CISP 41 -- Homework 3

Due Thursday, 11/29/2012

Note: Staple pages and arrange your work in this order.

Review Questions from Book:

- Chapter 9
  - 1, 2, 3, 6, 7, 8, and 11

1. Explain the differences between the execution of a Windows application

and a Web application.

A Web application runs in a browser whereas most Windows applications

run stand-alone.

2. Differentiate between the client and the server for a Web application.

Client is the local computer that runs the Web page in a browser

The server is a local or remote computer that stores the

Web page files and renders the page/content for the client.

3. What is meant by the statement that Web pages are stateless?

The pages never stores information about its contents. Never.

6. How does event handling differ from that for Windows applications?

Coding for C# events is the same as coding for Web controls

7. Describe at least two methods for controlling the layout of controls on a

Web page.

HTML tables have rows and columns that can be aligned

Absolute positioning uses X & Y coordinates

Cascading style sheets (CSS) creates styles within and between many pages

CSS uses a precedence of style rules to specify layout

8. What functions are done by validator controls? How can you set up a

validator control?

Functions include required field that makes sure it is filled out

Range to make sure the data is within right range

Compare to make sure it matches to another control or constant

Add a validator control and attach it to the input control

Code the error message to be displayed

11. What is the purpose of XML? of SOAP?

XML Extensible Markup Language

Tag-based notation that defines data and its format and allows its transmission over the Web.

SOAP Simple Object Access Protocol

An XML-based protocol. Allows exchange of component information among

distributed systems of many different types.

- Chapter 10
  - 1, 2, 3, 7, 9, and 11

1. Explain the purpose of a binding source.

It establishes a link to the actual data, such as the specific file and/or server

2. Explain the purpose of the table adapter component.

It handles the retrieving and updating of the data.

It also generates SQL statements that are used to retrieve or update the data

3. What is a dataset?

Has the actual data.

May include multiple binding sources and/or multiple table adapters.

7. What is the purpose of the Data Sources window?

It adds a new data source to the project. You can select a file and table.

9. What is a parameterized query? When would it be used?

A way to use a parameter to select the correct data for individual fields.

It ensures that values are obtained during runtime based on their properties.

They are required with a Web selection program to retrieve the

data matching the list selection.

11. What is the purpose of LINQ? Name three operators.

LINQ is a query language that works on object data types,

It can ask a question of any data

The primary operators in a query are *from, in, where* and *select*

Operator	Purpose
From	Name of a single element.
In	Specifies the source of the data (all of the elements to query).
Where	A Boolean expression that specifies the condition for the query.
Select	Execute the query. The identifier determines the type of data element(s) that will be returned from the query.

- Chapter 11
  - 1, 2, 4, 5, 6, and 7

1. Explain what occurs when a stream object is instantiated.

First of all, instantiation is the act of creating an instance of a class.

A *stream* is designed to transfer a series of bytes from one location to another

So when a stream object is instantiated, data gets transferred from the buffer to the file

2. Name two types of stream classes. What is the difference between

the two?

StreamReader and StreamWriter classes.

Write the StreamWriter code first, to create the data file.

Then write the StreamReader code to read the file that you just created.

StreamWriter works with original file, StreamReader works with newly created file

4. What steps are necessary for storing the list items from a list box into a

disk file?

If the list items are stored in a data file, read the file into

the list using the Form_Load method. Loop through the file until all elements are

read, placing each item in the list by using the Items.Addmethod.

Check to make sure the file exists.

Open a StreamWriter object then loop through the Items collection of the list box,

and save each element by using a WriteLinemethod.

Can also prompt user to save when form closes.

5. What is the format for the statements to read and write streams?

Declare an object of the class

Use the read/write method for the data (can use loops here)

Close the stream using the Close method

6. What method can be used to determine the end of file?

The StreamReader's *Peek method*

It looks at the next element without reading it

If you've peeked beyond the last element, the value returned is (-1)

7. When is exception handling necessary for stream handling?

To catch the exception for a missing file.

Can then display a message box and ask the user if he wants to create a new file.

- Chapter 12
  - 1, 2, 3, 6, 9, 10, and 12

1. What is an object? a property? a method?

An object has properties, events, and methods.

An object is a combination of variables and functions that work as one unit.

The key word is that an object is a unit.

Properties are the adjectives, they are the descriptions of the object.

So an object is a unit, and the property describes it.

A method is a unit of code that performs a task

2. What is the purpose of a class?

A class has properties, methods, and events.

A class allows creation of new object types.

A class saves time and reduces coding because all the objects are

sub levels of the class

3. Why should property variables of a class be declared as private?

To not violate the rules of encapsulation.

Encapsulation requires each object to be in

charge of its own data.

Private declaration allows data hiding.

6. What actions trigger the constructor and destructor methods of an object?

A constructor is a method that executes automatically when an object is

Instantiated, when you create an instance (an object) of the class

A destructor is a method that executes automatically when an

object is destroyed, when an object goes out of scope.

