Menus, Sub Menus, and Common Dialogs

Creating Menus

Menus are commonly found in windows applications. To create menus in VB, you simply add the MenuStrip control to the form. You can also easily create sub menus, separators (-), and assign access keys to any menu items.

Create a new application and add a text box to it. Set multiline property of the textbox to yes and add a vertical scroll bar. Create a menu by choosing the menuStrip control from the toolbox. Click on the smart tag on the control and choose “Insert standard items” to add all the standard menu items. However, in this exercise, we will not write event handler for any of the items except “File – Exit”. Now add two more menu items to the top-level menu so it looks like the following:

To facilitate dealing with font and color, VB provides the FontDialog and ColorDialog controls. Add a FontDialog and a ColorDialog control to the form. The FontDialog and ColorDialog controls do not physically appear when the program runs but just provide the capabilities for us to easily handle fonts and colors. For the Color menu, add two items: BackColor and ForeColor. Add a separator bar between the two items. The following screenshot shows how.
Besides menuStrip control, VB also has another control named ContextMenuStrip control. ContextMenuStrip allows our application to display a shortcut menu when the user clicks the right mouse button over a control or area of the form. Controls and Form have a ContextMenuStrip property that binds the ContextMenuStrip class to the control or form that displays the shortcut menu. Practice using ContextMenuStrip to create ForeColor and BackColor items for the textbox. You also need to add the Click event of the new items to the Handles list of corresponding event handlers.

**String Manipulation**

**Useful String Properties & Methods**

String is a very commonly used variable type in VB. For any String variable, some properties and methods are useful:

Property: `.Length` ➔ is the number of characters in the string  
Methods: `.ToUpper` ➔ is the string with all its letters capitalized  
           `.ToLower` ➔ is the string with all its letters in lower case  
           `.Trim` ➔ is the string with all spaces removed from the front and back of the string  
           `.IndexOf` ➔ check to see if a string is a substring of another string  
           `.Substring` ➔ return a substring extracted from another string

If we have

Dim str, str2, str3 as String  
str = “Welcome to”  
str2 = “ OPIM 3220 “  
str3 = “come”

then

str.Length is 10  
str.ToUpper is “WELCOME TO”  
str2.ToLower is “ opim 3220 “  
str & str2.Trim is “Welcome toOPIM 3220”  
str.IndexOf(str3) is 3

Note: str.IndexOf(str2) is –1 since str2 is not a substring of str. In VB the position of a character in a string starts from 0.

str.Substring(m, n) is the substring of str consisting of n characters beginning with the character in position m of str.

For example, str.Substring(2,3) is “lco”

**Practice:**

Use the above string properties and methods to create a scrolling text application. Use the following as the scrolling text: "Welcome to OPIM 3220, Spring 2016!!!". Set the timer’s interval to 200.