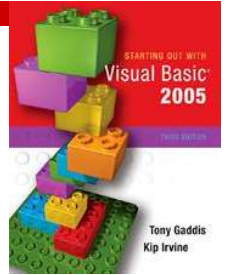


2.1

Building the Hotel Directions Application

In This Section You Create Your First Visual Basic Application: a Window That Displays a Map and Road Directions to a Hotel



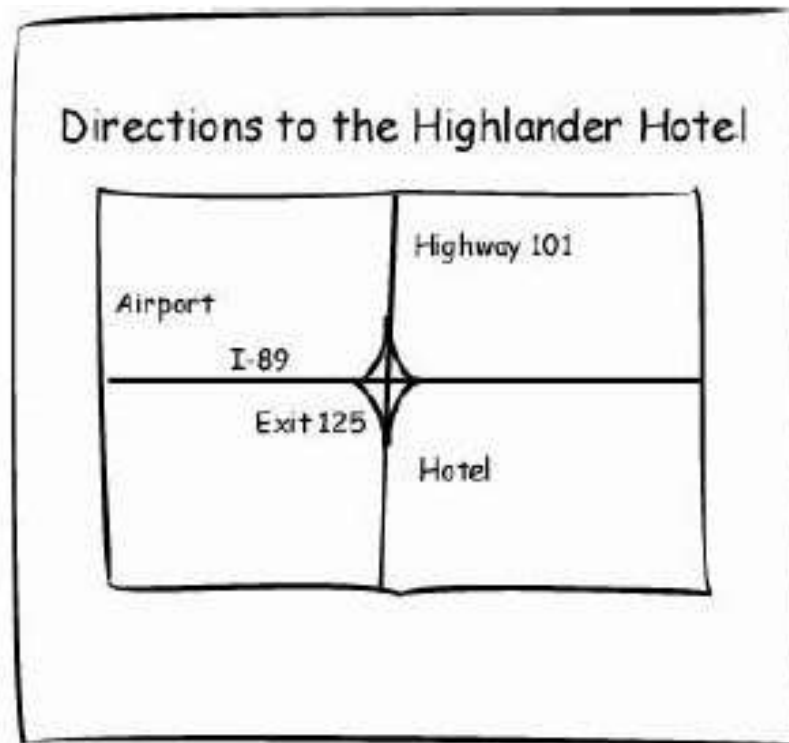
Define What the Program is To Do

- Purpose: Display a map to the Highlander Hotel
- Input: None
- Process: Display a form
- Output: Display a graphic image showing a map on the form



Visualize and Design the User Interface

- Below is a sketch of the form to be shown





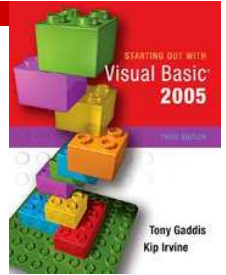
List the Controls Needed

<u>Control Type</u>	<u>Control Name</u>	<u>Description</u>
Form	Form1 (Default Name)	A small form that will serve as the window onto which the other controls will be placed
Label	Label1 (Default Name)	Displays the message "Directions to the Highlander Hotel"
PictureBox	PictureBox1 (Default Name)	Displays the graphic image showing the map to the hotel



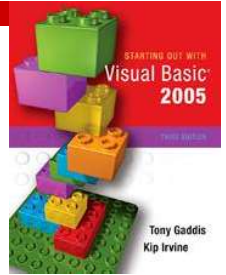
Define Control Relevant Property Values

- Form
 - Name: Form1
 - Text: "Directions"
- Label
 - Name: Label1
 - Text: "Directions to the Highlander Hotel"
 - TextAlign: MiddleCenter
 - Font: Microsoft sans serif, bold, 18 point
- PictureBox
 - Name: PictureBox1
 - Picture: HotelMap.jpg
 - SizeMode: StretchImage



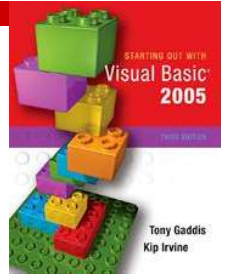
Use VB to Create the Application

- Establish the Form and set its Text property
- Add a Label control
 - Position and resize it on the form
 - Set Text, TextAlign, and Font properties
- Add a PictureBox control
 - Position and resize it on the form
 - Set Image property to display HotelMap.jpg
- Run the application
- Close and save the application



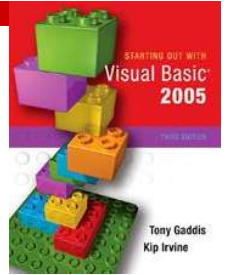
Project Organization on Disk

- User creates a new *project* in Visual Studio
 - A *solution* and a folder are created at the same time with the same name as the project
 - The project *belongs* to the solution
 - Multiple projects *can be* included in a solution
- The folder stores files related to the project including:
 - A *solution file* (.sln)
 - A *project file* (.vbproj)