This is handled automatically by CLR

9. Explain the steps necessary to inherit a class from another class.

Change the Private Members to Protected

Add the New Class

Add the Constructor

Add a Constant

Override a Method

Add Required Accessors

Allow a Method to Be Inherited

Modify the Form to Use the Inherited Class

10. Differentiate between overriding and overloading.

Overloading means that two methods have the same

name but a different list of arguments (the signature).

Can create over-loaded methods in your class by

using the same name for multiple methods, but

each has a different argument list.

Overriding refers to a method that has the same signature

(name and parameter list) as a method in its base class.

The method in the subclass, aka derived class, overrides (takes precedence over)

the identically named method in the base class.

12. When might you use the protected keyword on a method?

Protected variables and methods behave as private but are available in classes that inherit from the class

To ensure that variables/methods are only available within the class, or classes that have inherited from the class.

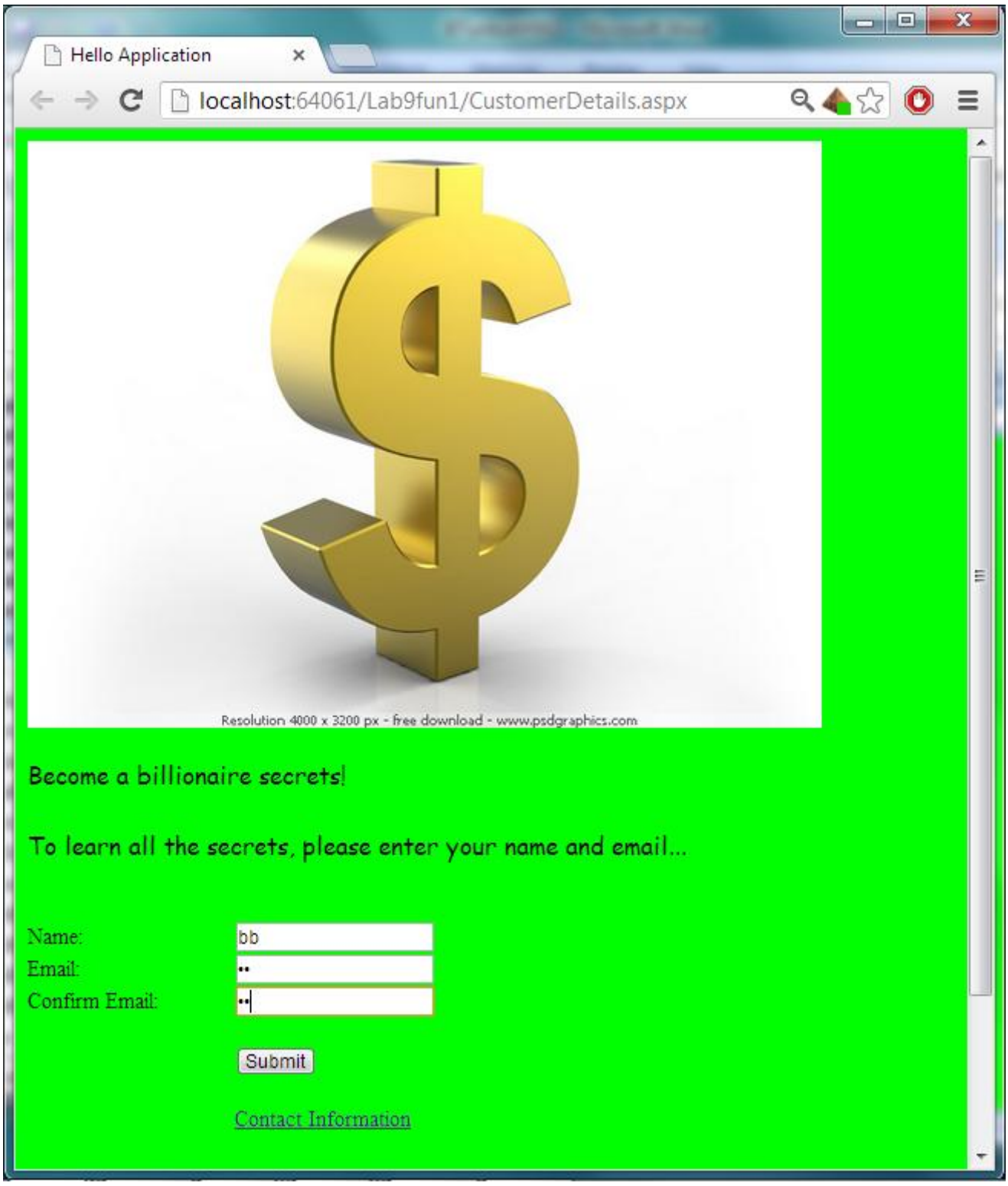
Programming Exercises from Book: Pick 2 out of 4, but at least 1 must be 11.4 or 12.1

