Chapter 2

How to design a Windows Forms application

Objectives

Applied

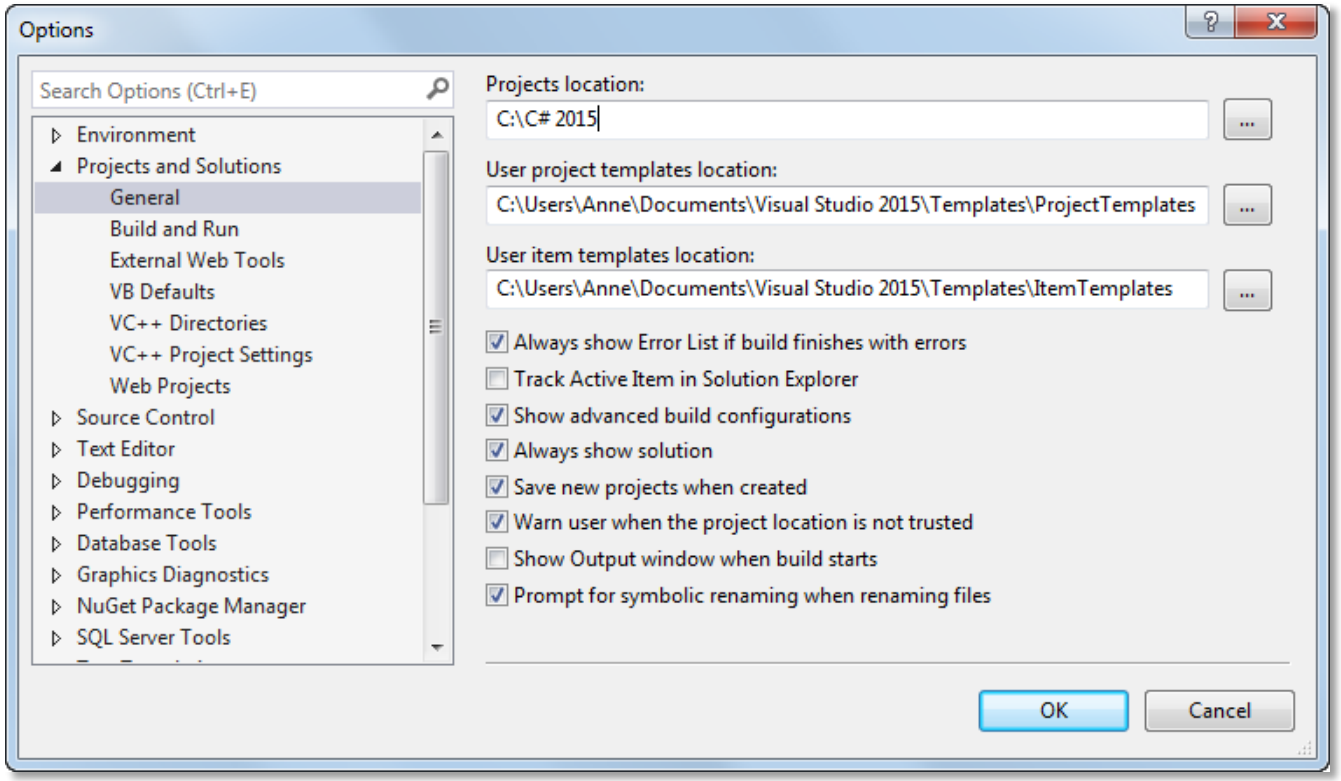
1. Given the form design and property settings for a simple application, use the Form Designer to design the form.
2. When necessary, rename the form, project, and solution files for an application.
3. Customize the Visual Studio environment for use with C# by setting the Visual Studio options and the import and export settings.

Objectives (cont.)

