

ECET 16400 Lab 06 Spring 2018

Lab 06 instructions:

Lab 06 Pre-Lab:

Be sure you have completed all pre-lab, reading and practice items as suggested in Lecture 05 Slides.

Assignment Instructions

- **Objective:**

Part 1. Create a Distance Calculator program that utilizes a loop based on the required reading and chapter tutorials. Read all instructions before doing Part2.

Part 2. Modify your working program such that the Distance Calculator program writes data to a file. The correct .cs file, correct .exe file, the correct zipped solution file and a design sheet word document that contains all of your UI design and completed pseudo code must be submitted to receive full credit!

If you are not sure if you have the correct files see the document “How to find your files” in the Resources Modules. It clearly shows which files you need to submit.

- **Lab 06 Program description:**

- You are going to create a project that shows the user their distance traveled based on two user inputs: Vehicle Speed (MPH) and Hours traveled (time in hours).
- See text book page 337: Programming Problems 1. Distance Calculator

- **Part 1:**

- Program Description:

- See text book page 337: Programming Problems 1. Distance Calculator.
 - Follow instructions in text and additional instructions provided here.

- Part 1:

- 1.1. Plan your program UI Design: (Use the Designer User Interface document)

- You are to come up with an appropriate user interface: This is where you should draw out your design on a sheet of paper and identify each object (name your controls). There are no specific sizes or colors of fonts. You have read enough to this point to where you know how to change all these properties. You must create something that looks “professional”.

- 1.2. Plan the Event Handler Methods: use Pseudo code, flow charts, etc..

- A Calculate button will take the user’s two inputs and display the Power calculation result in a label.
 - Variables must have appropriate data type and meaningful names
 - Only allow positive real numbers greater than 0. (double data type)
 - Exception handling must be included in this program with if-else try parse logic (advanced) or try-catch parse logic (average)
 - User should not be able to put letters or negative numbers in these input boxes and try to calculate
 - Appropriate error messages should prompt the user
 - Output must be a formatted text with result properly calculated and displayed in a label.
 - A Clear button should clear everything and set the focus back to the first input box
 - An Exit button will exit the user out of the program.

- 1.3. Write the program code:

- Create a new project. Call the project yourlastname_Lab06. Rename the Form1.cs file to DistanceCalc.cs (accept all changes if prompted). Change the text on the form to Distance Calculator.
 - Place and Set the properties of each object, as you have planned.

ECET 16400 Lab 06 Spring 2018

- **Before you proceed:** Make sure all objects (controls) are properly re-named and properties assigned as instructed.
- **Build your program to ensure all changes load correctly.**
- Working form the pseudo code, invoke and write each event-handler method.
- When you complete the code, thoroughly test the project.
 - Make several “good” and “bad” input tests to ensure your program performs correctly and handles all user input.
 - Use books’ test data
 - Output must be displayed as books’ example
 - Once your program works as desired: get three screen shots of your program when running. 1. Of the user interface when it first starts. 2. Of the program with result displayed when the user types in “good” data. And 3. Of the program with error message when the user types in “bad” data. You will include these screen shots at the end of your design sheet.
- **Part 2:** before preceding: Make sure to get screen shots of Part I as instructed
- Program Description:

Once Part 1 is working correctly you are to modify your program such that your program will also write the output to a file. This is explained on Page 338 Programming Problem 2: Distance File: Your program will follow the books description EXCEPT don’t delete the listbox....keep all the function of Part 1. Just add the code logic to write the data to a file as well.
- **Part 2 Planning:**
 - 2.1. Plan your program UI Design:
 - You are to modify your user interface: This is where you should modify your previous UI design to include and identify each new object (control).
 - Add a button to your form called Write Data to File.
 - When clicked, it will open a file, write the data, and then close the file.
 - Reference the books’ description as well (page 338)
 - 2.2. Plan the Event Handler Methods: use Pseudo code, flow charts, etc
 - When the Calculate button is clicked all of Part 1 should perform as previously instructed. When the Write Data to File button is clicked your code will write the data to a file.

Plan (write, or flow chart) your logic before you begin to code.

Identify on your design sheet where, in your outline logic, you are to put the loop logic. Flow chart or pseudo-code this logic including the output.

Do not attempt to change your UI and code in your program until you understand what you are doing and how/where to add the additional controls and code logic.
 - 2.3. Write the program code:
 - Place each new object and Set their properties as you have planned.
 - **Before you proceed:** Make sure all new objects (controls) are properly re-named and properties assigned as instructed.
 - **Build your program to ensure all changes load correctly.**
 - Working form the pseudo code, modify the appropriate event-handler method(s).
 - When you complete the code, thoroughly test the project.

Finally:

You must assign accept and cancel button assignments (page 179), Proper form tab order (page 175-176)

NOTE – COMMENT, COMMENT, COMMENT. Comment everything you program in the event handlers. Your code should have comments so anyone that looks at your program knows what it is doing and why. It doesn’t have to be a long drawn out explanation unless it’s absolutely necessary. The more concise you can make the comment the better, unless you deem it necessary to the viewer of your code to

ECET 16400 Lab 06 Spring 2018

know in extra detail why you did what you did. Do not forget your commented header: as exemplified in Lab 02.

In addition to instruction requirements, You will be graded on:

- User interface design: completeness, clarity, and ease of use. This would include labels telling the user what to do, buttons with descriptive text, easy to read, tab order (pg175-177), accept and cancel button assignments (pg179), use of proper data types for calculations and visually appealing.
- Source code: this would include code organization, no extra control methods in code, complete code comments, objects re-named.
- Catching user input errors
- Accept and cancel button assignments
- Logical form object tab order
- Project works as instructed
- Correct files submitted in Canvas assignments

Be sure to “save all” and run your program each time you make any changes to your program design and/or code. Otherwise your changes will not reflect on your submitted executable project.

- **Lab 06 Submission:**
Submit your project via the appropriate Canvas Assignments tool. You will **submit three files**: (the .exe executable file, the Visual C# .cs file and a zipped file of your entire project.) **and your Designer UI design document** that contains all of your completed pseudo code and screen shots as well. If you have any problems locating these files, see “How to find your files” document in Resources Module.

Note the closing date and time; as late submissions will not be accepted, nor any files sent as messages or attachments in email.