RED RIVER COLLEGE

ADEV-2005 Programming 2
Applied Computer Education
Red River College

WINDOWS FORM APPLICATIONS

RED RIVER COLLEGE

Topics

- What is a Windows Form Application?
- What is a form?
- Designing a form
- Developing a Windows Application

Windows Applications

- Graphical User Interface (GUI)
  - Forms
  - Controls
  - Mouse interaction

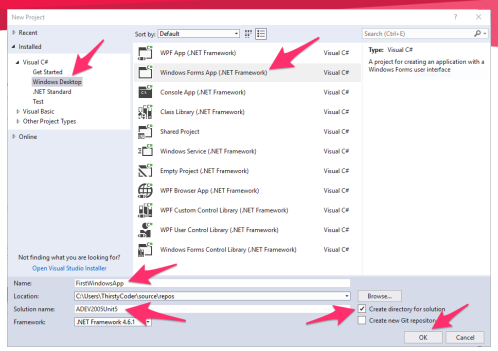
Event Driven Programing

- Common paradigm in GUI apps
- Flow of the program determined by events
- Usually dictated by the user
- Main (message) loop listens to events
  - Triggers callback function when detected

Windows Form Application

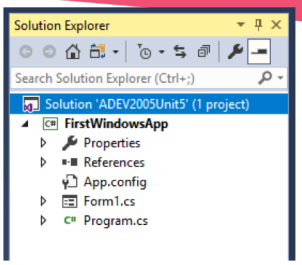
- A Visual Studio project template for creating windows applications

Creating a WFA Project

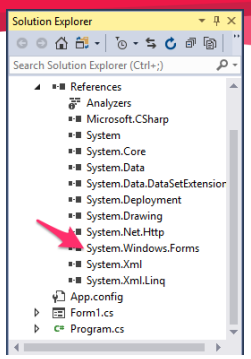


Windows Form Application Template

- References
- App.config
- Form1.cs
- Program.cs



Project References



Program.cs

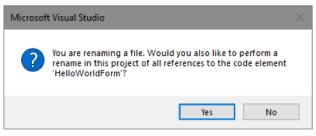
```
static class Program
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main()
    {
        Application.EnableVisualStyles();
        Application.SetCompatibleTextRenderingDefault(false);
        Application.Run(new Form1());
    }
}
```

Form1.cs

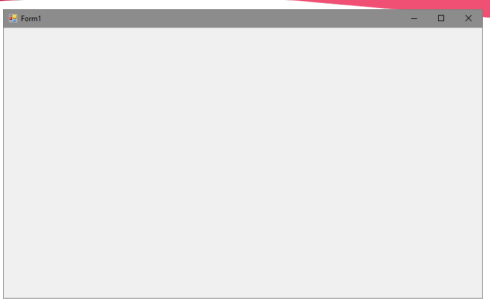
```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }
}
```

Renaming the Form Class

- You must rename the form class
- Form classes will always end with the word “Form”
- Renaming the .cs file, will prompt you to change the class name in the code



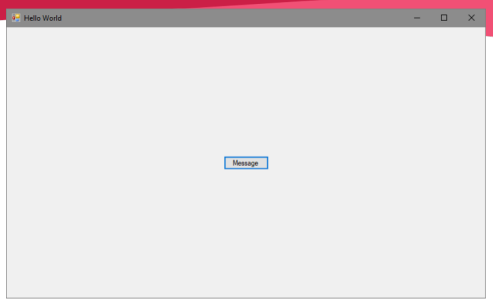
Run With No Code



Editing the Form Class

```
public partial class HelloWorldForm : Form
{
    private Button btnMessage;

    public HelloWorldForm()
    {
        InitializeComponent();

        this.btnMessage = new Button();
        this.btnMessage.Location = new System.Drawing.Point(363, 214);
        this.btnMessage.Name = "btnMessage";
        this.btnMessage.Size = new System.Drawing.Size(75, 23);
        this.btnMessage.Text = "Message";

        this.Text = "Hello World";
        this.StartPosition = FormStartPosition.CenterScreen;
        this.Controls.Add(this.btnMessage);
    }
}
```

Run with Code



Adding Functionality

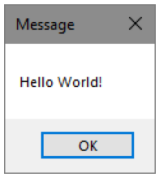
RED RIVER COLLEGE

```
public partial class HelloWorldForm : Form
{
    private Button btnMessage;

    public HelloWorldForm()
    {
        InitializeComponent();

        this.btnMessage.Click += BtnMessage_Click;
    }

    private void BtnMessage_Click(object sender, EventArgs e)
    {
        MessageBox.Show("Hello World!", "Message");
    }
}
```

Note: Not all of the original code is shown here.