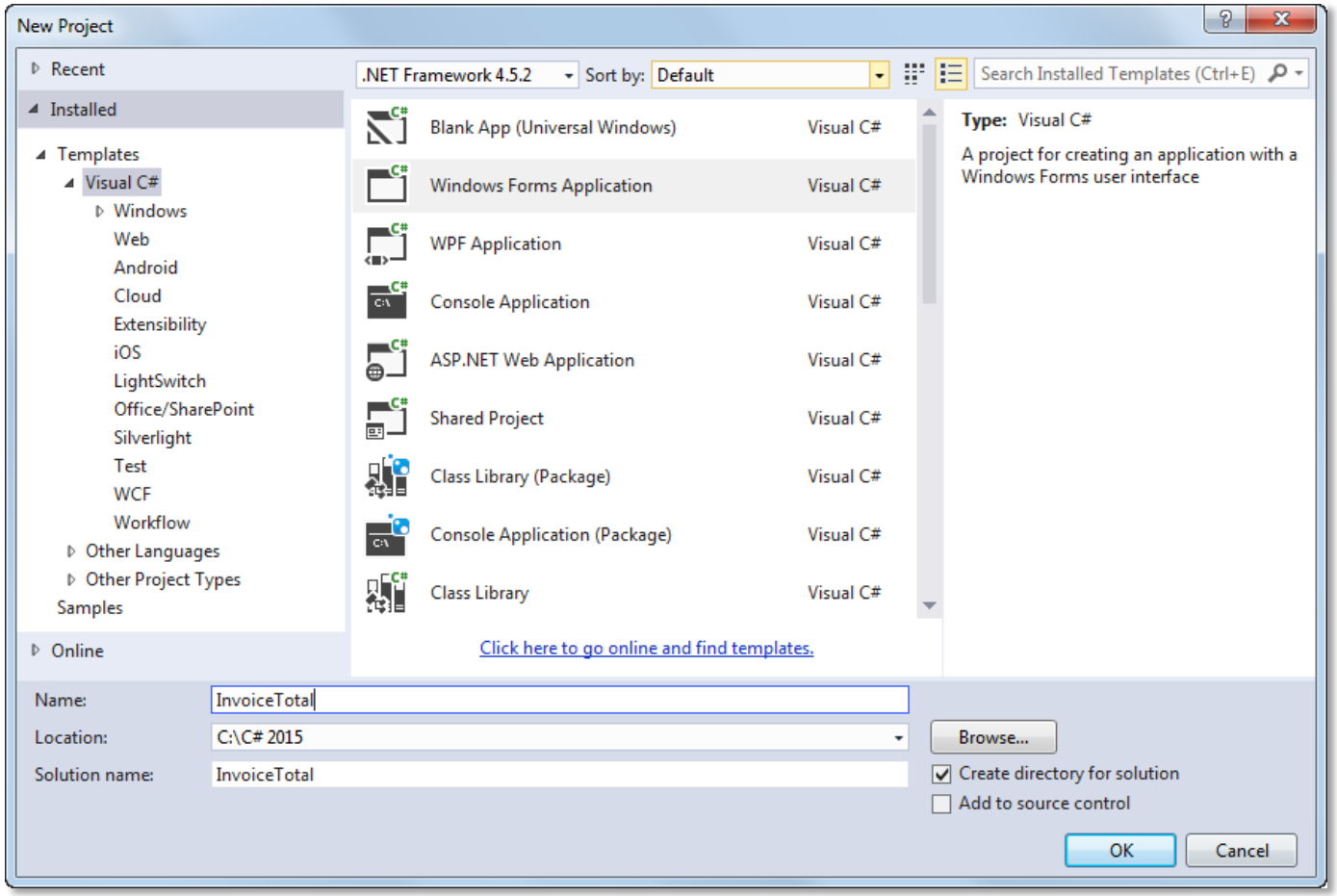
Knowledge

1. Describe the use of the Toolbox and Properties window.
2. Describe the Name and Text properties for a form or control.
3. Describe these additional properties for a control: Enabled, ReadOnly, and TextAlign.
4. Describe the way you adjust the tab order for the controls on a form, set access keys for controls, and set the default buttons for the Enter and Esc keys for a form.