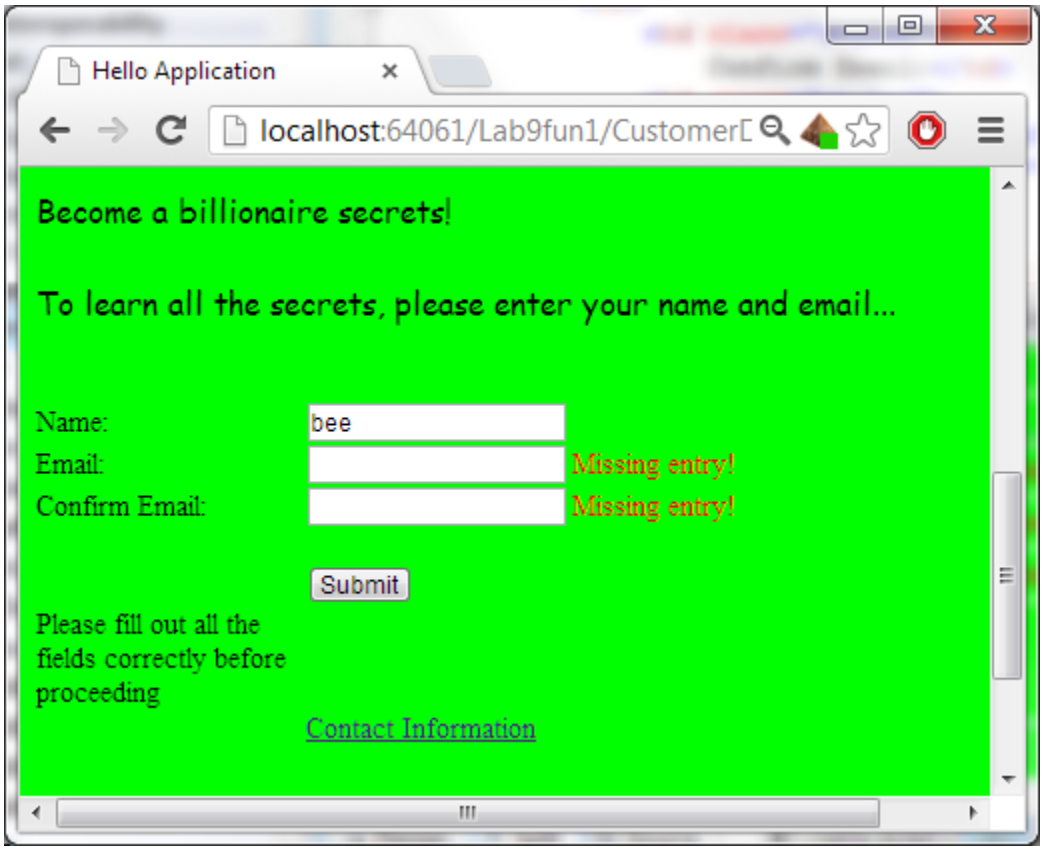
- Chapter 9
  - 9.4

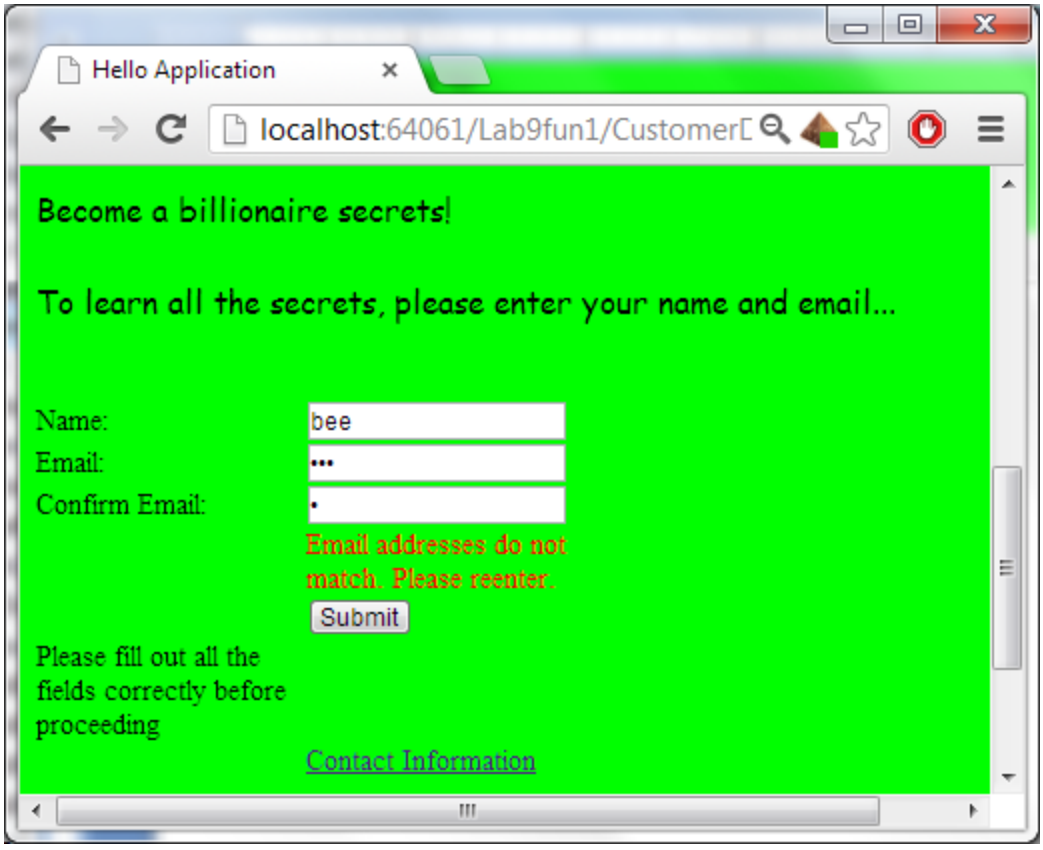
9.4 Create a Web page for a company of your choosing. Include multiple

