RED RIVER COLLEGE

ADEV-2005 Programming 2
Applied Computer Education
Red River College

CONTROLS

RED RIVER COLLEGE

Topics

- Common Windows Form Controls
  - Common properties, methods and events
  - Purpose

Controls (1 of 2)

- Each serves a specific purpose
- Choosing the wrong control can make the application difficult to use
- Goal is to develop your application like other Windows applications

Controls (2 of 2)

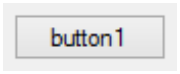
- Derive from System.Windows.Forms.Control Class
- All controls have a common set of:
  - Properties
  - Method
  - Events

Controls - Common Properties & Methods

- Properties
  - Dock
  - Enabled
  - Font
  - Height
  - Name
  - Size
  - Text
  - Visible
  - Width
- Methods
  - Focus()

Button Control

- User initiates an action
- Click or <enter> when focused

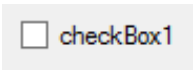


Button Class – Common Members

- Properties
  - Name (prefix btn)
  - Text
- Events
  - Click

Checkbox Control

- Boolean input (toggle)
- Select an option; true/false, yes/no



Checkbox Class – Common Members

- Properties
  - Name (prefix chk)
  - Checked
  - Text
- Events
  - CheckedChanged

Label Control

- Output
- Display text

Label Class – Common Members

- Properties
  - Name (prefix lbl)
  - AutoSize
  - BorderStyle
  - Text
- Events
  - TextChanged

NumericUpDown Control

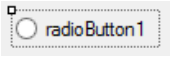
- Input numeric value
- Will not allow letter or special character input

NumericUpDown Class – Common Members

- Properties
  - Name (prefix nud)
  - DecimalPlaces
  - Increment
  - Maximum
  - Minimum
  - TextAlign
  - ThousandsSeparator
  - Value
- Events
  - ValueChanged

RadioButton Control

- Select one of various options
- Two or more options
- Grouped by the container control (Collection) they are in



RadioButton Class – Common Members

- Properties
  - Name (prefix rad)
  - Checked
  - Text
- Events
  - CheckedChanged

TextBox Control

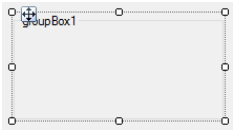
- Input
- Single or Multi-line

TextBox Class – Common Members

- Properties
  - Name (prefix txt)
  - Size
  - Text
  - TextAlign
- Methods
  - Clear()
  - SelectAll()
- Events
  - Enter
  - Leave
  - TextChanged

GroupBox Control

- Container control
- Groups other controls
- Visible border and text



GroupBox Class – Common Members

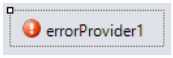
- Properties
  - Name (prefix grp)
  - Size
  - Text

Panel Control

- Same as Group Box
- Minus border and text
- Prefix pnl

ErrorProvider Component

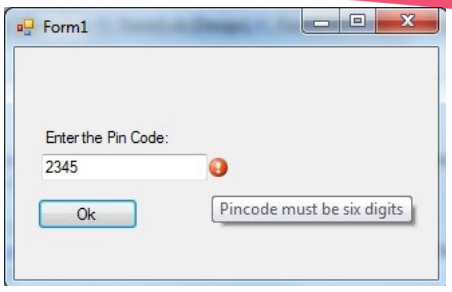
- Indicates an input error
- Design view – component panel
- Only one instance needed per form



ErrorProvider Class - Common Members

- Properties
  - Name Property (errorProvider)
  - BlinkStyle Property (NeverBlink)
- Methods
  - SetError(Control, String) : void
  - GetError(Control) : String
  - SetIconPadding(Control, int) : void

ErrorProvider On a Form



Form is a Control

Inheritance:

Object → MarshalByReference → Component → Control → ScrollableControl → ContainerControl → Form

Form Class – Common Properties & Methods

- Properties
  - AcceptButton
  - CancelButton
  - Controls
  - FormBorderStyle
  - Height
  - MaximizeBox
  - MinimizeBox
  - ShowIcon
  - StartPosition
  - Text
  - Visible
  - Width
- Methods
  - Close()
  - Show()
  - ShowDialog()

Form Events

Show

1. Control.HandleCreated
2. Control.BindingContextChanged
3. **Form.Load**
4. Control.VisibleChanged
5. Form.Activated
6. **Form.Shown**

Close

1. Form.Closing (obsolete)
2. **Form.FormClosing**
3. Form.Closed (obsolete)
4. **Form.FormClosed**
5. Form.Deactivated

Unit 5 - Controls

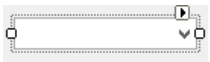
LIST CONTROLS

Items Property

- Collection of objects the list control stores
- Can do all your typical collection functions
- Objects displayed to the UI using their ToString method

ComboBox Control

- Used for selecting items from a list
- Often used in place of RadioButtons when there are many options

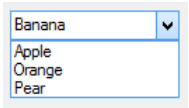


DropDownStyle Property

- Three different styles:
  - Simple
  - DropDown (default)
  - DropDownList

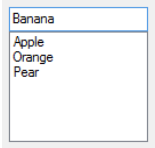
DropDown Style (default)