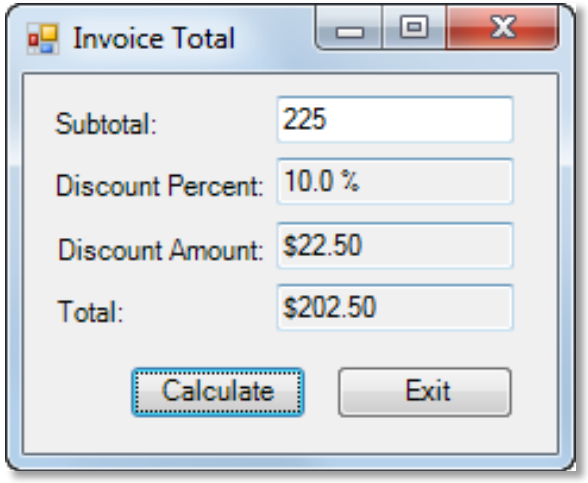
The Options dialog box for setting the project options



The New Project dialog box



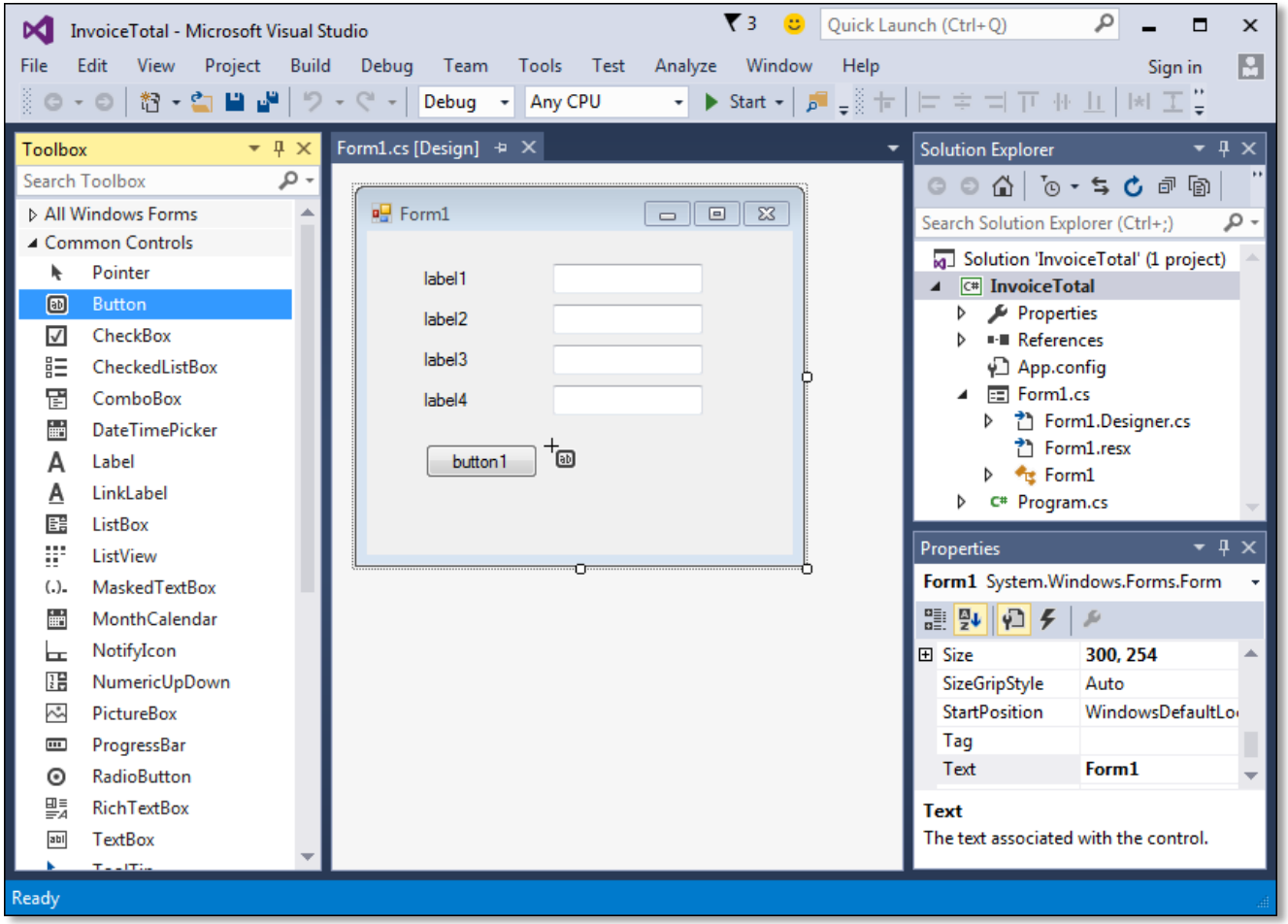
The Invoice Total form



Three types of controls

- A *label* displays text on a form.
- A *text box* lets the user enter text on a form.
- A *button* initiates form processing when clicked.

A form after some controls have been added to it



Three ways to add a control to a form

- Select the control in the Toolbox. Then, click in the form where you want to place the control. Or, drag the pointer on the form to place the control and size it at the same time.
- Double-click the control in the Toolbox. Then, the control is placed in the upper left corner of the form.
- Drag the control from the Toolbox and drop it on the form. Then, the control is placed wherever you drop it.

