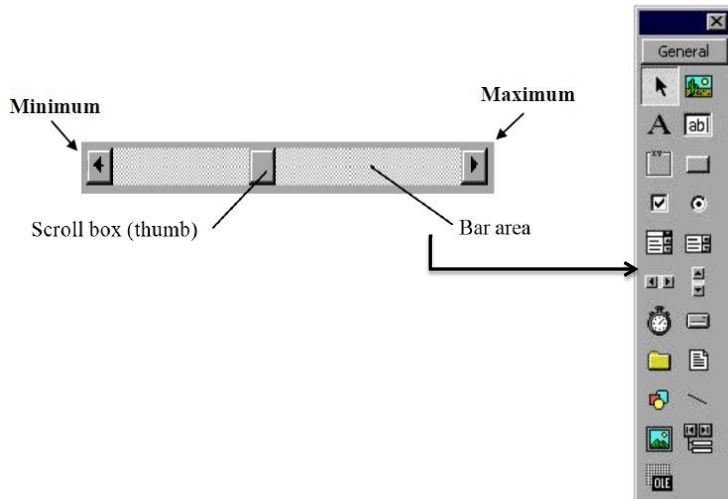Horizontal and Vertical Scroll Bar control(HScrollBar & VScrollBar)

Horizontal and vertical scroll bars are widely used in Windows applications.

Scroll bars provide an intuitive way to move through a list of information.

Both type of scroll bars are comprised of three areas that can be clicked, or dragged, to change the scroll bar value.
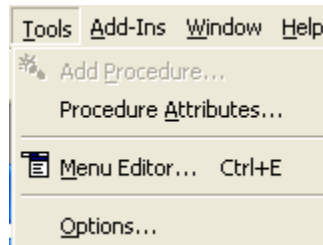

Example 1: Using Scroll Bar, change the TextBox font size and forecolorSolution:

Private Sub Form_Load()

HScroll1.Min = 1
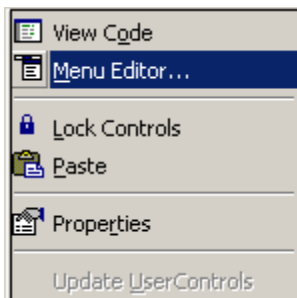
HScroll1.Max = 100

End Sub

HScroll1:

Private Sub HScroll1_Change()

Text1.FontSize = HScroll1.Value

Text1.ForeColor = RGB(Rnd * 255, Rnd * 255, Rnd * 255)
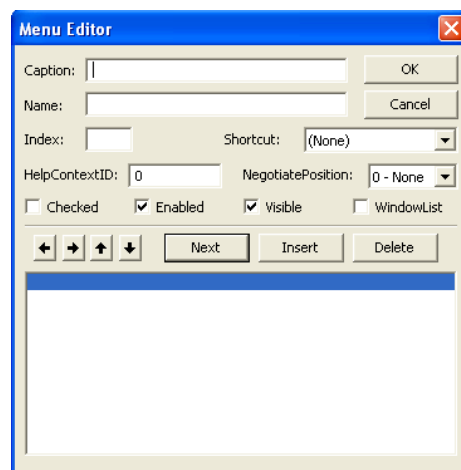
End Sub

<p align="center">Menus</p>

Windows applications provide groups of related commands in menus. These commands depend on theapplication, but some- such as Open and Save – are frequently found in applications. Visual Basic provides an easy way to create menus with the modal Menu Editor dialog. The dialog is displayed when Menu Editor is selected from the tools menu.

The menu editor command is grayed unless the form is visible. The menu dialog can also be displayedby right-clicking the form and selecting menu editor.

The menu editor dialog, shown in figure below, contains the textboxes Caption and Name. The value entered in the Caption Textbox is the menu name the user sees. The value entered in the Name Textbox is the variable name the programmer uses. Every menu must have a Caption
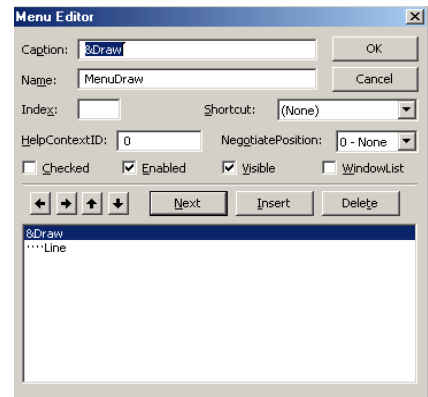

and a Name. Menus are like other controls in that they have properties and events. The menu editor is a way of setting select properties for a menu. Once a menu is created, its properties can be viewed in the properties window and its events in the code window. The programmer can create menu control

arrays. The Index Textbox specifies the menu's index in the control array. Menus that are not top- level menus can have shortcut keys (combinations of Ctrl, Shift and letter keys). Shortcut keys are specified using the Shortcut ComboBox. All shortcut keys listed in the shortcut ComboBox are predefined by visual basic. Programmers may not define their own.

The menu editor dialog also several provides several CheckBoxes for controlling the appearance of menus. The checked checkbox is unchecked (False) by default and allows the programmer the option of creating a checked menu item (a menu item that acts as a toggle and displays a check mark when selected). We discuss the menu editor buttons momentarily.

In our next example, we create two top-level menus, Draw and Exit, using the menu editor. Building the menus does not require us to write any code. Creating and designing menus is an excellentexample of visual programming (the ability to create windows GUIs writing only minimal code).

The following figure displays a GUI containing two menus, Draw and Exit – each of which is createdusing menu editor. Note that the separator bars are created by placing a hyphen, -, in the caption textbox.

The menu editor dialog shown in figure below, corresponding to the program of example. The next button allows the next menu or menu item to be edited. Insert allows a menu or menu item to be inserted between two existing menus or menu items. Delete remove the selected menu or menu item. The up and down arrow buttons perform insertion in the same manner as insert.

Notice that the top-level menus appear in the left column. Any submenus or menu items are indented (using the second arrow button). Submenu menu items are indented beneath their respective menus (using the second arrow button).

```
Private Sub DrawLine _click ()Line
(25,25)-(75,50),vbYallowEnd Sub
Private Sub Drawcircle _click ()Circle

(25,15), 5,vbGreen
End Sub

Private Sub MenuClose _click ()END
End Sub
```