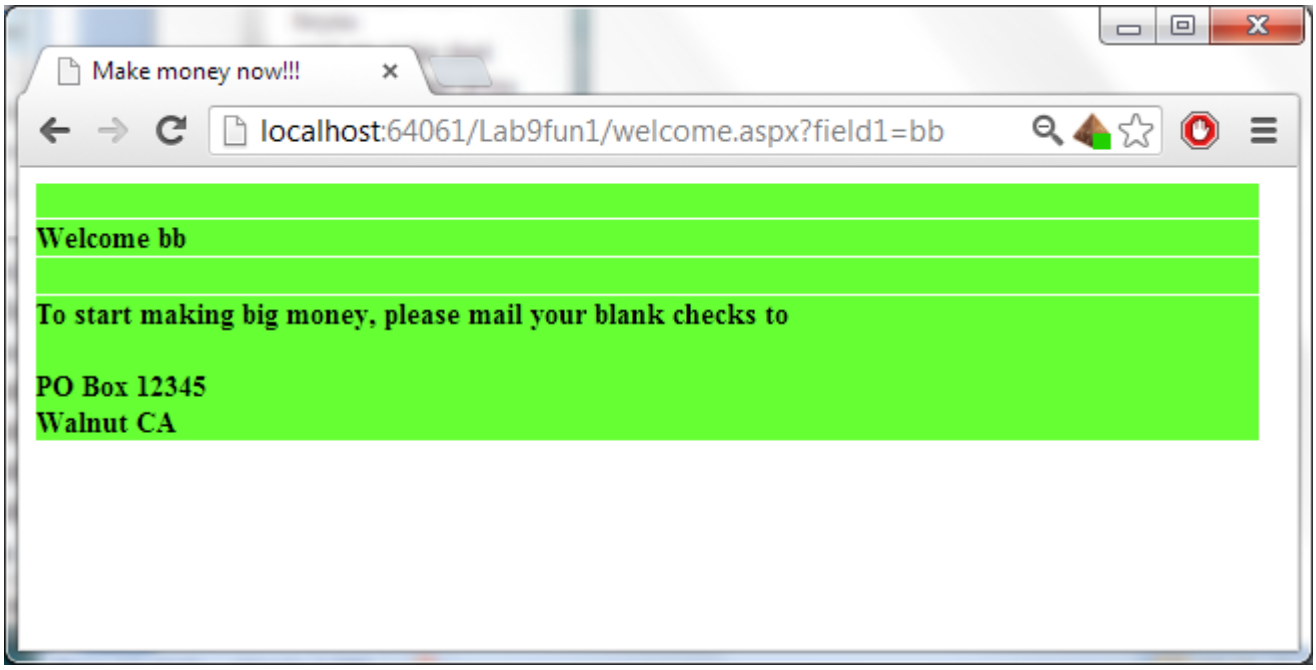
pages, a HyperLink control, and validator controls.

