CSC 7200
Final Project

The Calendar Program

The purpose of this final project is to give you a chance to use some of the data stream tools we have been discussing in a simple application that builds upon the Model-View-Controller (MVC) Design Pattern (DP). You will write a calendar application which will allow the user to select a date, and then either retrieve a previously stored calendar entry or save a new calendar entry for that date.

Your program should include a Graphical User Interface (GUI) interface that allows the user to specify the month, day, and year of the calendar entry. The GUI should also have a text area for displaying and editing a particular entry. It should also have two buttons: one for saving an entry and the other for retrieving an entry.

Required program elements

1. Your user interface must allow the user to enter the month, day, and year. You can do this using text fields for input or you can use ComboBoxes if you feel adventurous.
2. The only GUI components which create events that your program needs to handle are the save and retrieve buttons.
    a. Don't go overboard making your GUI beautiful! We are just looking for basic functionality here!
3. You must have a **separate class** which manages the calendar data. You will have a minimum of three classes in your application, a user interface class, the calendar manager class, and a calendar test class. The user interface class creates an instance of the calendar manager in its constructor and stores it in a member variable.
4. The calendar manager must provide methods which support saving a specific calendar entry and retrieving a specific calendar entry. The interfaces must be defined to **only pass a single day's calendar entry** across the interface.
5. The calendar manager must store calendar data into files according to YEAR-MONTH. That is, all the calendar entries for July 2015 must be stored in the single file called 2015-07.
6. The calendar manager must use ObjectInput/OutputStreams to read/write data from/to files. The calendar manager will use an array to store String objects. The position of a String in this array corresponds to the calendar entry for a specific day.
7. The save method of the calendar manager will need to determine if a file exists for the requested month and year. If so, the object from that file must be read into the calendar manager. Otherwise, the calendar manager must create an empty

String array. The new entry must be saved to the appropriate day’s location in the array. The modified array must be saved to the appropriate file.

8. The retrieve method of the calendar manager will need to determine if a file exists for the requested month and year. If not, return an error string indicating that there is no such entry. If the file exists, read the String array from the file and locate the requested day’s entry. If this entry is null, return an error string indicating that there is no such entry; otherwise return the entry.

Deliverables

- Screenshots of the running program
- Source Code for each class
- Paste Screenshots + Source Code into a single DOC/DOCX file

Final Project Grading Criteria

Program 1 160 pts.
Screenshots + Source Code 40 pts.

Total 200 pts.

*** Program1:

- GUI doesn't allow the user to enter the month, day, and year: -12/each
- GUI does not have Save or Retrieve buttons: -20/each
- Calendar data not managed in separate class, Calendar Manager: -40
- Does not have a User Interface class or a Calendar Test class: -20/each
- Calendar Manager Class does not have methods to save or retrieve a specific calendar entry: -12/each
- Calendar Manager Class does not store calendar data into files by month-year: -20
- Calendar Manager Class does not use ObjectInput/OutputStreams and an array of String objects: -20
- Save method does not use existing file data: -12
- Retrieve method does not use existing file: -12

- Things don't work in general: -8/instance
- No screenshots: -40