# IT 230 Coding Activity Submission Template

Document your work in the coding activity by completing each of the following items:

1. Provide a screenshot of the output that resulted from running your program successfully in Visual Studio. See the coding assignment instructions for an example of what should be included in the screenshot. Your screenshot must include the following elements:

A screenshot of a computer

Description automatically generated

1. Copy and paste the source code text you wrote for this assignment from the \*.cs file into the space below. Only providing the \*.cs files or a screenshot does not meet the requirements for this part of the assignment. Code should be logically organized. It should also follow proper syntax and conventions noted in the Coding Activity Guidelines and Rubric.

**Included in the zip file**

1. Show that you understand the task by explaining the design of your program in the space below. Include the process and steps you took to write your code. Explain how you arrived at the solution to the problem and completed the activity.

To start this assignment I first looked at the MainWindow.Xaml.cs code to see what would be called in the new Course.cs file. I specifically took note of the setName code as I knew that would be needed in the new cs file. I also noticed that the xaml.cs file specifically called the Course class.

After taking note of the variable and class names I then created a new .cs file named “Course.cs” and created a new public class named “Course”. I knew this class needed to store, set, retrieve, and return the course names listed in the initial MainWindow file.

private string name - to store the course names

public void setName – to set the name of the course

public string getName – to retrieve the name of the course

public override string ToString – to override the ToString method and return the name of the course.

After creating this cs file I then ran the program with no issues.

1. Reflect on your learning experience and what you learned from completing the activity.

Working on this assignment taught me how to properly encapsulate data in a C# class. By declaring the name field as private and using public methods like setName and getName, I learned how to control access to the class’s internal data. This ensures that the data can only be modified or retrieved through these methods, adding a layer of protection. I now understand how this design pattern helps prevent errors and maintains consistency, as the class itself manages how its data is handled.

Another concept I picked up was method overriding, specifically with the ToString method. I knew that ToString existed but didn’t realize how much more useful it becomes when customized. By overriding it in the Course class, I was able to return a meaningful string (the course name) instead of the default object representation. This not only made the output more useful but also improved the readability of the code. Overall, this assignment helped me build a stronger foundation in object-oriented programming, particularly in encapsulation and method overriding.