

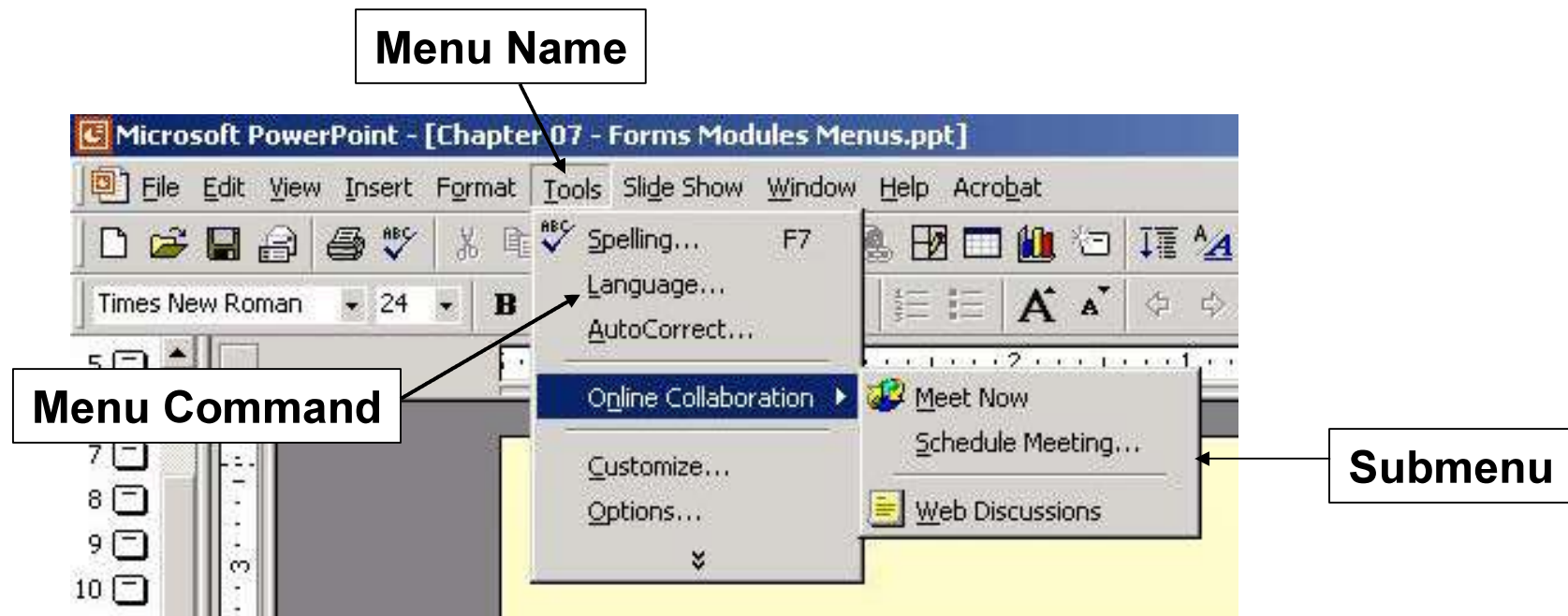
7.3

Menus

Visual Basic Allows You to Create a System of Drop-down Menus for Any Form in Your Application
You Use the Menu Designer to Create a Menu System