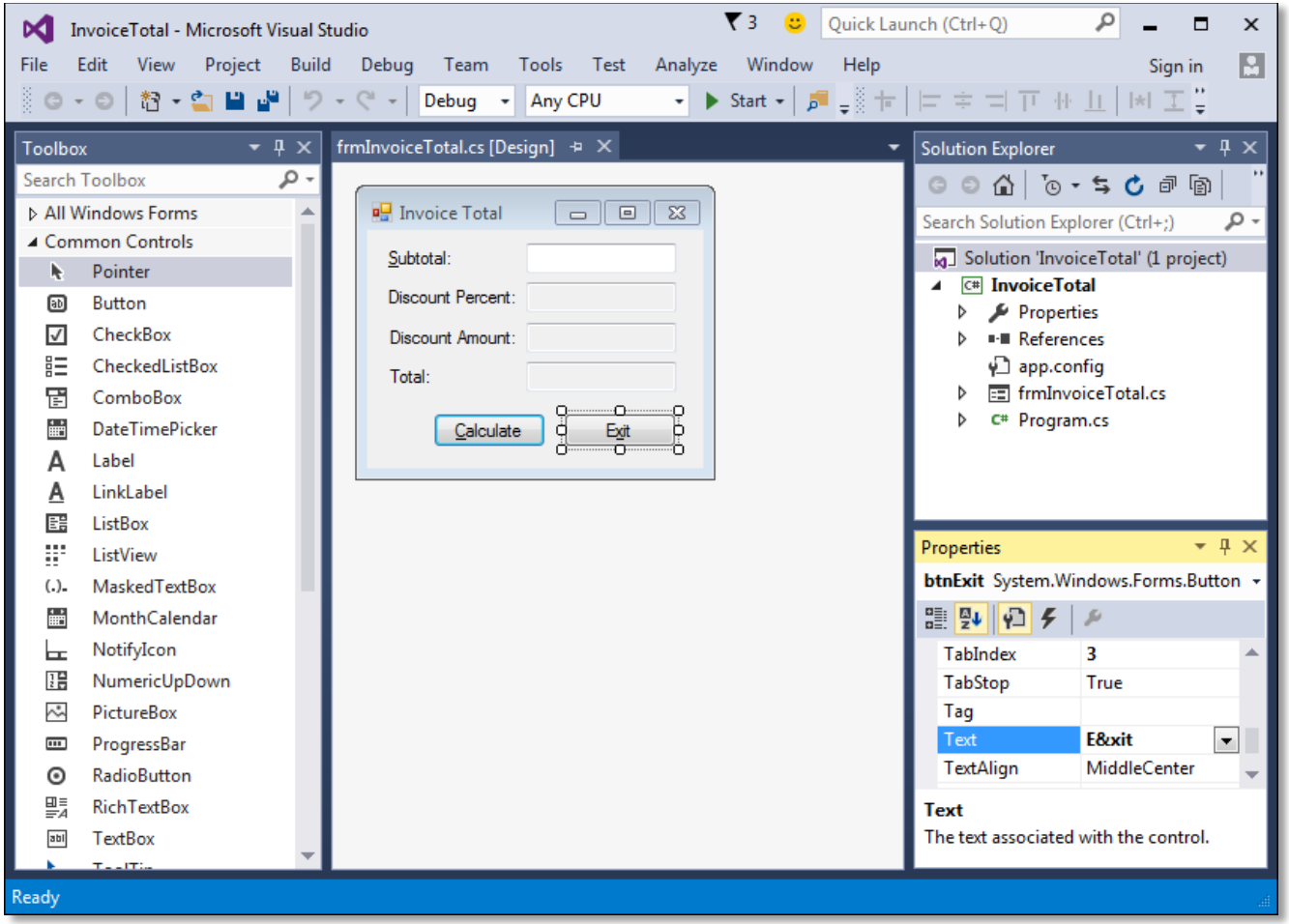
How to select and work with controls

- Use standard Windows techniques to select, move, size, or group the controls on a form.
- You can also select a group of controls by clicking on a blank spot in the form and then dragging around the controls.
- To align, size, or space a group of selected controls, click on a control to make it the *primary control*. Then, use the commands in the Format menu or the buttons on the Layout toolbar to align, size, or space the controls relative to the primary control.
- You can also size all of the controls in a group by sizing the primary control in the group.

Note

- A label is sized automatically based on the amount of text that it contains. As a result, you can't size a label by dragging its handles unless you change its AutoSize property to False.

A form after the properties have been set



The Name property

- Sets the name you use to identify a control in your C# code.
- Can be changed to provide a more descriptive and memorable name for forms and controls that you will refer to when you write your code (such as text boxes and buttons).
- Doesn’t need to be changed for controls that you won’t refer to when you write your code (such as most labels).
- Can use a three-letter prefix to indicate whether the name refers to a form (frm), button (btn), label (lbl), or text box (txt).

The Text property

- Sets the text that's displayed on the form or control.
- For a form, the Text value is displayed in the title bar. For controls, the Text value is displayed directly on the control.
- For a text box, the Text value changes when the user types text into the control, and you can write code that uses the Text property to get the text that was entered by the user.

Other properties for forms

Property	Description
`AcceptButton`	Identifies the button that will be activated when the user presses the Enter key.
`CancelButton`	Identifies the button that will be activated when the user presses the Esc key.
`StartPosition`	Sets the position at which the form is displayed. To center the form, set this property to CenterScreen.