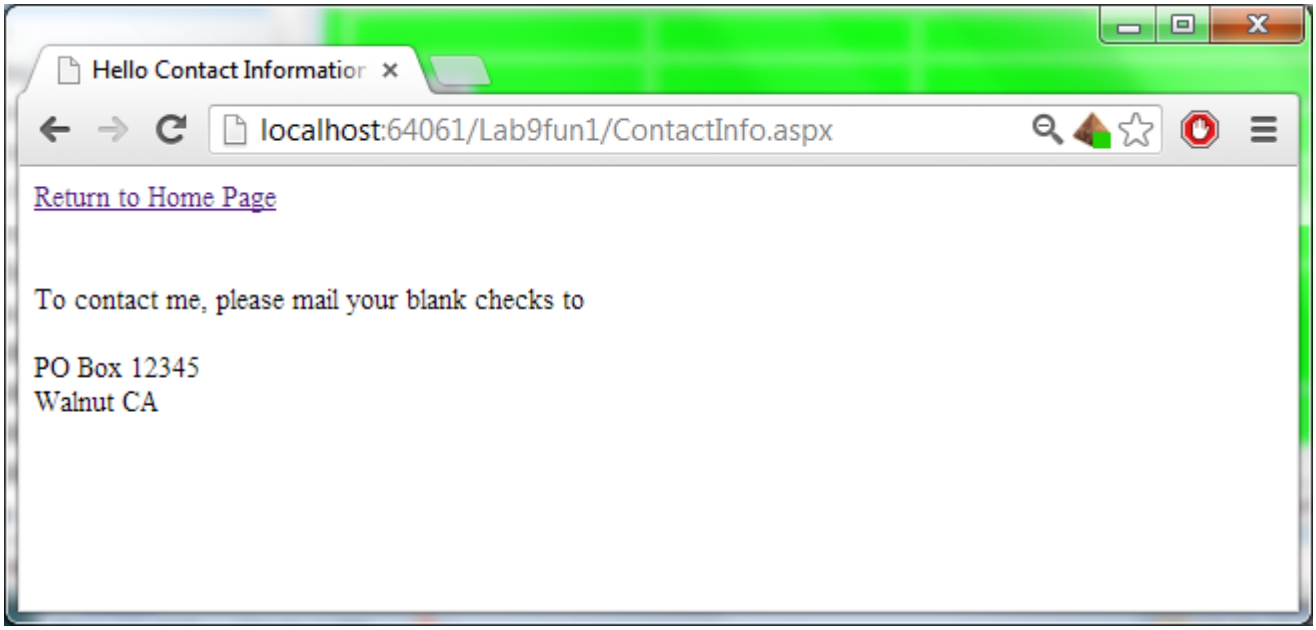
Lab 9.4 by Bryant Tunbutr




Hello Application
localhost:64061/Lab9fun1/CustomerD
Become a billionaire secrets!
To learn all the secrets, please enter your name and email...
Name:
bee
Email:
Missing entry!
Confirm Email:
Missing entry!
Submit
Please fill out all the fields correctly before proceeding
Contact Information


Hello Application
localhost:64061/Lab9fun1/CustomerD
Become a billionaire secrets!
To learn all the secrets, please enter your name and email...
Name:
bee
Email:
Confirm Email:
Email addresses do not match. Please reenter.
Submit
Please fill out all the fields correctly before proceeding
Contact Information


Make money now!!!
localhost:64061/Lab9fun1/welcome.aspx?field1=bb
Welcome bb
To start making big money, please mail your blank checks to
PO Box 12345
Walnut CA


Hello Contact Information
localhost:64061/Lab9fun1/ContactInfo.aspx
Return to Home Page
To contact me, please mail your blank checks to
PO Box 12345
Walnut CA

Source code for BTunbutrLab9.cs

CustomerDetails.aspx.cs

```
/*
 * Project:                 BTunbutrLab9.4
 * Programmer:       Bradley/Millspaugh
 * Date:               Nov 29 2012
 * Description:     Create a website that validates information
 * and creates a welcome screen
 * Have multiple pages, validation, and a hyperlink control
 * I certify that the code below is my own work.
 */
using System;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void submitButton_Click(object sender, EventArgs e)
    {
        //Make sure text fields are filled.
        //If so display validator message
        if (nameTextBox.Text != "" && emailTextBox.Text != ""
            && confirmTextBox.Text != "")
        {
            //Check if the emails match
            if (String.Compare(emailTextBox.Text, confirmTextBox.Text) == 0)
            {
                //if so display welcome
                StatusLabel.Text = "Welcome " + nameTextBox.Text;
                Response.Redirect("welcome.aspx?field1=" + nameTextBox.Text);
            }
            else
            {
                //If different make user enter again and clear boxes
                confirmTextBox.Text = "";
                emailTextBox.Text = "";

                StatusLabel.Text = "Email addresses do not match! Enter email
addressesagain";
            }
        }
        else
```

```
            StatusLabel.Text = "Please fill out all the fields correctly before
proceeding";
    }

}
```