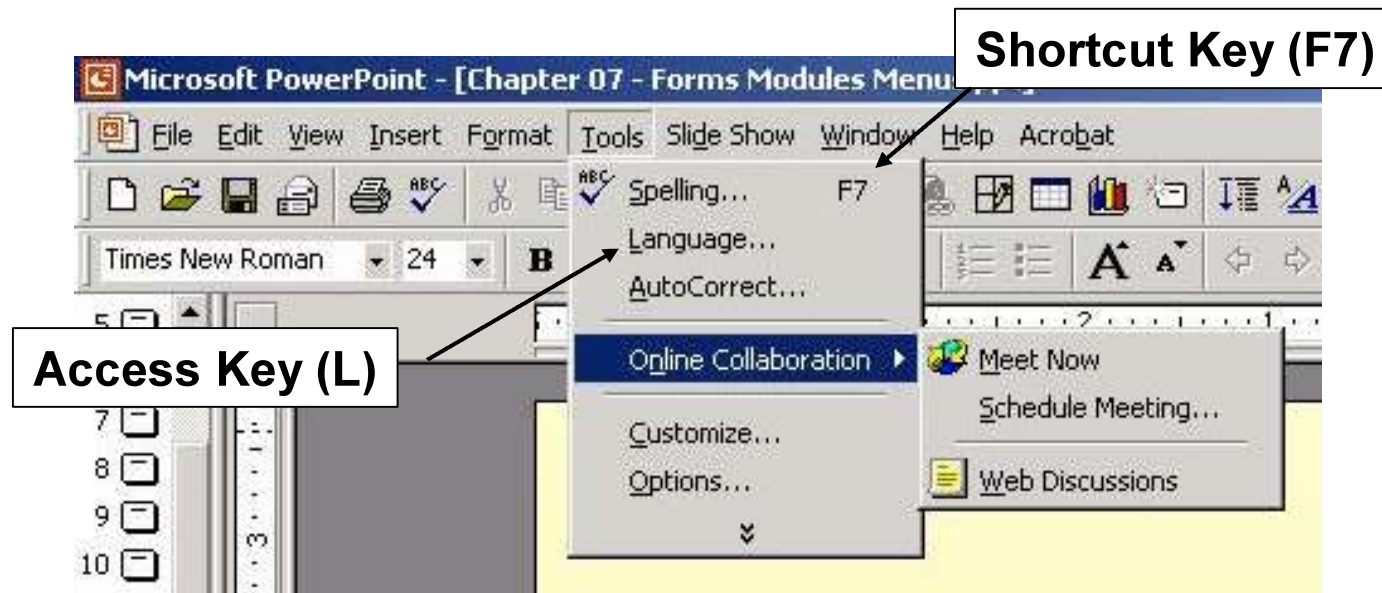


Components of a Menu System



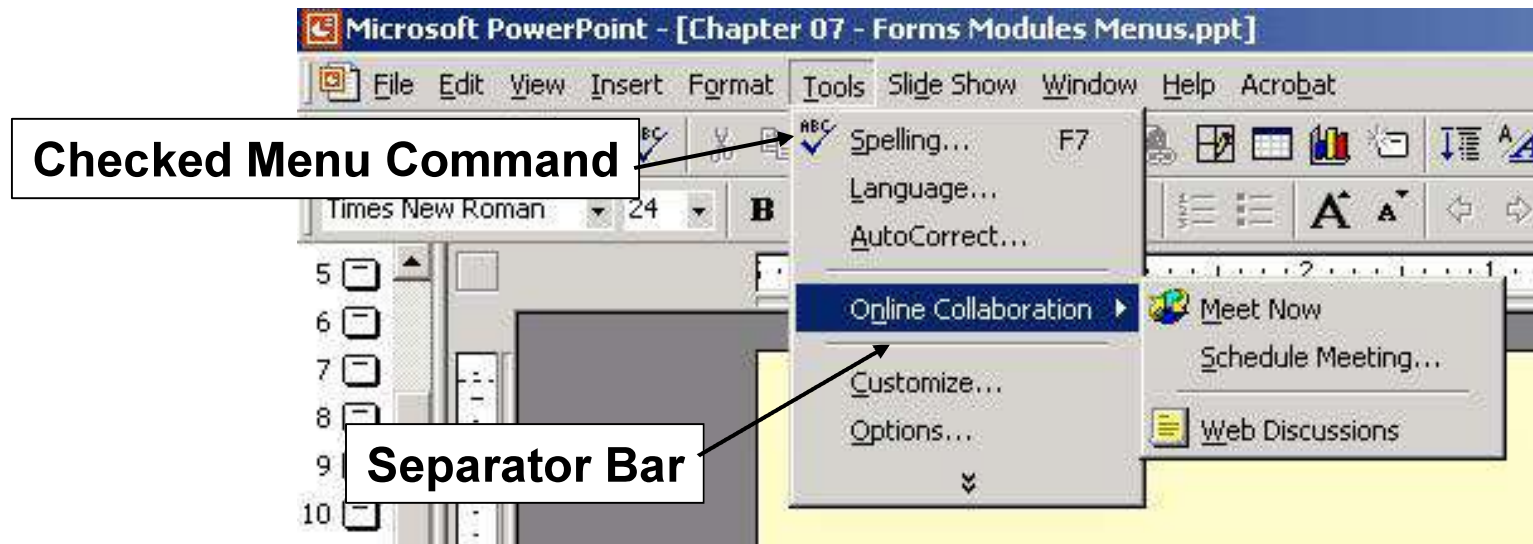
- Each drop-down menu has a *menu name*
- Each drop-down menu has a list of actions or *commands* that can be performed
- Some commands may lead to a *submenu*

