SYSTEMS DESIGN / CAPSTONE PROJECT
MIS 316

Project 1

Goals: Demonstrate the capability to build a new project and demonstrate the use of variables, basic math functions and conversions of strings to numbers, and formatting.

1. Using the 1.0 Starting a New Project Guide found on the class website, start a new project on your DESKTOP, name the folder your initialsProject1 and name the project name your initialsProject1 as in tnjProject1. **Please make sure the *Location* is on your desktop.**
    a. Also before hitting the OK button, check that in the left column you are under C# - Web and then Previous Versions, the middle column should be the second entry or ASP.Net Web Forms Site.

2. One you hit OK, it will build a project for you, with many files. On the right side of your screen should be the *Solution Explorer* which lists all the files in your project. Over time, we will learn more about the project. We will do all our work on the About.ASPX page for this project.

3. Click the About.aspx file in the right Solution Explorer window.

4. It will open the About.aspx page in your center window. Near the bottom, click the SPLIT tab to see both the coding and the actual web view (design) page.

5. Putting your name on the page:
    a. In Line 1 you should see a Title="" tag, place your name in the title tag and Project 1 as in Title="My Name – Project 1"
    b. Modify Line 4 to be your name between the <h2> and </h2> tags, your line should be <h2>**Your Name**</h2>
    c. Modify line 5, to say: **Project 1**between the <h3> and </h3> tags
    d. Revise line 6, between the and tags to say: Demonstrate Basic C# Features.
    e. Before Line 7 (the </asp:Content> tag), hit enter 5 times as the /asp.Content tag must be the last command on the page. All your new work will go after the tag and before the </asp:Content> tags
    f. Save your work, click the Green Arrow and select Firefox, and bring up the web page you just built. Does it contain all the features you just built?

6. Creating new styles for your site:
    a. In the CONTENT folder, click on the site.css file
    b. After the last } add a comment so you know where you created your own styles as in:

```
/* Styles created by Tom J*/
```

c. After your comment add in your first style to bold items and center items

```
.center {
    text-align: center;
}
.bold {
    Font-weight: bold;
}
.failureNotification {
    color: red;
    font-weight: bold;
}
```

7. Save the file

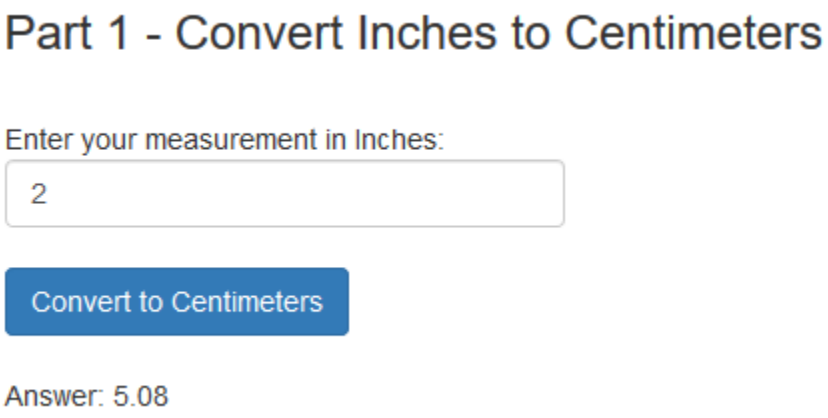
8. Back on your About.aspx page, and after the </p>; we desire to implement the bootstrap 'input form' object so it scales properly on a smart device as well as a desktop. Bootstrap input forms are like a table that it has rows and columns. Type the following after the </p> you placed after the _instruction label. We are saying start a new row on the page and then add an item that is 6 columns wide (our text boxes etc) but first you leave 3 blank columns so that it is better centered on a monitor (that is the offset style).

```
<div class="row">
   <div class="col-md-6 col-md-offset-3 column">

          'rest of input boxes go here


   </div>
</div>
```

9. Our goal is to get an information (larger label), one label, one text box, one error message, a button and an answer label for each of our tasks, each should look like:



10. The code to accomplish the above is:

a. Start a from group div (division) by typing
   i. (it will create a tag, put all the items in 10/11/12 before this tag

ii. Add an h3 Tag to say what this series of labels, text boxes and buttons will accomplish. For the first example it would be: <h3>Part 1- Convert Inches to Centimeters</h3>
iii. Insert a label, change the text property to: Enter your measurement in Inches:
iv. Give the label the Style (CssClass) of bold
v. Next add the text box to the form group, name the text box: __1Inches, add in a CssClass of form-control
vi. Finally, between the label and the text box, slide in a Required Field Validator, you set three properties here:
    1. CssClass : failureNotification (will turn it red)
    2. Control to Validate : _1Inches
    3. ErrorMessage: *Please enter Inches
    4. Set the Validation Group to: _1

11. Now add in the button, under the last
 Slide a Button to the web page. Change the following properties:
    a. Name the button - _1Calculate
    b. Text to Convert To Centimeters
    c. Set the CssClass to be btn btn-primary (this will make it blue and nicer)
    d. Change the validation group to _1

12. Insert two line breaks

13. Now insert two labels, one label should have a text property of Answer:
    a. The second label should have a name of _1Answer, and a text property of blank or “”.

