



## Overview

In the professional landscape, there are times when multiple different programming languages will be used together on one development project. When this occurs, you will need to be able to understand the role each different language plays in the overall project and how they connect. For this assignment, you will be taking time to practice working in two different languages while designing code that can complete some simple mathematical tasks in C++ and Python.

## Prompt

Begin by opening a Visual Studio project file that has correctly combined C++ and Python, which you already set up in a previous module. Remember you will be working in Release mode rather than Debug mode for this work because it includes Python. Next, add the code from the [CS210 Starter CPP Code](#) and [CS210 Starter PY Code](#) files in the appropriate tabs of your project. Be sure you have already reviewed the [Module Six Assignment Video](#) or its [transcript](#), as they contain important information regarding what is included in the starter code. Understanding the starter code will help you effectively build off the C++ and Python components you have been given in order to complete the design of a simple math-based program.

The simple math-based program you create will need to be able to print the multiplication table for a given number and double the value of a given number.

As you work, continue checking your code's syntax to ensure your code will run. Note that when you compile your code, you will be able to tell if this is successful overall because it will produce an error message for any issues regarding syntax. Some common syntax errors might be missing a semicolon, calling a function that does not exist, not closing an open bracket, or using double quotes and not closing them in a string, among others.

Specifically, you must address the following rubric criteria:

- **Design a menu with appropriate user interactions and checks for valid entry.** Use C++ to successfully complete this criterion. Your simple program will need a menu that can validate user input and is easy to use. It needs to include options for the display of a multiplication table, doubling a value, and exiting the program. If either of the first two options are selected, then users need to be prompted to input a numeric value. The menu should be displayed using a loop, where the user can choose to exit the program only by selecting option 3. Any user input other than 1, 2, or 3 should result in an error message that returns the user to the menu. An example menu might look like the following:

```
1: Display a Multiplication Table
2: Double a Value
3: Exit
Enter your selection as a number 1, 2, or 3.
```

- **Create code that prints a multiplication table for a given numeric value.** Both C++ and Python will be necessary to successfully complete this criteria. Be sure to focus on their interactions as you work. Consider the following steps to help organize your code design. Note that you should have already written C++ code that prompts a user to input a number while working on the menu portion of this assignment.
  - Write C++ code that reads and passes a number, as an integer, to Python. C++ should also call a function in Python to display the multiplication table for the passed parameter. Note that you will be creating that function in the next step. For this step, be sure to check the starter code you were given and use the applicable components.
  - Write Python code to create a multiplication table for the given integer. Name this function *MultiplicationTable* for consistency. The printed table should include values for the multipliers one through ten. An example result is shown below.

6 X 1 = 6  
 6 X 2 = 12  
 6 X 3 = 18  
 6 X 4 = 24  
 6 X 5 = 30  
 6 X 6 = 36  
 6 X 7 = 42  
 6 X 8 = 48  
 6 X 9 = 54  
 6 X 10 = 60

- **Create code that doubles a given numeric value.** Both C++ and Python will be necessary to successfully complete this criteria. Be sure to focus on their interactions as you work. Consider the following steps to help organize your code design. Note that you should have already written C++ code that prompts a user to input a number while working on the menu portion of this assignment.
  - Once the number has been read from the user in C++, call the callIntFunc function. Then pass the Python function name (which you will create in the next step) and the value entered by the user as parameters to the callIntFunc function.
  - Write a Python function to receive the user input and then return that value multiplied by two. For consistency, name the function you create DoubleValue. Refer to the example in your starter code as you work, but remember the example is squaring the value (or multiplying the value by itself) rather than doubling it.
  - In C++, receive the value sent from the Python function (DoubleValue) and display the value on the screen.
- **Apply industry standard best practices such as in-line comments and appropriate naming conventions to enhance readability and maintainability.** Remember that you must demonstrate industry standard best practices in all your code to ensure clarity, consistency, and efficiency. This includes the following:
  - Inputting validation and error handling to anticipate, detect, and respond to run-time and user errors (for example, make sure you have option 3 on your menu so users can exit the program)
  - Inserting in-line comments to denote your changes and briefly describe the functionality of the code
  - Using appropriate variable, parameter, and other naming conventions throughout your code

## Guidelines for Submission

Submit your completed work as a ZIP file, including all Visual Studio project files that are required to run the program. Reference the Visual Studio Export Tutorial, linked in this week's Resources section, for guidance on how to download the necessary ZIP folder.

Module Six Assignment Rubric

| Criteria | Exemplary  | Proficient   | Needs Improvement  | Not Evident                     | Value |
|----------|--|--|--|---------------------------------|-------|
| Menu     | Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner | Designs a menu with appropriate user interactions and checks for valid entry (85%) | Shows progress toward proficiency, but with errors or omissions; areas for improvement may | Does not attempt criterion (0%) | 20    |

|                             |   |   |   |                                 |    |
|-----------------------------|---|---|---|---------------------------------|----|
|                             | (100%)  |   | include clearly prompting the user to interact with the menu, effective error handling, or formatting the information like a menu to ensure each option is on a separate line (55%)   |                                 |    |
| <b>Multiplication Table</b> | Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner (100%) | Creates code that prints a multiplication table for a given numeric value (85%) | Shows progress toward proficiency, but with errors or omissions; areas for improvement may include printing all multipliers from one to ten and not stopping at nine, presenting the information in an easily readable format, or effective communication between languages (55%) | Does not attempt criterion (0%) | 35 |
| <b>Doubling a Value</b>     | N/A   | Creates code that doubles a given numeric value (100%)                          | Shows progress toward proficiency, but with errors or omissions; areas for improvement may include ensuring the Python function does not perform a different mathematical operation or effective communication between languages (55%)  | Does not attempt criterion (0%) | 35 |
| <b>Best Practices</b>       | Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or                        | Applies industry standard best practices such as in-line comments and           | Shows progress toward proficiency, but with errors or omissions; areas for  | Does not attempt criterion (0%) | 10 |

|        |                           |   |  |  |      |
|--------|---------------------------|---|--|--|------|
|        | creative manner<br>(100%) | appropriate naming<br>conventions to<br>enhance readability<br>and maintainability<br>(85%) | improvement may<br>include naming<br>conventions or in-<br>line comments (55%) |  |      |
| Total: |                           |   |  |  | 100% |