Components of a Menu System

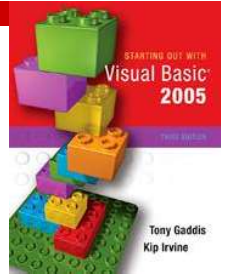


- Commands can be performed by
 - A key or key combination called a *shortcut key*
 - A single letter and the Alt key called an *access key*
- Menu must be open to use an access key
- Shortcut key works when menu is not displayed

Components of a Menu System

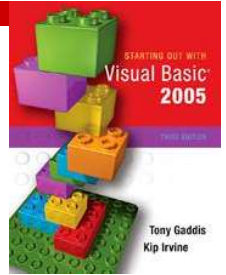


- A *disabled menu command* can't be selected at present and shows as a light color (grayed out)
- A *checked menu command* toggles between the checked (if on) and unchecked (if off) states
- A *separator bar* helps group similar commands



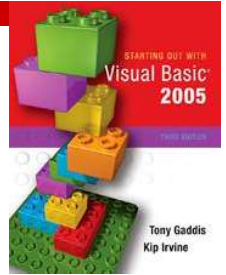
MainMenu Control

- Add to form with double-click on MainMenu control in *Toolbox Menus & Toolbars* section
- The *MainMenu control* is displayed in the component tray (bottom of Design window)
- May have many MenuItem objects with the following key properties:
 - MenuItem name used by VB to identify it
 - MenuItem text displayed to the user
 - Actions in the form of a MenuItem click event
 - Submenu
 - Separator bar



MenuItem Object Names

- Should begin with *mnu*
- Then by convention are spelled, specifying their hierarchical position:
 - `mnuFile`
 - `mnuFileSave`
 - `mnuFilePrint`



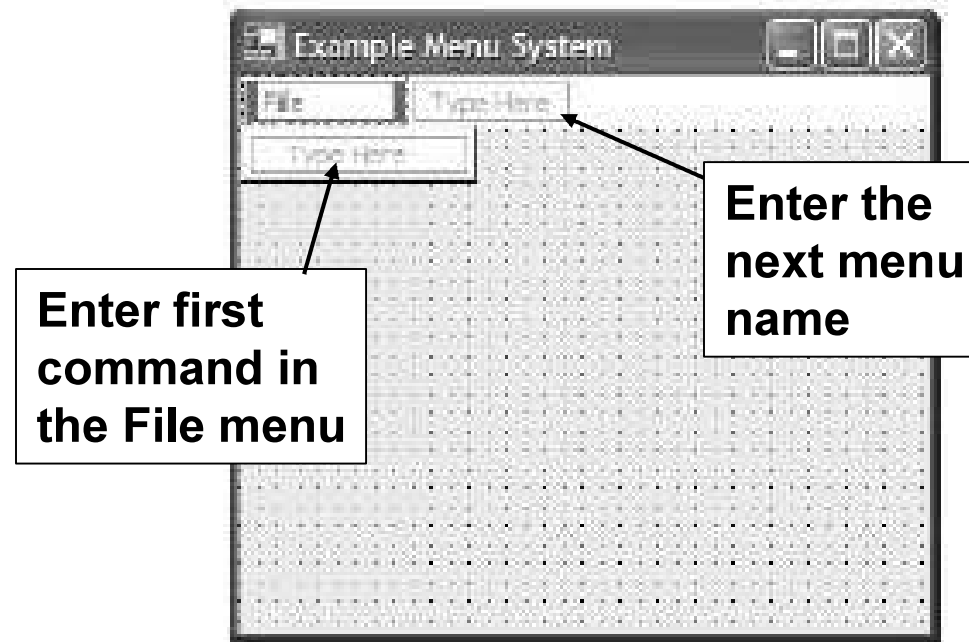
MenuItem Text Properties

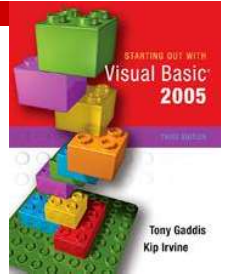
- The text property holds the item description displayed to the user
- If an access key is assigned, that letter must be preceded with an ampersand