welcome. aspx.cs

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class Default2 : System.Web.UI.Page
{
    String name;
    protected void Page_Load(object sender, EventArgs e)
    {
        name = Request.QueryString["field1"];
        Label1.Text = "Welcome " + name;
    }
}
```

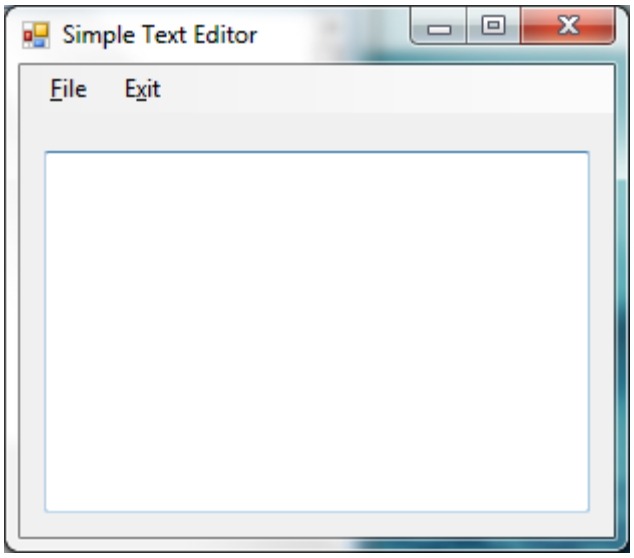
- Chapter 11
  - 11.4

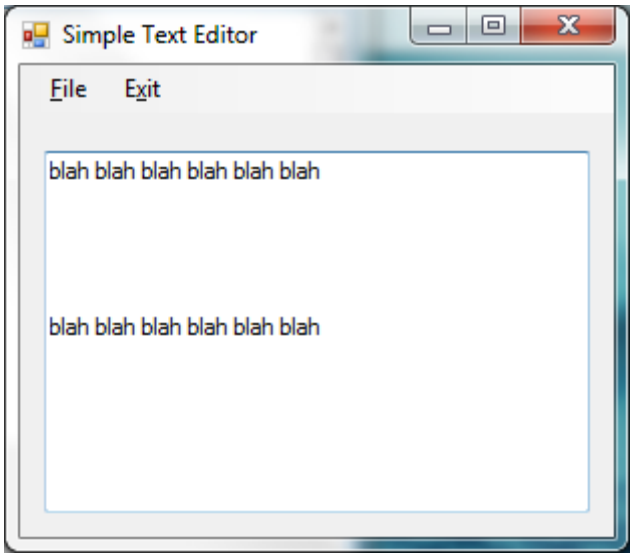
11.4 Create a simple text editor that has one large rich text box (with its Multiline property set to true). Set the text control to fill the form and set its Anchor property to all four edges so that the control fills the form even when it is resized.

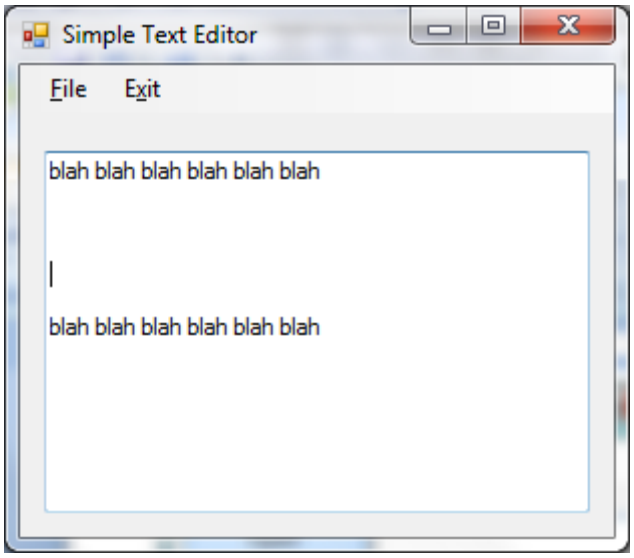
Allow the user to save the contents of the text box in a data file and load a data file into the text box using theOpen Filedialog box.

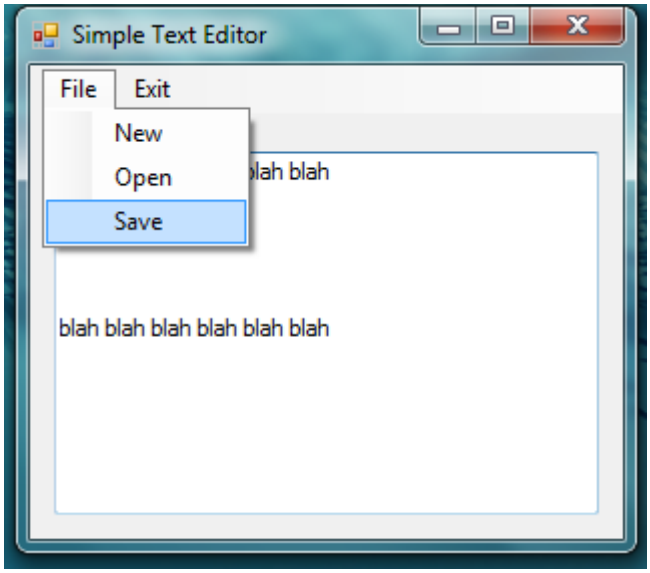
Use a StreamWriter and StreamReader or File.WriteAllTextand File.ReadAllText.

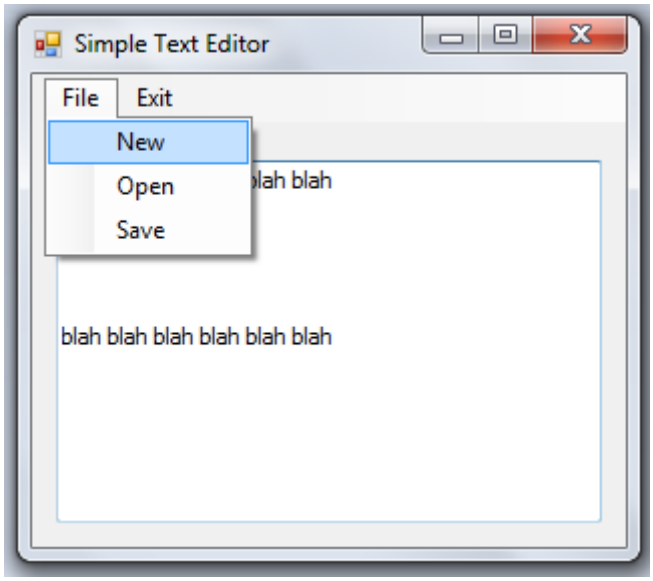
Lab 11.4 by Bryant Tunbutr

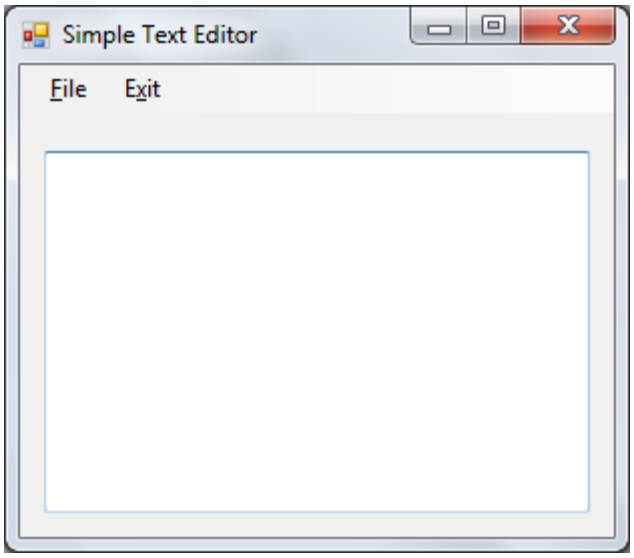


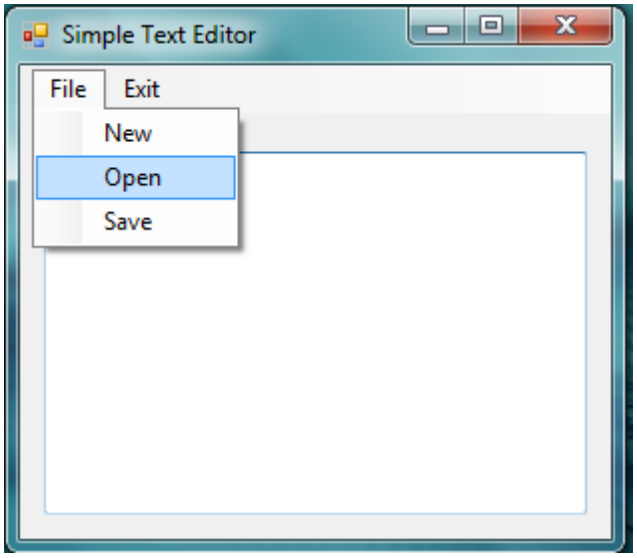


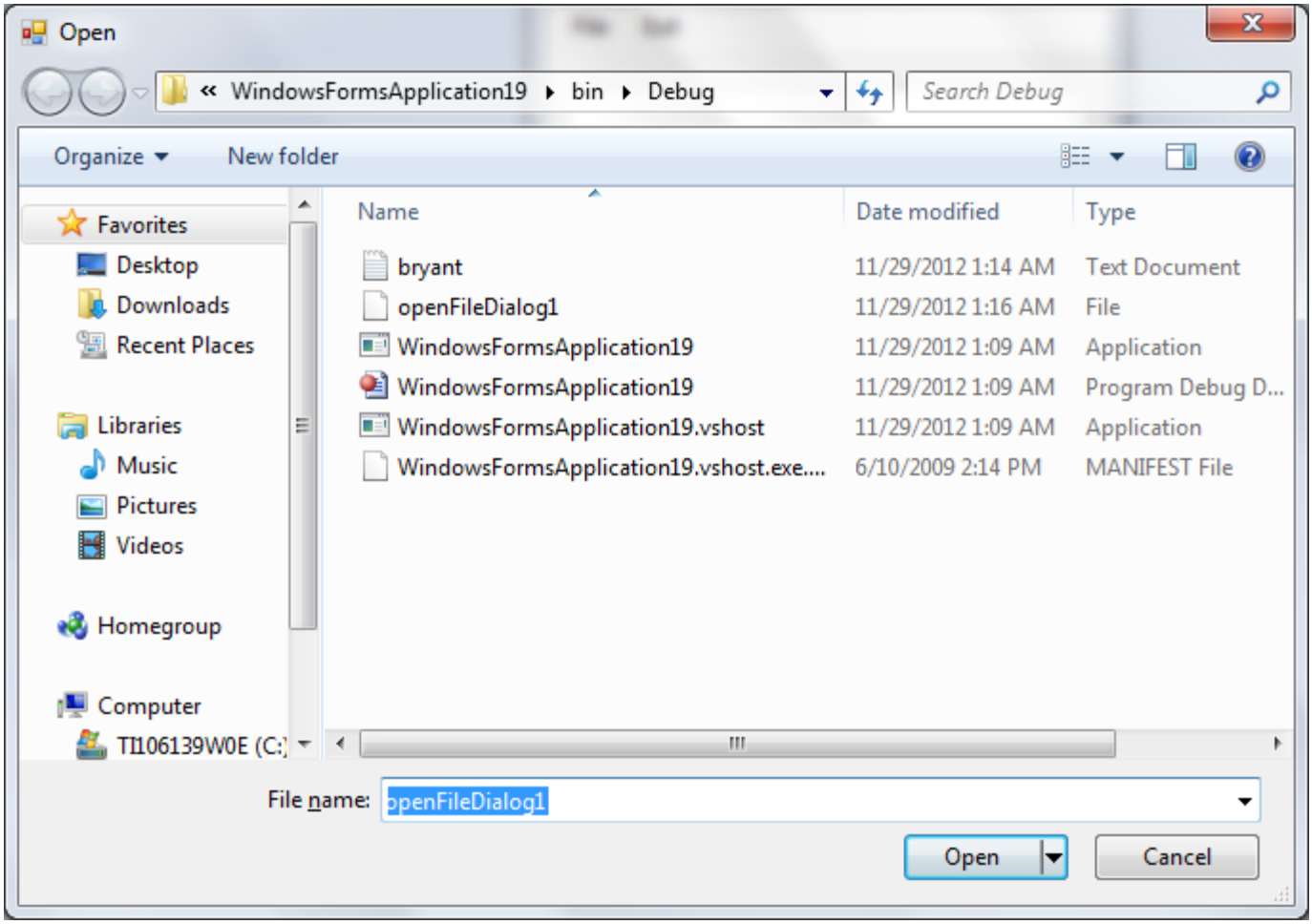