- Choose item from a list by clicking the arrow
  - List is only visible when arrow is clicked
  - Selected Item text displays in the control
- Text can be entered by the user if the item is not in the list



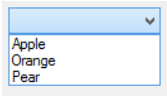
Simple Style

- Choose item from a list
  - List is always visible
  - Selected Item text displays in the control
- Text can be entered by the user if the item is not in the list



DropDownList Style

- Choose item from a list by clicking the arrow
  - List is only visible when arrow is clicked
  - Selected Item text displays in the control
- Text cannot be entered by the user

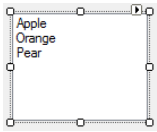


ComboBox Class – Common Members

- Properties
  - Name (Prefix cbo)
  - DropDownStyle
  - Items
  - SelectedIndex
  - SelectedItem
  - Text
- Events
  - SelectedIndexChanged Event

ListBox Control

- Multiple items visible
- Supports multiple selections



SelectionMode Property

- Four modes:
  - None
  - One (default)
  - MultiSimple
  - MultiExtended

ListBox Class – Common Members

- Properties
  - Name (Prefix lst)
  - Items
  - SelectedIndex
  - SelectedIndices
  - SelectedItem
  - SelectedItems
  - SelectionMode
- Methods
  - ClearSelection()
- Events
  - SelectedIndexChanged

Unit 5 - Controls

MENUS

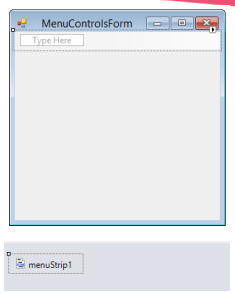
MenuStrip Control

- AKA Main menu
- Text menu docked at the top of the window
- Contains all tasks of the application
- Can contain contextual items

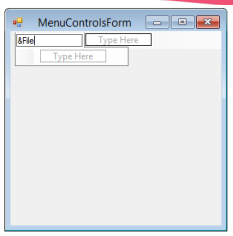
Menu Strip Standards

- File always the first item
  - Exit always last child item
- Help always last item
  - About always last child item
- Always use standard shortcuts
  - Ctrl+S
  - Ctrl+C
  - F1, etc
- Menu items that open a dialog window must end with "…"
  - Example: Tools > Options…

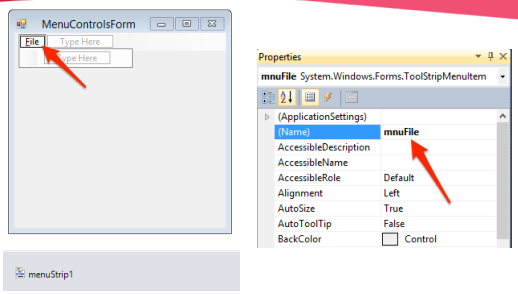
Designing a Menu Strip



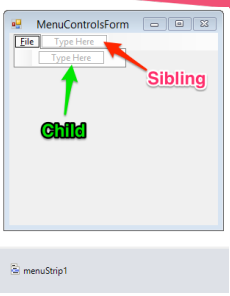
Adding Menu Items



Rename Control



Sibling and Child Items



MenuStrip Class – Common Members

- Properties
  - Name (prefix ms)
  - Items

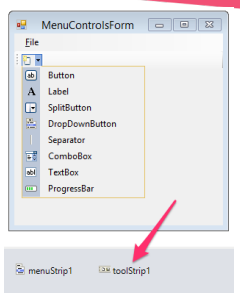
ToolStripMenuItem Class - Common Members

- Properties
  - Name (mnuRootSub[Sub…]Item)
  - Image
  - ShortcutKeys
  - Text
    - & for keyboard accessor
    - - (hyphen) for separator bar
- Events
  - Click

ToolStrip Control

- AKA Toolbar
- Icon bar docked at the top of a container
- Common tasks

Designing ToolStrip



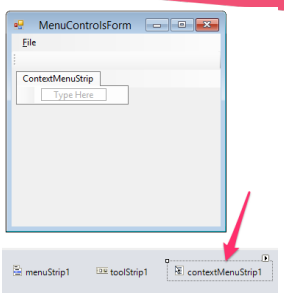
ToolStrip Class - Common Members

- Properties
  - Name (prefix ts)
  - Items

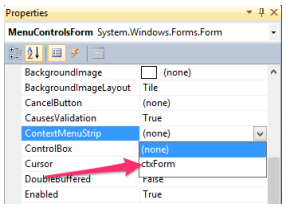
ContextMenuStrip Control

- Contextual menu
- Display tasks for the item (control) that was right-clicked on
- Menu displays and hides automatically

Designing a ContextMenuStrip



Assigning ContextMenuStrip to Control



ContextMenuStrip Class - Common Members

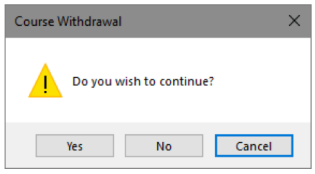
- Properties
  - Name (prefix cms, ctx)
  - Items

