Lab 2-1 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.

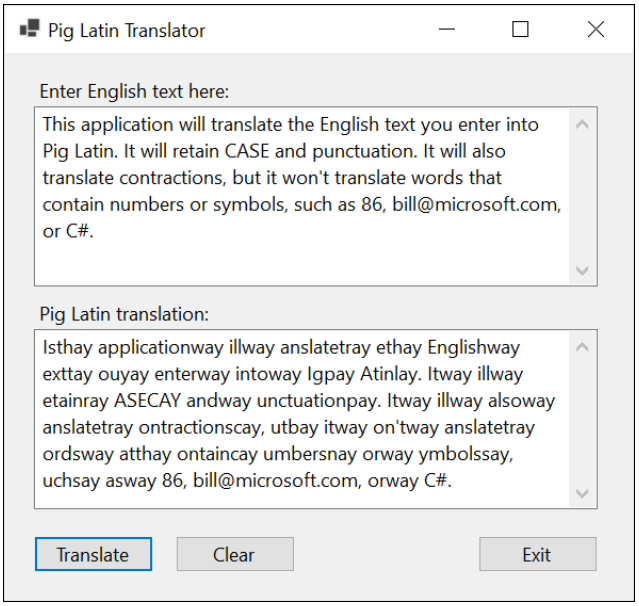
Reference Lectures:
PowerPoint: Ch 4 to 9 (8 & 9 important)
Video Lecture Part 3 Working with data Types and Strings (YouTube Video)

Due Dates: **Thursday, Jan 28 @ 6:00 pm**

Translate English to Pig Latin

For this project, you'll create a form that accepts a text entry from the user and converts the text to Pig Latin. **Prerequisites: chapters 1-9**.
In 3-6 this will be expanded.

The Pig Latin Translator form



Operation

- The user enters text in the first multi-line text box and clicks the Translate button or presses the Enter key.
- The application translates the text to Pig Latin and displays it in the second multi-line text box.
- To clear both text boxes, the user clicks the Clear button.

Specifications

- If a word starts with a vowel, just add *way* to the end of the word.
- If a word starts with a consonant, move the consonants before the first vowel to the end of the word and add *ay*.
- If a word starts with the letter Y, the Y should be treated as a consonant. If the Y appears anywhere else in the word, it should be treated as a vowel.
- Keep the case of the original word whether it's uppercase (TEST), title case (Test), or lowercase (test).
- Keep all punctuation at the end of the translated word.
- Translate words with contractions. For example, *can't* should be *an'tcay*.
- Don't translate words that contain numbers or symbols. For example, 123 should be left as 123, and bill@microsoft.com should be left as bill@microsoft.com.
- Check that the user has entered text before performing the translation.

Hints

- To create a multi-line text box, just set the Multiline property of the text box to True and size the text box accordingly.
- To add vertical scroll bars to the text box, set the **ScrollBars** property to **Vertical**.
- The integer values for all ***uppercase*** letters are 65 to 90, and the integer values for all ***lowercase*** letters are 97 to 122.

Methods you should have from the Buttons:

```
btnTranslate_Click(object sender, EventArgs e)
btnClear_Click(object sender, EventArgs e)
btnExit_Click(object sender, EventArgs e)
```

Methods you should make:

```
private string TranslateWordWithCaps(string word)
private string TranslateWord(string word)
private bool IsUpper(string word)
private bool IsLower(string word)
private bool IsInitialCap(string word)
private bool IsUpper(char c)
private bool IsLower(char c)
private string ToInitialCap(string word)
```

Starting Code:

```
1 reference
private string TranslateWordWithCaps(string word)
{
    // remove punctuation from the end of the word and store for later
    string punct = "";
    if ( word.EndsWith(".") || word.EndsWith(",") || word.EndsWith(";") ||
         word.EndsWith(":") || word.EndsWith("!") || word.EndsWith("?") )
    {
        punct = word.Substring(word.Length - 1);
        word = word.Remove(word.Length - 1, 1);
    }

    // translate word and apply correct capitalization
    if (IsInitialCap(word))
        word = ToInitialCap(TranslateWord(word));
    if (IsUpper(word))
        word = TranslateWord(word).ToUpper();
    if (IsLower(word))
        word = TranslateWord(word).ToLower();

    // restore punctuation to the end of the word
    word += punct;

    return word;
}
```

```
3 references
private string TranslateWord(string word)
{
    char c = word[0];
    if (c == 'a' || c == 'e' || c == 'i' ||
        c == 'o' || c == 'u' ||
        c == 'A' || c == 'E' || c == 'I' ||
        c == 'O' || c == 'U')
    {
        word += "way";
    }
    else
    {
        if (c == 'y' || c == 'Y')
        {
            word = word.Remove(0, 1);
            word += c.ToString();
            c = word[0];
        }
        while (c != 'a' && c != 'e' && c != 'i' &&
            c != 'o' && c != 'u' &&
            c != 'A' && c != 'E' && c != 'I' &&
            c != 'O' && c != 'U' &&
            c != 'Y' && c != 'y')
        {
            word = word.Remove(0, 1);
            word += c.ToString();
            c = word[0];
        }
        word += "ay";
    }
    return word;
}
```

```
1 reference
private bool IsUpper(string word)
{
    for (int i = 0; i < word.Length; i++)
        if (IsUpper(word[i]) == false)
            return false;
    return true;
}
```

```
1 reference
private bool IsLower(char c)
{
    // check if c is a lowercase letter (int value 65 to 90
    // or an apostrophe
    if (c >= 97 && c <= 122 || c.ToString() == "'")
        return true;
    else
        return false;
}
```

Submission Requirements via D2L:

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