

Clock Class

This assignment is designed to give you practice with object-oriented programming, particularly defining classes, and with the use of multiple files. This project will parallel the development of the *date* class described in class.

The Project

You should create a class definition for dealing with “clock” times. This should allow you to create a time object that records the hours and minutes for a time. In addition to the default constructor (`__init__`), you should have, at a minimum, four methods in your class—`DisplayTime`, `DisplayMilitaryTime`, `IncrementMinute`, and `MinutesSinceMidnight`. The first of these methods should return a string giving the time in the standard AM/PM format, e.g., “3:00 PM”. The second method should return a string giving a time in military format, e.g., “1500”. The third method should increment the time stored in the object by one minute with appropriate wrap around. The last method returns the number of minutes between the time and the previous midnight. You may need other methods and are free to implement any you feel you need.

Once you have this class coded, you should use it to write an interactive program that calculates the elapsed time between two times that uses the class definition. For example, if the inputs were 8:15 PM and 1:00 AM, your program would display 4 hours and 45 minutes. You may assume that both times are within a 24-hour period but not necessarily the same day.

Pair Programming

For this assignment, you will work in pairs. (One pair may be a team of three.) Please choose a partner that you haven’t worked with before. If you were on a team of three for the last assignment, you may NOT be on a team of three for this assignment. Be sure you have compatible schedules before you commit to working together. Your pair (team) will submit one program and you will each receive the same grade.

Constraints

Your class code should be implemented as a module in a single file. The demonstration code for your program should be in a separate file. *Be sure to include doc-strings with your class definition.*

What to Hand In

1. Turn in *all* files used for your project by attaching the files to an email. See due dates listed below. Be sure to keep archival copies of your files! There should be one email submission per pair with two attached files. Be sure your name is part of each file name.
2. Turn in one copy per team of the printout of sample output for your code.
3. Each of you should also turn in the answers to the following questions.

Questions

Answer the following questions and turn in your answers with your program.

1. How much time did you spend planning before you started coding?
2. How much time did you spend coding?
3. How much time did it take to correct the errors in your program?
4. What problems did you have getting your code to work?
5. How did pair-programming work for you? What worked well? What problems did you have?

This project will count as three routine assignments.

Due Dates

Class Definition: Start of class, Wednesday, April 22.

Finished Project: Start of class, Friday, April 24.