B BELLEVUE
C O L L E G E

Project 1: Quotations

Due: See Calendar
Use Assignment Tool
Points Possible: 30

Objective: Create an application which will display to the console a "favorite" quotation and some information about it.

Solution Name: yourLastName Project1Sol (e.g. Unwin Project1Sol)
Project Name: yourLastNameProject1
Source File Name: Quotation.cs, **Class Name**: Quotation

Follow the **Software Development Standards** for implementing the required documentation, naming conventions, whitespace, and other details.

A sample of what the output screen **may** look like is attached. This is just an example, and you may add more, (not less) to the textual comments displayed on the console.

Application Specification:
1. When the program first starts, there must be a brief message explaining the purpose of the application – ensure readability, sensibility and **correct spelling**.
2. The application must display individual paragraphs to the user as required below with identifying labels and you, the programmer, will control the end of each line so as to appear that the text is word wrapping in an easy-to-read manner. In other words, do not let the computer automatically wrap the text in the console window. Each of the following bullets is a single paragraph to display:
    - one of your favorite quotations
    - the name of the person who said this – anonymous quotes will not be accepted
    - an explanation of the situation or background for the quote and why it is one of your favorites
    - the source of the quote (e.g. book, web site, magazine) in a simple, easy, readable manner. [not a direct link, be more generic]
3. When all the information has been displayed, as a final paragraph, write a brief message to include:
    - the programmer's name (you)
    - the application has ended
    - and what the user needs to do to close (exit out of) the application.

Requirements:
Along with the specifications, the code must include:
4. File documentation as detailed in the **Software Development Standards,** page 2.
5. As a comment within your code, where you state the "source" of your quote, cite the quote appropriately (including the complete URL, or if a book – the title, author, date, etc). The comment line will not be displayed to the user.
6. Two of the many WriteLine () methods in your code must each display multiple lines.
7. The program must run with Debug/Start Debugging (F5).

Hints for Success
- Review your application and check that you have correct spelling, correct English grammar, and that the text "wraps" in an easy-to-read (non-cluttered) format.
- Have a friend run your application; they may suggest changes to the interface for smoother interaction.
- If you have questions, please post on the discussion board under the appropriate topic, but do not post your code. In answering another student's question, do not post "answers" (code), but you may provide guidelines/hints on how to obtain the desired result.
- Start early and plan your time for questions in class and on the discussion board. Do not wait until the last minute to ask questions.

Follow the **Program Development Life Cycle** to plan, design, code, and test your application.

1. Analyze the specifications. Do you have questions? If so, ask them.

2. Plan your steps one by one and write them down. For example:
   Step 1: Display an introduction
   Step 2: Display my favorite quote (control when the line "ends" on the monitor (word wrapping))
       etc.

   This is your pseudocode.

3. After the planning and design, open up VS IDE and write the comments to document the code as required (explaining who, when, why, and so on). In the Main method, write the steps of your pseudocode within your code as comments.
   ```
   //Display an introduction

   //Display my quote (control word wrapping)

   . . . etc . . .
   ```

4. Next, write the code underneath each of the comment lines
   ```
   //Display an introduction
   Write the C# statement(s) here

   //Display my quote (control word wrapping)
   Write the C# statement(s) here
   ```

5. As you type the code, fix any syntax errors. Intellisense really helps! Run the code, does it have any logic errors? Fix those using the debugging tools and other debugging techniques (more in depth later in the course).

6. Once you meet the project's requirements, then add final touch ups. Consider ease of use, the user's experience; fix any grammatical, spelling errors. Add your "flare", be creative, and have fun.

Time Sheet Requirement
1. Create a Word document that will have your time sheet for this project. Include the estimate, actual and total hours. Review the format as shown on page 8 of the Software Development Standards. If you want to include your pseudocode, you may, though, for now it is not required.
2. Save as *yourLastName*MemoP1 (e.g. UnwinMemoP1)

Turn in:
1. The complete Visual Studio solution you developed.
2. Place the time sheet outside the VS solution folder and submit all files in one Zip file named *yourLastName*P1 (e.g. UnwinP1.zip).
   The zip file hierarchy will look like the following:
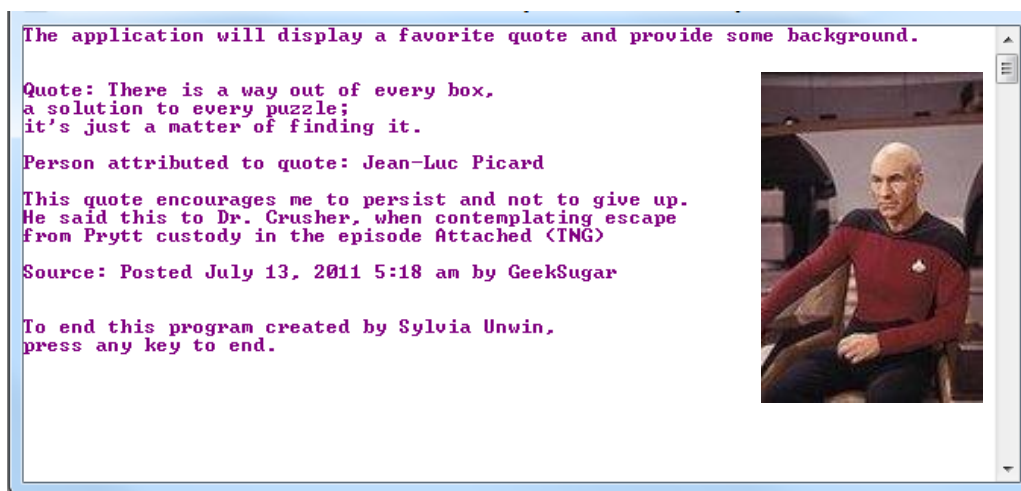   UnwinP1.zip
   　　UnwinMemoP1.docx
   　　UnwinProject1Sol <folder>

It is your responsibility to ensure that all files are submitted and that I don't receive a "shortcut", or missing a file. If I don't have the correct files, I cannot grade your project and therefore, you will receive a zero for the assignment. Review the syllabus paragraph identified as "Multiple Submissions – Same Assignment"

If you have questions, please post in the appropriate category/topic, but do not post any actual code. In answering another student's question, do not post actual code, but you may provide guidelines/hints on how to obtain the desired result.

Output Example
You may use other text for describing and explaining your quotation, but make sure to include the homework specifications, formatting (blank lines, new lines, etc).

Note: the picture is embedded within this document and not on the output console. ☺