Dialog Box

- AKA Message Box
- MessageBox static class
- Show() method
- Window is opened as modal

MessageBox Class Example

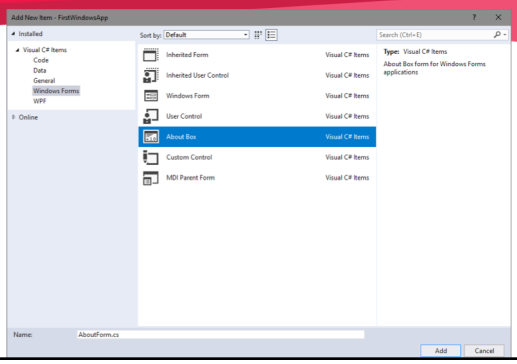
```
MessageBox.Show("Do you wish to continue?",
                "Course Withdrawal",
                MessageBoxButtons.YesNoCancel,
                MessageBoxIcon.Exclamation,
                MessageBoxDefaultButton.Button3);
```



Unit 5 - Controls

ABOUT BOX

Adding an About Box

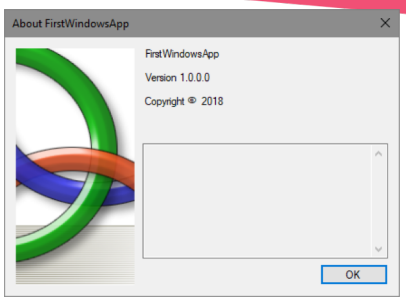


About Box Template

```
partial class AboutForm : Form
{
    InitializeComponent();
    this.Text = String.Format("About {0}", AssemblyTitle);
    this.labelProductName.Text = AssemblyProduct;
    this.labelVersion.Text = String.Format("Version {0}", AssemblyVersion);
    this.labelCopyright.Text = AssemblyCopyright;
    this.labelCompanyName.Text = AssemblyCompany;
    this.textBoxDescription.Text = AssemblyDescription;
}
```

Note: Not all of the code is shown here.

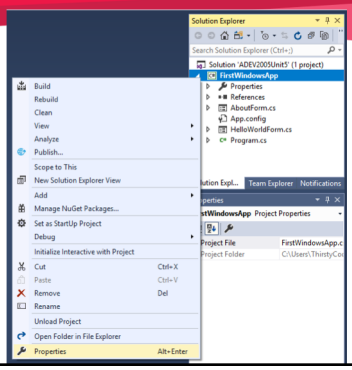
About Box Design



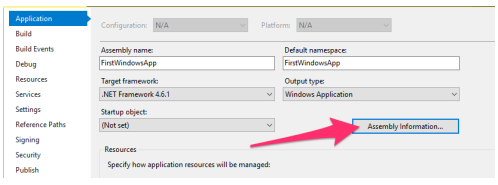
About Box Implementation

- Already coded; DO NOT modify
- Accesses the assembly (.exe) metadata

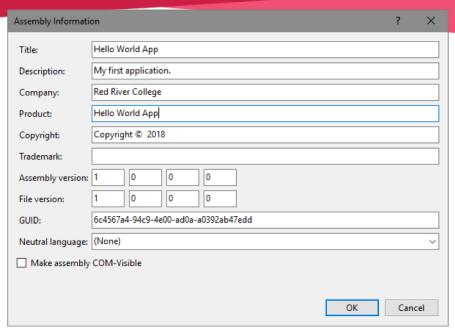
Accessing Project Properties



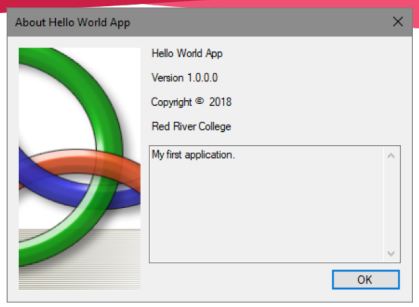
Accessing Assembly Information



Editing Assembly Information

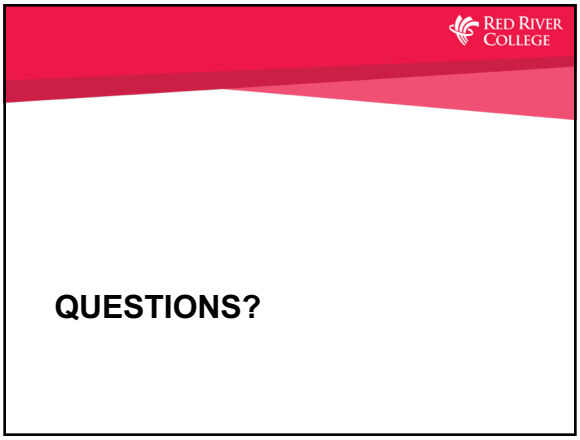


About Dialog Window



Summary

- All controls derive from Control class
  - Common properties, methods and events
- Every control has a purpose; choose wisely
- About Box – Class already contains implementation


RED RIVER COLLEGE
QUESTIONS?