CIS 246 – C++ Programming

Program:	**Week 02**
Points:	**20**
Chapter(s):	**3**
File(s) to Submit:	**Date.h, DateDriver.cpp (as a zip)**

Description

Write a C++ program that uses a class (Date) and a driver (DateDriver) to display a date to the screen.

You must write **a driver and a class** using separate files for full credit.

Learning Objectives

In this assignment, you will practice:

- Writing a driver with a separate class
- Writing a class with a constructor, and getter and setter functions
- Calling functions located in a class.

Requirements for the Date Class

Data Members

1. integer to store the month
2. integer to store the day
3. integer to store the year

Be sure to use the appropriate access modifier for these.

Member Functions

1. **One 3-argument constructor** – The constructor has three integer arguments representing the month, day, and year values. Use member initializers.

2. **Getters and setters for each of the above integers** – Each data member needs its own setter and its own getter function, for a total of six functions. Be sure to use the const keyword where appropriate.

3. **displayDate function** – This function uses the integer data members and displays the date to the screen using the format mm/dd/yyyy.

Requirements for the DateDriver Program

Data members – none

Functions

1. **main**

   - Instantiate an object of type Date, passing any three integers to the constructor. Use integers that form a valid date. You can hard code values here.
   - Call the displayDate function of the Date object to display this date to the screen.
   - Prompt the user for three new integer values. Be sure to use prompts that explain what the user needs to do.
   - Call the appropriate setter functions of the Date object, using the three values input by the user.
   - Call the displayDate function of the Date object to display the new date to the screen.

General Requirements

For complete credit, you must:

1. **MEET ALL REQUIREMENTS ACCORDING TO THE INSTRUCTIONS** – Follow the instructions as written for completing this project, even if you [think you] know a "better" way to do something.
2. **INCLUDE COMMENTS** – Include comments in your code. There must be a comment at the top of each source code or header file that includes your name, the assignment number, and a description of the code in that file. There must be comments at each important step in your algorithm that describes that step.
3. **FOLLOW BEST PRACTICES** – Follow best practices in C++ programming, including, but not limited to, appropriate use of private/public, appropriate use of classes and/or header files, sets & gets, white space, alignment, meaningful variable names, naming conventions, using statements, etc. Points will be deducted for sloppy code that is hard to read, even if it works, so pay attention to these details.
4. **SUBMIT ALL FILES BEFORE THE DUE DATE** – Submit a .zip of ONLY source code files to the dropbox for this assignment on Canvas before the due date. Do not submit anything except .cpp and/or .h, within a zip. Do not submit .exe files. Do not submit a folder structure. Do not submit project files from an IDE.

Submission

Submit a compressed folder (zip) of ONLY the Date.h and DateDriver.cpp source code file to the dropbox for this assignment on Canvas before the due date. Do not submit anything except Date.h and DateDriver.cpp, within a zip. Do not submit exe files. Do not submit project files from an IDE.

Sample Run

```
Date is 7/22/2020
New month: 9
New day: 5
New Year: 2020
Date is 9/5/2020
```