|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CS 1400 Lab #18**  **Leap Years Implementation**  **Version 1.0**  **Introduction**  In this exercise, you will create a method and test it with an instructor provided Console program.  **The problem**  In the previous lab you created the Pseudo-Code for a method that determines if a given year is a leap year. In this lab you will write the actual code for your method and add it to the program supplied herein.  **Writing the Code**   * Start a new Console Project. * A copy of the .cs file is located below. Where shown in the code below, copy your Pseudo-Code from the previous lab. Now write the class ***LeapYear*** and other required class and methodcode, following the steps in your Pseudo-Code. Name your method ***IsLeapYear***. Write a complete Method Prolog for your ***IsLeapYear*** method and any other methods. Trace through your class and method code, by hand and make sure if it makes sense to you. Then compile your code. It may take several tries to get a program that has no syntax errors. Once your code compiles successfully execute your program and test it. You can only change the code in the ***IsLeapYear*** method in the following code.   **// File Prolog**  **// Name: <student\_name>**  **// CS 1400 Section <number>**  **// Project: CS1400\_Lab\_18**  **// Date: 11/01/2014**  **//**  **// I declare that the following code was written by me or provided**  **// by the instructor for this project. I understand that copying source**  **// code from any other source constitutes cheating, and that I will receive**  **// a zero on this project if I am found in violation of this policy.**  **// ---------------------------------------------------------------------------**  **using** **System**;  **static** **class** Program  {  /// <summary>  /// Purpose: Entry point to your C# program  /// </summary>  **static** **void** **Main**()  {  **new** LeapYear()**.LeapMenu**();  }**//End Main()**  }**//End class Program**  /// <summary>  /// Your class  /// </summary>  **class** LeapYear  {  **string** **inputStg** **=** **null**;  **char** **decision** **=** **'\0'**;  **int** **year** **=** 0;  **public** **void** **LeapMenu**()  {  **// main program loop**  **// get and validate response**  **do**  {  Console**.Clear**();  Console**.WriteLine**(**"----------------- Leap Year Tester -----------------\n\n"**);  Console**.Write**(**"\tTest for leap year Y)es or N)o: "**);  **switch** (((**inputStg** **=** Console**.ReadLine**()) **!=** **""**) **?**  (**char.ToLower**(**inputStg**[0])) : **'X'**)  {  **case** **'y'**:  **TestLeapYear**();  **continue**;  **case** **'n'**:  Console**.WriteLine**(**"\n\t..... Goodbye! ....."**);  Console**.ReadLine**();  **return**;  **case** **'X'**:  **inputStg** **=** **"<empty>"**;  Console**.WriteLine**(**"\n\tInvalid input -> {0}, try” +**  **"again!"**,**inputStg**);  Console**.ReadLine**();  **continue**;  **default**:  Console**.WriteLine**(**"\n\tInvalid input -> {0}, try” +**  **"again!"**,**inputStg**);  Console**.ReadLine**();  **continue**;  }  } **while** (**true**);  }**//End LeapMenu( )**  **public** **void** **TestLeapYear**()  {  **do**  {  **// declare some variables**  Console**.Clear**();  **// get a year from the user**  Console**.Write**(**"\n\tEnter in a year (example, 1923) or (Exit) Press Enter: "**);  **if** (**int.TryParse**(((**inputStg** **=** Console**.ReadLine**()) **==** **""**) **?** **inputStg**  **=** **"X"** : **inputStg**, **out** **year**) **==** **false**)  {  **if** (**inputStg** **==** **"X"**)  **return**;  Console**.WriteLine**(**"\tInvalid year value -> {0}"**,**inputStg**);  Console**.ReadLine**();  **continue**;  }  **else**  **// test it with method and output result**  Console**.WriteLine**(((**IsLeapYear**(**year**)) **?** **"\n\tIs a leap year ->**  **{0:D4}"** : **"\n\tIs not a leap year -> {0:D4}"**), **year**);  Console**.ReadLine**();  **continue**;  } **while** (**true**);  }**//End TestLeapYear( )**  **// The IsLeapYear method**  /// <summary>  /// Purpose: Determines if the year passed as a parameter is a leap year  **// it is a leap year, if the year is divisible by 4,**  **// but not divisible by 100 unless it is divisible by 400**  /// </summary>  /// <param name="year">the year to test as an int</param>  /// <returns>true if its a leap year otherwise false</returns>  **public** **bool** **IsLeapYear**(**int** **year**) { **return** **false**; }**//End IsLeapYear( )**  **//**  **// Parameters: The year, as an integer greater than 0**  **// Returns: A Boolean, true if the year is a leap year**  **// ------------------------------------------------------------------**  **// write the code for the method IsLeap above.**  }**//End class LeapYear**  **File(s) to Submit:**  Place your complete project folder in a zip file and name the zip file Lab\_18\_your-initials\_V1.0.zip. For example, I would name my file Lab\_18\_DAF\_v1.0.zip. Submit this assignment as Lab #18 on Canvas.   |  |  |  |  | | --- | --- | --- | --- | |  | **Grading Checklist** |  |  | | # | Program | 1st Submission | 2nd Submission | | 1 | Meets & works to specifications | Correct | Xnot | 6 points | | 2 | Error Free, elegant & efficient | C | X | 4 points | | 3 | Pseudo-Code | C | X | -3 points | | 4 | Style Guidelines | C | X | -2 points | | 6 | Source Files(s) & Formatting | C | X | -2 points | | 7 | Project Prolog | C | X | -1 points | | 8 | Function Prologs | C | X | -1 points | | 9 | Zip Filename | C | X | -1 points | | 10 | Lab & Project Names | C | X | -1 points | | 11 | Zip File is invalid or will not unzip | Lab = 0 pts | Lab = 0 pts | |  | Total Points | 10 | 0 | 10 | 0-9 | |