Assessments

The programming solution for each chapters' questions can be found in our GitHub repository at the following URL: https://github.com/PacktPublishing/Demystifying-Object-Oriented-Programming-with-CPP/tree/master. Each full program solution can be found in the GitHub under the appropriate chapter heading (subdirectory, such as `Chapter01`) in the subdirectory `Assessments`, in a file that corresponds to the chapter number, followed by a dash, followed by the solution number in the chapter at hand. For example, the solution for question 3 in chapter 1 can be found in the subdirectory `Chapter01/Assessments` in a file named `Chp1-Q3.cpp` under the aforementioned GitHub directory.

The written responses for non-programming questions can be found in this file. Should an exercise have a programming portion and a follow-up question, the answer to the follow-up question may be found both in the next sections and in a comment at the top of the programming solution on GitHub (as it may be appropriate to review the solution in order to fully understand the answer to the question).

Chapter 1 – Understanding Basic C++ Assumptions

1. A `flush` may be useful, rather than and `endl`, for clearing the contents of a buffer associated with `cout` for the situations where you do not wish the cursor to be advanced to the next line for output. Recall, an `endl` manipulator is merely a newline character plus a buffer flush.
2. Choosing a pre versus a post increment for a variable, such as `++i` (versus `i++`) will have an impact on the code when used in conjunction with a compound expression. A typical example would be `result = array[i++];` versus `result = array[++i];`. With the post-increment (`i++`), the contents of `array[i]` will be assigned to `result` and then `i` is incremented. With the pre-increment, `i` is first incremented and then `result` will have the value of `array[i]` (that is, using the new value of `i` as an index).
3. Please see Chapter01/Assessments/Chp1-Q3.cpp in the GitHub repository.