# IT 230 Coding Activity Submission Template

Submit your work on the coding activities for Modules One, Two, Three, Four, and Six in this document. In addition to this document, you should submit a ZIP file containing all your Visual Studio project files and source code that can be run in Visual Studio on a different computer.

For each coding activity, complete the following steps:

* Download and rename this document to meet the file naming conventions requested in the assignment instructions.
* Fill in the required information below by replacing the bracketed text with the relevant information.
* Submit this document and your ZIP file for grading and feedback. Your ZIP file should follow the same naming conventions.

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:
   1. Your last name as the first printed text on the screen
   2. Verification that the program is fully functioning and data results are accurate for the given problem

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.

namespace CreateClassesObjs

{

public class Course

{

private string name;

// Method to set the name field

public void setName(string courseName)

{

name = courseName;

}

// Method to get the name field

public string getName()

{

return name;

}

// Override ToString method to return the name field

public override string ToString()

{

return name;

}

}

}

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 complete the WPF application, the task involved creating a Course class and integrating it into an existing Visual Studio project. The class required a private string field for the course name, methods to set and retrieve this name, and an overridden ToString method for display purposes. First, I created the Course.cs file, implemented the Course class with the necessary methods, and integrated these into MainWindow.xaml.cs. This involved instantiating Course objects, setting their names, and adding them to a ComboBox in the Window\_Loaded event handler. The button\_Click method was updated to add the selected course to a ListBox. Additionally, I updated the window title to "Couch's Copy" in MainWindow.xaml. Finally, I built and tested the application, ensuring it functioned correctly and met all requirements.

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

This activity taught me valuable lessons in class design, WPF application development, and project integration. I learned the importance of encapsulation, how to override methods, and how to handle WPF control interactions and events. The process also highlighted the significance of thorough testing and troubleshooting to ensure seamless integration of all application components. Additionally, the activity emphasized the need for organized project files and detailed documentation of development steps. Overall, this experience enhanced my understanding of object-oriented programming and WPF development, reinforcing the importance of methodical problem-solving and attention to detail.