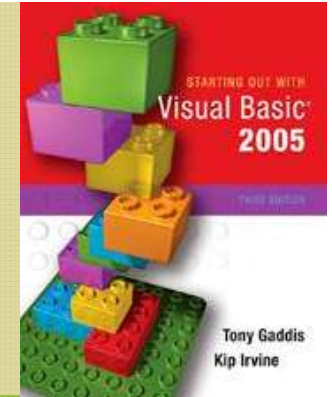
Opening an Existing Project

- Use Recent Projects list on Start Page
 - Provided it hasn't been moved or deleted
- Use Open Project button on Start Page
 - Then browse using Open Project dialog box
- Use Open Project option on File menu
 - Then browse using Open Project dialog box



Properties Window

- Used to view and modify the property values of a given object
- Two views of the properties are available:
 - Alphabetic (across all properties)
 - Categorized (groups properties by logical use)



2.2

Responding to Events

An Application Responds to Events, Such As Mouse Clicks and Keyboard Input, by Executing Code Known As Event Procedures

Write the Event Procedures for the Directions Application

Augment the Hotel Application

- Now the hotel owner wants to add an option to view written directions:





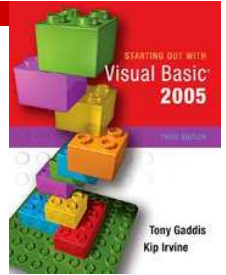
Controls to be Added

<u>Control Type</u>	<u>Control Name</u>	<u>Description</u>
Label	lblDirections	Displays written directions to the hotel
Button	btnDisplayDirections	When clicked, causes lblDisplayDirections text to appear on the form
Button	btnExit	Stops the application when clicked



Control Properties

- Label:
 - Name: lblDirections
 - Text: "Traveling on I-89, take..."
 - Visible: False
- Button:
 - Name: btnDisplayDirections
 - Text: "Display Directions"
- Button:
 - Name: btnExit
 - Text: "Exit"



Method btnDisplayDirections_Click

Marks the beginning of this event procedure

Name of the control that owns the event procedure

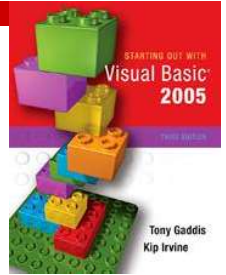
Name of the event the procedure responds to

Line Continuation Mark

```
Private Sub btnDisplayDirections_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles btnDisplayDirections.Click
    ' Make the directions visible
    lblDirections.Visible = True
End Sub
```

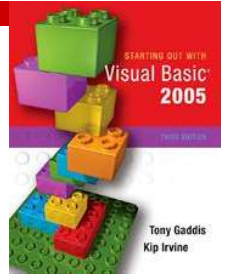
Event handled by this procedure

Makes the control lblDirections visible:
Assigns the value True to the Visible Property
of the lblDirections control.



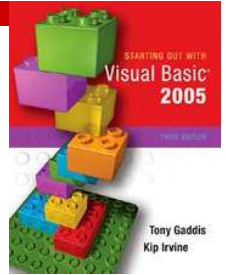
Syntax for Referring to the Value of a Control's Property

- Specify the control name (**lblDirections**)
- Then a dot
- Then the PropertyName (**Visible**)
- For example:
 - **lblDirections.Visible**
 - refers to the Visible property of the lblDirections control
 - The visible property values may only be true or false



Syntax for an Assignment Statement

- Specify the item to receive the value
- Then the equal symbol
- Then the value to be assigned
- For example:
 - `lblDirections.Visible = True`
 - Assigns the value True to the Visible property of the lblDirections control
 - Causes the text of the lblDirections control to become visible to the user



Method btnExit_Click

Marks the beginning of this event procedure

Name of the control that owns the event procedure

Name of the event the procedure responds to

Line Continuation Mark

```
Private Sub btnExit_Click (ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles btnExit.Click  
    ' End the application by closing the window  
    Me.Close()  
End Sub
```

Event handled by this procedure

Closes the current form,
referred to as *Me*,
and ends the program



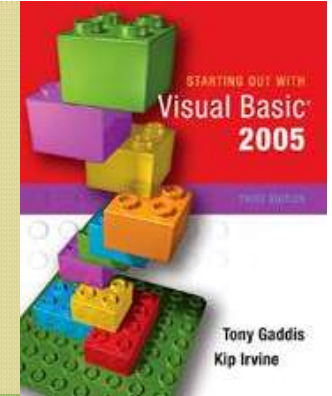
Use Visual Basic to Update the Application