Other properties for controls

Property	Description
`Enabled`	Determines whether the control will be enabled or disabled.
`ReadOnly`	Determines whether the text in some controls like text boxes can be edited.
`TabIndex`	Indicates the control's position in the tab order, which determines the order in which the controls will receive the focus when the user presses the Tab key.
`TabStop`	Determines whether the control will accept the focus when the user presses the Tab key to move from one control to another. Some controls, like labels, don't have the TabStop property.
`TextAlign`	Sets the alignment for the text displayed on a control.

How to adjust the tab order

- *Tab order* refers to the sequence in which the controls receive the *focus* when the user presses the Tab key. You should adjust the tab order so the Tab key moves the focus from one control to the next in a logical sequence.
- Each control has a TabIndex property that indicates the control's position in the tab order. You can change this property to change a control's tab order position.
- If you don't want a control to receive the focus when the user presses the Tab key, change that control's TabStop property to False.
- Label controls don't have a TabStop property so they can't receive the focus.

How to set access keys

- *Access keys* are shortcut keys that the user can use in combination with the Alt key to quickly move to individual controls on the form.
- You use the Text property to set the access key for a control by placing an ampersand immediately before the letter you want to use for the access key.
- Since the access keys aren't case sensitive, &N and &n set the same access key.
- When you set access keys, make sure to use a unique letter for each control.
- You can't set the access key for a text box. However, if you set an access key for a label that immediately precedes the text box in the tab order, the access key will take the user to the text box.
- If you assign an access key to a button, the button is activated when you press Alt plus the access key.

