# CS 1400 Lab 4: Creating and using a MenuStrip

## Objectives:

The objective of this lab is to use Visual Studio to create a Graphical User Interface program that contains a MenuStrip.

## The Interface

Every GUI project that you build should include a MenuStrip with a menu selection to Exit the application, and a menu selection to show an About Box. This lab will demonstrate how to create such an application. The About Box displays your name and project information. This simple interface is shown in figure 1 below.
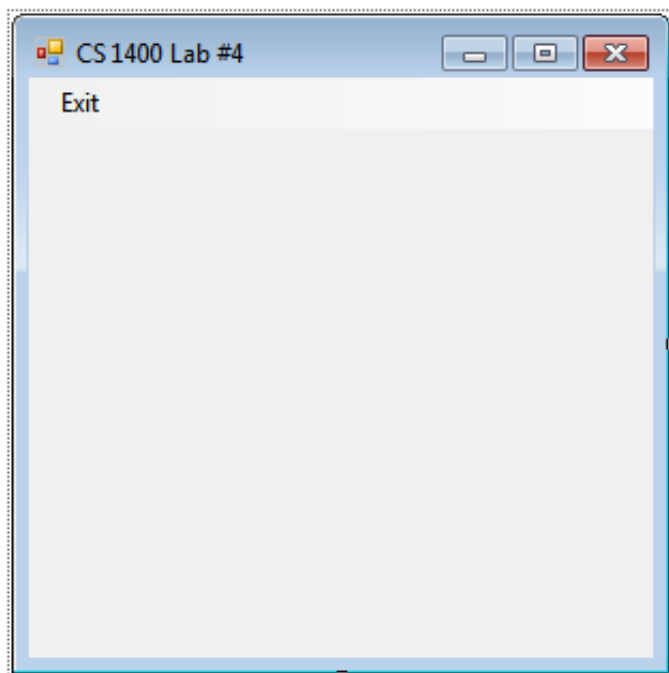


Figure 1: The Graphical User Interface

## Creating the Interface

1.  Create a folder for this lab assignment.
2.  Start Visual Studio and create a new Windows Forms Application.
3.  Name your Project Lab04.
4.  From the the Toolbar, drag a MenuStrip onto the Client Area of your Form.

5. In the Text area on the Menu Strip type `Exit`
6. In the Text area below this, type `Exit` again.
7. In the Text area below `Exit` type `About`

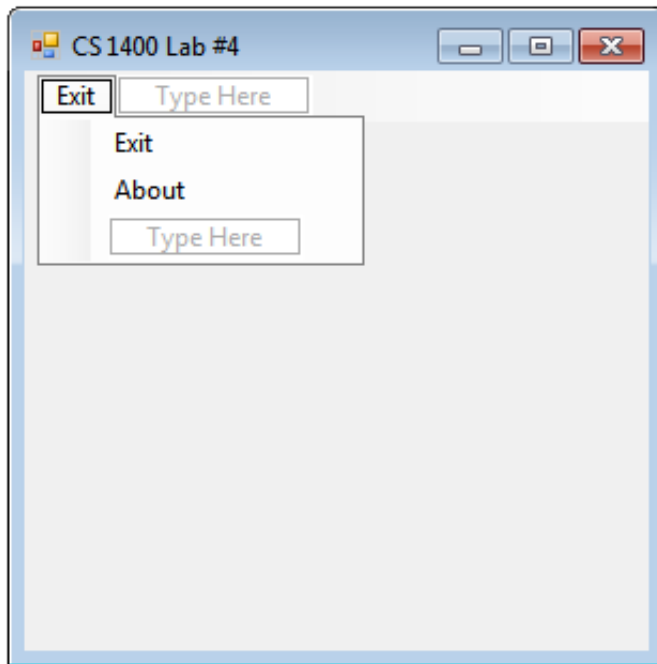At this stage your form should look about like Figure 2.



Figure 2: The Completed MenuStrip

## Creating Event Handlers

1. Double click on the Menu Item `Exit` (the second `Exit` down, not the one on the MenuStrip itself).
2. This will open up the code editor window and create a **method** to handle the event that gets generated when a user clicks on `Exit`.
3. Add the line of code to close the Window, as shown below. The Form's Close( ) method closes the Window. When the Window closes, the application terminates.

```
void exitToolStripMenuItem1_Click(object
sender, EventArgs e)
{
    this.Close();
}
```

4. Double click on the Menu item  `About`. This will add a method to handle the event that will get generated when a user clicks on `About`.
5. Add the line of code to show a MessageBox as shown below. The MessageBox.Show( ) method takes a string argument that is displayed in the box. Of course, you will supply your own name.

```
void aboutToolStripMenuItem_Click(object
sender, EventArgs e)
{
    MessageBox.Show("Roger deBry\nCS1400\nLab
#4");
}
```
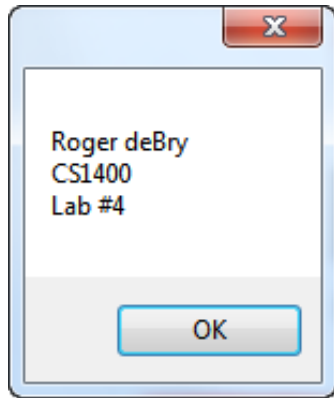
My About box is shown in figure 3.



Figure 3: A MessageBox

Now build and test your code.

## Documenting your Code

Before you turn in your lab be sure that you have written a complete file prologue in your source code file (usually this is Form1.cs), and that you have written method prologues for each method you have added to your Form class. For example, the complete documented method to handle the Exit Menu event should look something like this:

```
// The exitToolStripMenuItem1 method
// Purpose: To close the window and terminate the
application
// Parameters: The object generating the event
// and the event arguments
// Returns: None
void exitToolStripMenuItem1_Click(object sender,
EventArgs e)
{
    this.Close();
}
```

## Submitting Your Assignment

Place your complete project folder into a zip file and name the zip file lab_04_your-initials_V1.0.zip. For example, I would name my file lab_04_RKD_V1.0.zip. Submit this assignment as Lab #4 on Canvas.

## Grading Guidelines

| Description | Points possible |
|---|---|
| Assignment meets grading guidelines:<br>o Source code files contain a declaration that you did not copy any code, except that provided.<br>o Assignment has been properly submitted to Canvas<br>o Code meets style guidelines | 2 |
| Your user interface looks like the example, and the Exit and About menu items work as directed. | 3 |
| Total | 5 |