- Place the label and the buttons on the form
- Enter the code for the two procedures
- Test the application



Additional Properties

- Color properties for a control:
 - BackColor: Sets the background (fill) color
 - ForeColor: Sets the foreground (text) color
- Border style properties for a form:
 - Sizable: (Default) Has min, max, and close buttons; can be resized by dragging edges
 - Fixed3D: Has a 3D look; min, max, and close buttons; cannot be resized
 - FixedSingle: Has single line border; min, max, and close buttons; cannot be resized



2.3

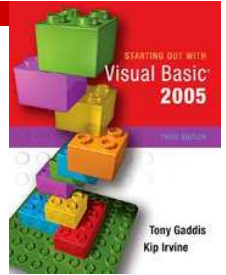
Modifying the Text Property With Code

Quite Often, You Will Need to Change a Control's Text Property With Code
This Is Done With an Assignment Statement



Modifying the Text Property in Code

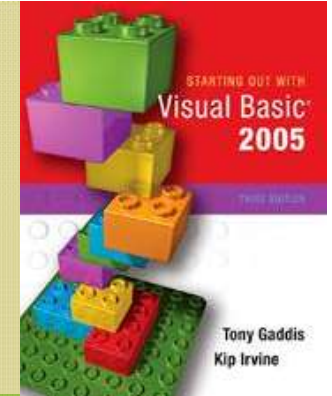
- Suppose a form is established with a label `lblMessage` whose Text property is:
1 Kilometer = ?
- And on a `btnFeet` button click, we want to change the value of the text property to:
1 Kilometer = 3,281 feet



Modifying the Text Property in Code

```
Private Sub btnFeet_Click(ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles btnFeet.Click  
    ' Display the conversion to feet.  
    lblMessage.Text = "1 Kilometer = 3,281 feet"  
End Sub
```

↑
Assigns the string to the right of the equal sign to the text property of lblMessage
This replaces the previous text property of lblMessage with the new value shown



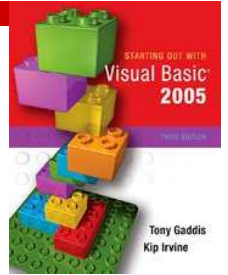
2.4

The AutoSize, BorderStyle, and TextAlign Properties

The Label Control's AutoSize Property Allows a Label to Change Size Automatically to Accommodate the Amount of Text in its Text Property

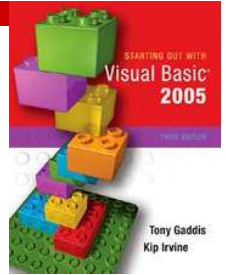
The BorderStyle Property Allows You to Set a Border Around a Label Control





AutoSize Property for Labels

- AutoSize is a *Boolean* (either True or False) Property of labels
- False (the default) means the box size will not change, regardless of the amount of text assigned to it
- True means the box will automatically resize itself to fit the amount of text assigned to it



BorderStyle Property for Labels

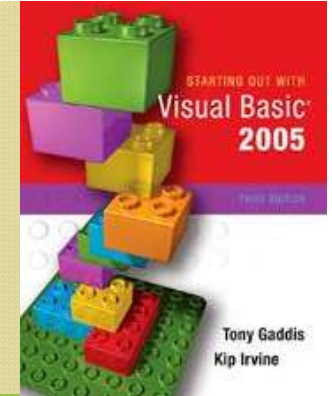
- BorderStyle determines the look of the box
 - None (the default) means no border
 - FixedSingle results in a border one pixel wide
 - Fixed3D gives the border a recessed 3-dimensional look



TextAlign Property for Labels

- The value of TextAlign establishes the alignment (or justification) or the text:
 - TopLeft ■ MiddleLeft ■ BottomLeft
 - TopCenter ■ MiddleCenter ■ BottomCenter
 - TopRight ■ MiddleRight ■ BottomRight
- The assignment statement below forces the text of lblTitle to appear in the middle center of the label

```
lblTitle.TextAlign = ContentAlignment.MiddleCenter
```



2.5

Clickable Images

Controls Other Than Buttons Have Click
Event Procedures

PictureBox Controls Can Respond to
Mouse Clicks



PictureBox Control

- As we saw earlier the Image Property can be set to a graphic image of some sort
- The flag images in Tutorial 2-16 are clickable
- The click event can be handled by code to take whatever action is desired



PictureBox Click Event code

- When PictureBox picUSA is clicked, the lblMessage text property is set to display *United States of America*

```
Private Sub picUSA_Click(ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles picUSA.Click  
    ' Display the country name  
    lblMessage.Text = "United States of America"  
End Sub
```