Simple Text Editor
File
Exit
blah blah blah blah blah blah
blah blah blah blah blah blah

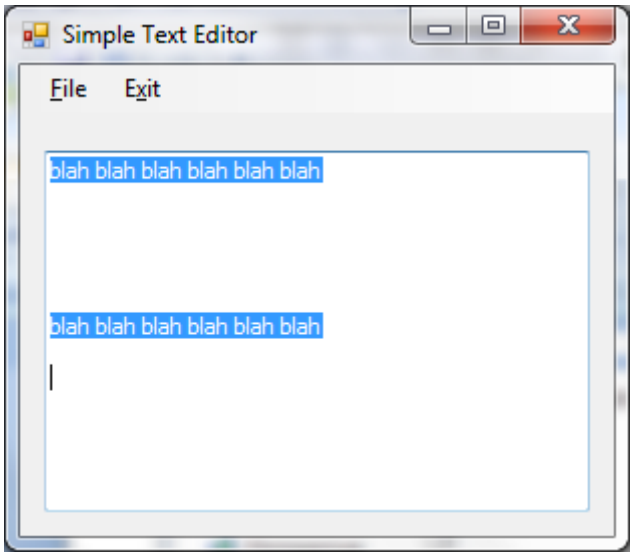

Simple Text Editor
File
Exit
New
Open
Save
blah blah
blah blah blah blah blah blah


Simple Text Editor
File
Exit
New
Open
Save
blah blah
blah blah blah blah blah blah


Simple Text Editor
File
Exit


Simple Text Editor
File
Exit
New
Open
Save


Open
« WindowsFormsApplication19 ▸ bin ▸ Debug
Search Debug
Organize
New folder
Favorites
Desktop
Downloads
Recent Places
Libraries
Music
Pictures
Videos
Homegroup
Computer
TI106139W0E (C:)
Name
Date modified
Type
bryant
11/29/2012 1:14 AM
Text Document
openFileDialog1
11/29/2012 1:16 AM
File
WindowsFormsApplication19
11/29/2012 1:09 AM
Application
WindowsFormsApplication19
11/29/2012 1:09 AM
Program Debug D...
WindowsFormsApplication19.vshost
11/29/2012 1:09 AM
Application
WindowsFormsApplication19.vshost.exe....
6/10/2009 2:14 PM
MANIFEST File
File name:
openFileDialog1
Open
Cancel


Simple Text Editor
File
Exit
blah blah blah blah blah blah
blah blah blah blah blah blah

Source code for BTunbutrLab11.4.cs

BTunbutrLab11.4.cs

