

Lab 3-2 Grading = 3%

Objectives

Applied

1. Use Visual Studio to perform any of these operations:
 - Open and close an existing C# project or solution
 - Display the Form Designer for each of the forms in a project
 - Display the Code Editor for each of the forms in a project
 - Use the Solution Explorer to review the files in a project
 - Open, hide, and adjust the windows for a project
 - Build and run a project

Knowledge

1. Describe the main difference between a desktop application and a web application.
2. Name three platforms you can use to develop Windows desktop applications with .NET.
3. Name two platforms you can use to develop web applications with .NET.
4. Name three programming languages that you can use to develop .NET applications.
5. Distinguish between .NET Framework and .NET Core.
6. Describe the two main components of .NET.
7. In general terms, describe the C# compiler, Microsoft Intermediate Language, the assembly, and the Common Language Runtime.
8. Describe the use of each of these windows in the Visual Studio IDE: Form Designer, Code Editor, and Solution Explorer.
9. In general terms, describe how to use Visual Studio to target a version of .NET.

Reference Lectures:

PowerPoint: Ch 10 (review and Practice Ch 12 from book or slides)

Video Lecture Part 4 Multi-Forms and Controls (YouTube Video)

Video Lecture Part 5 Classes (YouTube Video)

Video Lecture Part 6 Inheritance (YouTube Video)

Due Dates: **Thursday, Feb 18 @ 6:00 pm**

Project 3-2 Assign tickets with time slots

For this project, you'll develop an application that assigns tickets that include a time slot when a guest can return to visit an attraction without waiting in line, similar to the way the Fast Pass system works at Disneyland. Prerequisites: chapters 1-12.

The Tickets form

The screenshot shows a window titled "12:25:05 PM (Open)". Inside, there are three main sections. The top section, "Guests with the following tickets may now enter:", contains a text box with "1 - 5". The middle section, "Ticket Availability", shows "Total tickets outstanding: 7" and "Next available entry: 12:35 PM", with an "Issue Ticket" button below. The bottom section is a list box containing tickets 6 through 12 with their respective times. At the very bottom are "Options" and "Exit" buttons.

Guests with the following tickets may now enter:	
1 - 5	

Ticket Availability	
Total tickets outstanding:	7
Next available entry:	12:35 PM
<input type="button" value="Issue Ticket"/>	

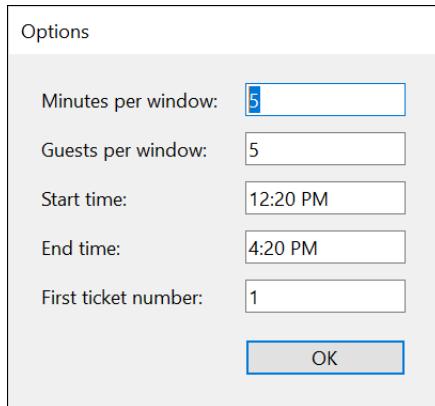
Ticket 6: 12:30 PM
Ticket 7: 12:30 PM
Ticket 8: 12:30 PM
Ticket 9: 12:30 PM
Ticket 10: 12:30 PM
Ticket 11: 12:35 PM
Ticket 12: 12:35 PM

<input type="button" value="Options"/>	<input type="button" value="Exit"/>
--	-------------------------------------

Operation

- To issue a ticket, the user clicks the Issue Ticket button. A ticket with the next ticket number in the next available time slot is issued.
- Each time a ticket is issued, it is added to the list box.
- The title bar for this form displays the current time and an indication of whether the current time slot is open or closed. This information is updated once per second.
- The labels in the Ticket Availability section of the form indicate how many tickets are outstanding and the time slot that will be assigned to the next ticket that's issued. These labels are updated as tickets are issued and the time changes.
- When a time slot begins, any outstanding tickets for that time slot are removed from the list box and the beginning and ending ticket numbers assigned to that slot are displayed in the label inside the group box at the top of the form.
- To change the options for issuing tickets, the user clicks the Options button. Before the Options dialog box is displayed, a dialog box is displayed that warns the user that all outstanding tickets will be deleted and confirms that the user wants to continue.

The Options form



Options

Minutes per window:

Guests per window:

Start time:

End time:

First ticket number:

Operation

- The user can enter values into the Options dialog box to specify the number of minutes for each time slot, the number of guests allowed into the attraction during each time slot, the time the attraction opens, the time the attraction closes, and the number for the first ticket. The defaults are five minutes per time slot, five guests per time slot, a start time of the current time, an end time of four hours after the current time, and an initial ticket number of 1.
- When the user clicks the OK button, the Tickets form is displayed.

Specifications

- Use a Timer control to display the current time in the title bar of the main form and to determine the current time slot.
- When the application starts, it should display the Options dialog box from the Load event handler for the Tickets form. For this to work without an exception being thrown, you'll need to disable the Timer control until the Options dialog box is completed.
- The data the user enters in the Options dialog box should be validated to be sure that the minutes per time slot, guests per time slot, and first ticket number are integers; that the start and end times are DateTime values; and that the difference between the start time and the end time provides for at least two time slots.
- Create a class that represents a time slot for assigning tickets. This class should have public fields that indicate the time the time slot begins, the length of the time slot, and the number of tickets that have been issued for the time slot. This class should also have a property that gets the time the time slot ends. Use this class to create a time slot object for each time slot between the start and end times entered on the Options dialog box and store these objects in a collection.
- Create a class that represents a ticket. This class should have public fields that store the ticket number and the time slot that's assigned to the ticket and a method that sets the next ticket number. As each ticket is issued, the application should create a ticket object with the next ticket number and add it to a queue. Then, when the time slot for the ticket begins, the ticket should be removed from the queue.
- Once a time slot begins and the tickets in that time slot are removed from the outstanding list of tickets, the user should not be able to issue any more tickets in that time slot even if the maximum number of tickets for that time slot haven't been issued.

Extra notes

This project uses several custom objects, including custom objects that have properties that are other custom objects. It also has extensive collection and date/time requirements. Finally, it requires that the student research and use the Timer control, which isn't presented in the book.

The complexity of the project can be varied as follows:

- Implement just the ticket assignment portion of the application, with no real-time monitoring of the time.
- Issue a ticket for the current time slot if tickets are still available.
- Instead of issuing a ticket for the next available time slot, let the user choose the time slot for the ticket. Then, the tickets in the current time slot will have to be listed individually at the top of the form. Depending on how the student designs the data store, this may result in a major redesign of the application.
- Use NumericUpDown controls to select the minutes per window and guests per window from the Options dialog box, and use the DateTimePicker control to select the start and end times. This will require some additional research by the student.

Submission Requirements via D2L:

1. **Add a Screenshot to your Project Files, called 3-2.PNG. Display your NAME as the student here.**
2. **.Zip the Project Files and submit to D2L (ZIP is the ONLY file format I will accept)**