How to set the Enter and Esc keys

- The AcceptButton property of the form sets the button that will be activated if the user presses the Enter key.
- The CancelButton property of the form sets the button that will be activated if the user presses the Esc key. This property should usually be set to the Exit button.
- You set the AcceptButton or CancelButton values by choosing the button from a drop-down list that shows all of the buttons on the form. So be sure to create and name the buttons you want to use before you attempt to set these values.

The property settings for the Invoice Total form

Default name	Property	Setting
Form1	Text	Invoice Total
	AcceptButton	btnCalculate
	CancelButton	btnExit
	StartPosition	CenterScreen

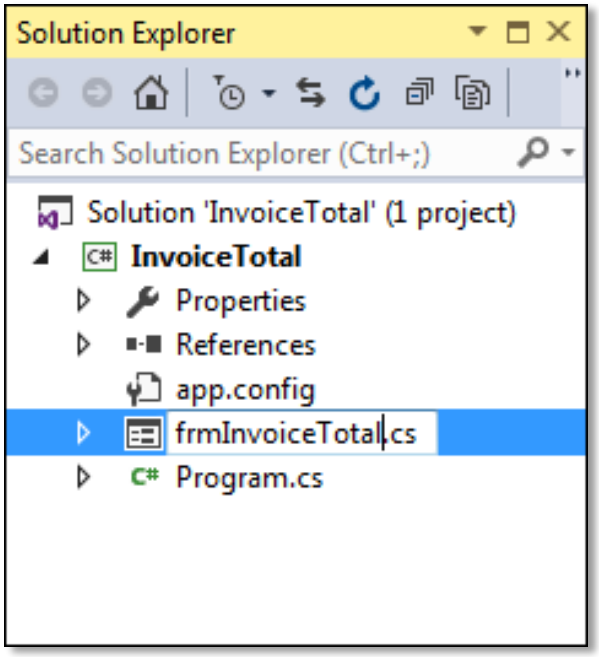
The property settings for the controls

Default name	Property	Setting
label1	Text	&Subtotal:
	TextAlign	MiddleLeft
	TabIndex	0
label2	Text	Discount percent:
	TextAlign	MiddleLeft
label3	Text	Discount amount:
	TextAlign	MiddleLeft
label4	Text	Total:
	TextAlign	MiddleLeft
textBox1	Name	txtSubtotal
	TabIndex	1
textBox2	Name	txtDiscountPercent
	ReadOnly	True
	TabStop	False

The property settings for the controls (cont.)

textBox3	Name	txtDiscountAmount
	ReadOnly	True
	TabStop	False
textBox4	Name	txtTotal
	ReadOnly	True
	TabStop	False
button1	Name	btnCalculate
	Text	&Calculate
	TabIndex	2
button2	Name	btnExit
	Text	E&xit
	TabIndex	3

The Solution Explorer as a form file is being renamed



How to rename a file, project, or solution

- Right-click on it in the Solution Explorer window and select the Rename command from the shortcut menu. Or, you can select it in the Solution Explorer and press F2. Then, you can enter the new name.
- Be sure not to change or omit the file extension when you rename a file. Remember too that using a three-letter prefix to indicate the contents of the file (like *frm* for a form file) makes it easier to tell what each file represents.
- When you change the name of a file, Visual Studio also changes the File Name property for the form and asks you if you want to change all references to the file.
- If you change the name of a project, you may also want to change the name of the assembly that's created for the project and the name of the namespace that contains the project from the Application tab of the Project Properties window.

How to save a file, project, or solution

- You can use the Save All button in the Standard toolbar or the Save All command in the File menu to save all files and projects in the solution.
- You can use the Save button in the Standard toolbar or the Save command in the File menu to save a file, project, or solution. The files that are saved depend on what's selected in the Solution Explorer window.
- If you try to close a solution that contains modified files, a dialog box is displayed that asks you if you want to save those files.