels.
* Determine the attributes and methods for the classes for Happy Valley Kennels.

To do:

1. THIS IS AN INDIVIDUAL PROJECT
2. Create a class library called yourinitials\_HVK. The solution is am\_HVK.sln for me.
3. Looking at the data model and the user case descriptions and the prototype and prototype comments, construct a set of classes for the objects related to Happy Valley Kennels.
4. Start off by simply identifying the classes. You may not get them all to start with.
5. Add the member variables (attributes) of the class that you know about.
6. Add getters and setters as well as constructors for each class. Each class will have at least one constructor and likely more. Remember, there are required and non-required fields and you may want different constructors for each.
7. Consider any classes that may be derived from another (such as an Invoice class from Reservation). There may be others.
8. Consider the associative entities of the classes and how you will declare them in your class library. You will likely use a collection of some sort.
9. Using the use case descriptions, consider the operations that must be performed on each object.
10. Create a unit test library to test the classes you’ve created. Add test methods to test the various conditions of each class. You have to provide tests for each of the methods including the constructors, getters and setters. This is a precursor to the test testing you will be performing in assignment 2.

Do:

1. Prepare ONE class library with multiple classes.
2. Prepare a unit test project that is a second project in the HVK solution or its own solution. I would recommend the former, but it is up to you.
3. Document the test methods with comments before the method such that it is obvious what condition(s) the method is testing.
4. Make sure that the classes perform error checking as required.

Don’t:

1. Perform random testing…think and plan the tests you will perform.
2. Rely on a WPF or other program for the testing. The test methods should automatically test and report on all the test conditions.

**To submit**

1. A ZIP format (*username*\_B42\_A01.zip) containing all submitted files on Moodle.