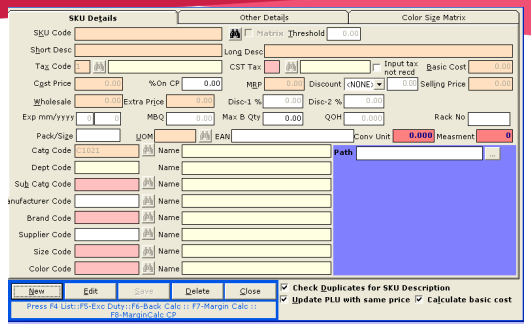
Test Functionality

RED RIVER COLLEGE



Imagine the Amount of Code

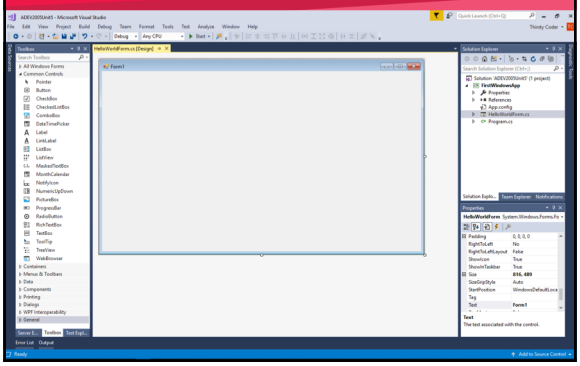
RED RIVER COLLEGE



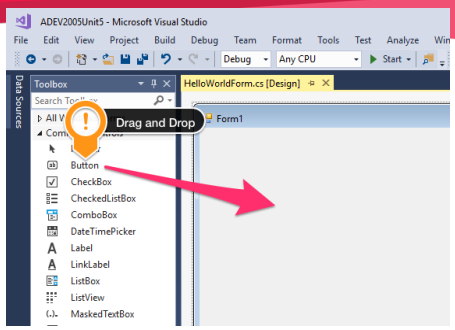
Editing a Form Class

- Two modes:
  - Design View
  - Code View
- Open design view
  - Double-click the .cs in solution explorer
- Open code view
  - F7
  - Right-click file in solution explorer
  - View Code icon in Solution Explorer toolbar

Form Class Design View



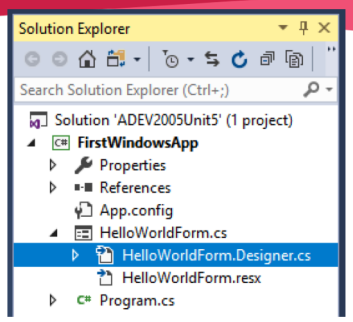
Adding Controls



Closer Look At The Form Class

```
public partial class HelloWorldForm : Form
{
    public HelloWorldForm()
    {
        InitializeComponent();
    }
}
```

Designer File



Original Designer File Code

```
partial class HelloWorldForm
{
    #region Windows Form Designer generated code

    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.components = new System.ComponentModel.Container();
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(800, 450);
        this.Text = "Form1";
    }

    #endregion
}
```

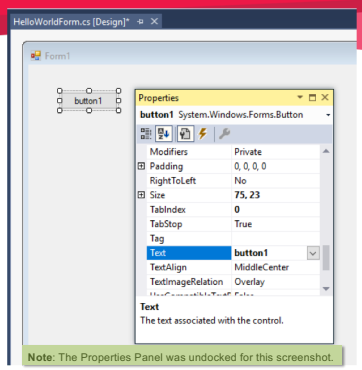
Note: Not all of the original code is shown here.

After Adding a Control In Design Mode

```
partial class HelloWorldForm
{
    private void InitializeComponent()
    {
        this.button1 = new System.Windows.Forms.Button();
        //
        // button1
        //
        this.button1.Location = new System.Drawing.Point(92, 71);
        this.button1.Name = "button1";
        this.button1.Size = new System.Drawing.Size(75, 23);
        this.button1.TabIndex = 0;
        this.button1.Text = "button1";
        this.button1.UseVisualStyleBackColor = true;

        this.Controls.Add(this.button1);
    }

    private System.Windows.Forms.Button button1;
}
```

Note: Not all of the original code is shown here.

Modifying a Controls State



Note: The Properties Panel was undocked for this screenshot.

What is a Form?

- Class
  - Instance (object) from class
  - Fields (states)
  - Properties
  - Inherited Properties (states)
    - Override properties
  - Inherited Methods (actions)
    - Override methods
  - Events
  - Abstract, Inheritance, Polymorphism
- Read the documentation

Unit 5 – Windows Form Applications

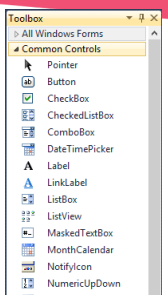
DESIGNING FORMS

Form is a Container

- Forms contain other controls
- Controls are added to a Form's collection of controls
  - **Controls** Property

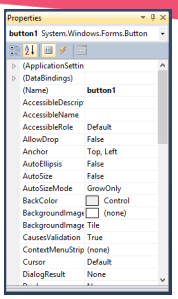
Toolbox

- Controls are classes
  - Example: System.Windows.Forms.Button
- All common windows controls can be found here
- Custom controls can be added



Properties Panel

- Contextual
- Not all properties listed
- Set initial state before runtime
- Bold means code has been added to the class
- Not bold means default value



Naming Controls

- The **Name** property in the Properties Panel
  - The variable name used in code
- Control names are prefixed with a code
- Example:
  - btnButton
  - txtFirstName
  - lblSalePrice
- All prefixes listed in Learn

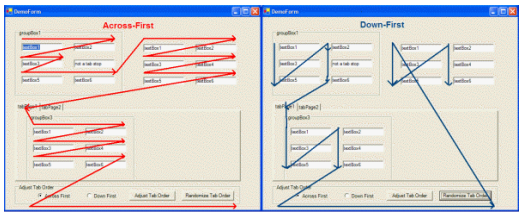
Adding Functionality

- Completed in the *.cs, **not** *.Designer.cs
  - Designer file is for design view generated code
- Starts with construction of the Form object
- Handle events to respond to interaction with the Form and its Controls

Tab Order

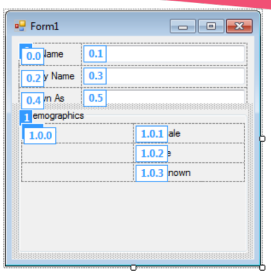
- Dictates the order controls receive input focus
- Keyboard
  - Tab and Shift+Tab
- Properties
  - TabIndex : int
  - TabStop : boolean

Sample Tab Order



Tab Order Tool

- View > Tab Order
- **Tip**: This tool is toggled off/on.



Summary

- Windows Form Application = GUI
- Forms are classes (two parts)
- Form container for controls
- Controls are classes
- Designer = visual programming

QUESTIONS?