<u>Object Name</u>	<u>Text Property</u>	<u>Access Key</u>
mnuFile	&File	F
mnuFileSave	&Save	S
mnuFileExit	E&xit	X

Menu Designer

- The *Menu Designer* allows menu creation by filling in a box with the menu text:





Shortcut Keys

- Keyboard based shortcuts that execute menu commands without using the menu system
- For example, *ctrl-c* to Copy to the clipboard
- These are set via the *Shortcut* property of each menu item
- A shortcut is displayed to the user only if the *ShowShortcut* property is set to true

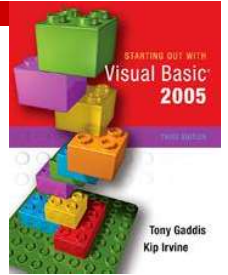


Disabled MenuItem Objects

- A menu item is *grayed out* (disabled) with the *Enabled* property
- Paste option is initially disabled and only enabled after something is cut or copied
 - Code initially disables the Paste option

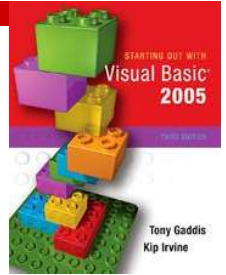
```
mnuEditPaste.Enabled = False
```
 - Following a cut or copy, Paste is enabled

```
mnuEditPaste.Enabled = True
```



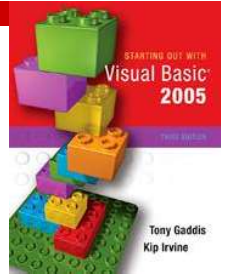
Adding Separator Bars

- Right-click menu item, select *Insert Separator*
 - Separator inserted above the menu item
- Or create a menu item with one hyphen (-) as the text property



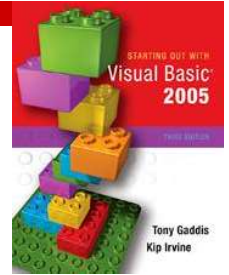
Submenus

- When selecting a menu item in the designer, a *Type Here* box appears to its right
 - Begin a submenu by setting up this menu item
- If a menu item has a submenu, a solid right-pointing arrow will be shown for this item



Inserting, Deleting, & Rearranging

- To insert a new menu item within the list
 - Right-click the item to follow the new one
 - Choose *Insert New* from the shortcut menu
- Use Menu Designer to add new menu items at the end by entering the text to appear
- To remove a menu item
 - Right-click on the item
 - Choose *Delete* from the shortcut menu
- The Menu Designer can rearrange items using a click and drag approach



MenuItem Click Event Procedures

- Menus and submenus require no code
- Commands require a click event procedure
 - Double click on the menu item
 - Event procedure created in the code window
 - Programmer supplies the code to execute
- Double click the MenuItem object named `mnuFileExit` to create the following

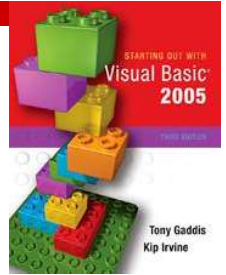
```
Private Sub mnuFileExit_Click(ByVal sender as System.Object, _  
    ByVal e as System.EventArgs) Handles mnuFileExit.Click
```

```
    Me.Close()
```

Programmer supplied code

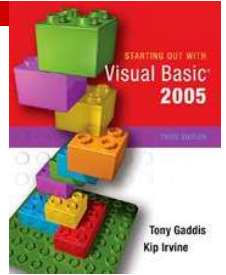
```
End Sub
```

Click event procedure created by VB



Standard Menu Items

- In general follow the conventions that most application menu systems use
 - *File* is leftmost item with access key *Alt-F*
 - File item has *Exit* command, access key *Alt-X*
 - *Help* is the rightmost item
 - Help menu has an *About* command
- Tutorial 7-4 demonstrates how to create a menu system



Context Menus

- A pop-up menu that appears on a right-click
- *Context menus* are designed for a particular control or set of controls
- To set up a Context Menu:
 - Double-click *ContextMenu* control in the *ToolBox* to add it to the component tray
 - Build menu system using Menu Designer
 - Build *Click* event procedures as needed
 - Use *ContextMenu* property of form controls to link desired control(s) to the menu