

CS 210 Module Five Assignment Guidelines and Rubric

Overview

Reading and writing to files is a constantly used way of transferring data in many different kinds of programs. Websites, such as news sites, and various phone applications are both excellent examples of this widespread use. Phone apps in particular will always read and write to files, since the alternative of writing to databases would be too large for their purposes. Another example is an application you use for daily tasks such as writing in a Microsoft Word file and uploading it as a Brightspace submission (to later be downloaded by your instructor). All of these demonstrate the importance and prevalence of reading and writing to a file as a method of information sharing.

Later in this course, you will need to be able to share information between C++ and Python as part of completing a larger task. Practicing reading and writing to a file this week will help prepare you for that later work. Although you will eventually have a broader understanding of working with files in multiple different programming languages, for this assignment your work should only be completed in C++.

Prompt

In this assignment, you are presented with a text document that includes six cities and their average yearly temperature in Fahrenheit. Your goal is to read that data, convert it to Celsius using the provided formula, and then write that new data to its own file.

To begin your work, open Visual Studio and create a new C++ project. Save the provided `FahrenheitTemperature.txt` document in a location where you will easily be able to access it while you work in Visual Studio. For this assignment, you will be submitting only your `C++.cpp` file. You do not need to submit the final converted data; the C++ code you create just needs to be able to generate that file.


Specifically, you must address the following rubric criteria:

- **Develop code to read data from a text file.** Your work should be completed using C++. Read the provided document, [FahrenheitTemperature.txt](#), which includes data on the average yearly temperature for six different cities in degrees Fahrenheit. Note that a space separates each city from its temperature. Assume the city's name does not include any spaces or special characters (the name should consist of only a single word). Also assume the provided temperature is presented as an integer. Consider the following steps as you work:
 - Open the provided file so it is ready to be read. Remember the file is named `FahrenheitTemperature.txt`. Watch out for the class you use, and make sure it is for reading a file and not writing to a file.
 - Read data from the provided file. Remember, to read this file you will need to declare a variable. Begin by reading the first value and putting it in the first variable. Then read the next value and put it in the second variable.
 - Once this is complete, be sure to close the file. This releases the file so it can be used again.
- **Develop code to write data to a text file.** Your work should be completed using C++. Title the new document you are creating *CelsiusTemperature.txt*. The name of the output file needs to be different from the name of the input file so you do not overwrite and erase the input file. Consider the following steps as you work:
 - Declare a variable to point to the file that will be written to. Watch out for the class you use, and make sure it is for writing to a file and not reading a file.
 - Create the code instructions for writing data to the new output file. In this new file, include space for both the name of the city and the temperature in Celsius for each city included in the original input file. You will need to complete the Fahrenheit-to-Celsius conversion calculation before you write to the new file. Use the following formula to make this conversion. Note that °F represents the temperature in degrees Fahrenheit while °C represents the temperature in degrees Celsius.

$$(\text{°F} - 32) \times \frac{5}{9} = \text{°C}$$

 Listen

 Dictionary

 Translate

- Close the file once you are done writing to it. If you attempt to look at the results in the file before completing this step, your file may appear empty.

What to Submit

Submit your finalized code as a single CPP file. Remember, you can locate where the Visual Studio IDE is automatically saving your C++ work by right-clicking on the project title in the Solution Explorer, then selecting Open in File Explorer.

Module Five Assignment Rubric

Criteria	Proficient (100%)	Needs Improvement (70%)	Not Evident (0%)	Value
Read Data	Develops code to read data from a text file	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include avoiding minor mistakes in the code that affect overall functionality such as mixing up << and >>	Does not attempt criterion	45
Write Data	Develops code to write data to a text file	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include using different file names, using << and >> appropriately, adding a space between the string and the integer in the output file, or displaying each element on a separate row	Does not attempt criterion	45
Best Practices	Applies industry standard best practices such as in-line comments and appropriate naming conventions to enhance readability and maintainability	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include naming conventions or in-line comments	Does not attempt criterion	10
Total:				100%