```
/*
 * Project:                BTunbutrLab11.4
 * Programmer:       Bradley/Millspaugh
 * Date:               Nov 29 2012
 * Description:     Create a website that validates information
 * and creates a welcome screen
 * Have multiple pages, validation, and a hyperlink control
 * I certify that the code below is my own work.
 */

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO;

namespace WindowsFormsApplication19
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void openToolStripMenuItem_Click(object sender, EventArgs e)
        {
            //Shows the openFileDialog
            openFileDialog1.ShowDialog();

            //Reads the text file
            System.IO.StreamReader OpenFile = new
System.IO.StreamReader(openFileDialog1.FileName);

            //Displays the text file in the textBox
            textBox.Text = OpenFile.ReadToEnd();

            //Closes the proccess
            OpenFile.Close();
        }

        private void saveToolStripMenuItem_Click(object sender, EventArgs e)
        {
            //Determines the text file to save to
            System.IO.StreamWriter SaveFile = new
System.IO.StreamWriter(openFileDialog1.FileName);

            //Writes the text to the file
```

```
            SaveFile.WriteLine(textBox.Text);

            //Closes the process
            SaveFile.Close();
        }

        private void addToolStripMenuItem_Click(object sender, EventArgs e)
        {

        }

        private void exitToolStripMenuItem_Click(object sender, EventArgs e)
        {
            // End the project.
            this.Close();
        }

        private void newToolStripMenuItem_Click(object sender, EventArgs e)
        {
            textBox.Clear();
        }
    }
}
```