14. When complete your code should be:

```
<div class="form-group">
            <h3>Part 1 - Convert Inches to Centimeters</h3>

        <asp:Label ID="Label7" runat="server" Text="Enter your
measurement in Inches:" CssClass="bold"></asp:Label>
           <asp:RequiredFieldValidator ID="RequiredFieldValidator1"
runat="server" ErrorMessage="*Please enter Inches"
CssClass="failureNotification"  ValidationGroup="_1"
            ControlToValidate="_1Inches"></asp:RequiredFieldValidator>
         <asp:TextBox ID="_1Inches" runat="server" CssClass="form-
control"></asp:TextBox>
            <br />
            <asp:Button ID="_1Calculate" runat="server" Text="Convert
to Centimeters" CssClass="btn btn-primary" ValidationGroup="_1"
OnClick="_1Calculate_Click" />
            <br />
            <br />
            <asp:Label ID="Label1" runat="server" Text="Answer:
"></asp:Label>
```

```
            <asp:Label ID="_1Answer" runat="server"
Text=""></asp:Label>
    </div>
```

15. Save and run your project to see if you get the label and text box to appear; do not enter anything into the text box and click the button and you should see an error message
16. Now time for the calculation, double click on the "Convert" button, the C# window should appear and the cursor should be in block of code that is headed:
    protected void _1Calculate_Click.

17. Let's build some good coding habits by following a systematic method in coding

    // declare variables
    double dblInchToCentimeter = 2.54;
    double dblCentimeters;
    double dnlInches;

    //grab the values from the text boxes and convert if necessary
    dblInches = Convert.ToDouble(this._1Inches.Text);

    //calculate
    dblCentimeters = dblInches * dblInchToCentimeter;

    //output and format the anwer
    this._1Answer.Text = dblCentimeters.ToString("F2")

18. Save and run; test your answers. At this point we have not verified that htye have entered a #, but we will in future projects.

Task 2:

19. If all is correct, on the .aspx page, copy from the <form-group> down to and including the first and paste before the last two to start your second task.

20. The second task will be to calculate a 4% raise to individual's hourly pay.

21. In the new group you just pasted, change the h3 tag to be: Part 2 – Calculate a 4% Raise

22. Change:
    a. The text of the label to be: Enter Current Hourly Rate
    b. The name of the text box to be _2Pay
    c. For the Required field validator, change the Error Message, the Control to Validate, and the ValidationGroup to _2
    d. For the button, change its name to be _2Calculate, change the text to be "Calculate Raise", change the ValidationGroup to _2, and delete the OnClick =….. code
    e. The last label should be named _2Answer.

23. Double click the Calculate Raise button to start your coding, follow the 4 steps described in #15 above;
    a. //declare variables (I believe 3 are needed)
    b. //grab values (1 'grab value')
    c. //calculate
    d. //display the result in _2Answer as currency.

Task 3:

24. Again repeat the copy process you did in step #17 and rename appropriate items

25. Task 3 will have the user enter the # of minutes they worked, you should display back the answer in Hours and Minutes, as if they enter 155 minutes, your _3Answer label should display back: You worked 2 hours and 35 minutes.

26. Follow the 4 steps to declare, grab, calculate and display as part of your C# code

Task 4:

27. Repeat the copy step and rename appropriate items

28. Task 4 – Will have you calculate the cost of hiring a moving firm.

29. You will need to provide two text boxes for this task, and therefore the label and required field validator for the next text boxes
    a. The two text boxes will be _4Hours and _4Miles

30. The calculation should has three components:
    a. All moves have a $200 base fee charged
    b. The hourly charge is $150/hour
    c. The mileage fee is $2/mile

31. An example would be for 10 hours and 50 miles, the total charge should be $1800.

Grading

Items graded include:
    a. Title Tag
    b. All Text Boxes Named Properly
    c. All Buttons Named Properly
    d. All Labels (as needed) Named Properly
    e. All Calculations follow the 4 step model described
    f. Variable Names appropriate
    g. It works!

Submitting your project:

32. When your project is ready for grading:
    a. Make sure Visual Studio is closed
    b. Using a file folder, move to \\miscapstone\MIS316
    c. Then Select your section (001 is 9:30, 002 is 11:00)
    d. Find your F19 folder
    e. Then slide your entire project folder from the desktop to your F19 folder area.

33. Following is a sample completed screen showing the calculations: