Unit 3

Container

- Container are those controls which contain other controls.
- for example, a set of radio buttons, checkboxes,button etc
- Container
  - Panel
  - Groupbox
  - Tab Control
  - Splitter

Panel

- Panels are those controls which contain other controls.
- You use panels to group other controls, usually to divide a form into regions by function.
- If the Panel's Enabled property is set to False then the controls which the Panel contains are also disabled.
- Properties :
  - BorderStyle property : The default value of the BorderStyle property is set to None.
  - AutoScroll property : Default value is set to False.
    - Set it to True if you want a scrollbar with the Panel.

Group Box

- Group box are similar to panel control which contains other controls.
- You use Groupbox to group other controls, usually to divide a form into regions by function.
- Properties :
  - Text : You set the group box's caption with the **Text** property.
- Code

Panel vs. Groupbox

Panel	Group Box
Panels cannot display captions .	Groupbox can display Caption using Text Property.
Panels can have scrollbars if AutoScroll property is true.	Groupbox can't have scrollbars.
Panels are based on the ScrollableControl class.	The Groupbox class is based on the Control class.

Tab Control

- Tab controls work much like the tabs in a set of folders.
- Tab controls let you divide your display into overlapping tab pages, and each page can contain other controls.
- The central property of the **TabControl** is **TabPages.**
  - contains the individual tab pages in the control, each of which is a **TabPage** object.
  - When a tab is clicked, it displays its page and causes a **Click** event for that **TabPage** object.
  - You can add new tab pages with the **TabPages** collection's **Add** method, and remove them with the **Remove** method.

Tab Control Properties

Property	Means
Alignment	Gets/sets where the tabs appear (top, left, etc.).
Appearance	Gets/sets the appearance of tabs in a tab control.
DisplayRectangle	Gets the bounding rectangle of the tab pages.
Multiline	Gets/sets whether the tab control can show more than one row of tabs.
RowCount	Gets the number of rows in the tab strip.
SelectedIndex	Gets/sets selected tab page's index.
SelectedTab	Gets/sets the selected tab page.
ShowToolTips	Gets/sets whether a tab's tooltip can be displayed.
TabCount	Gets the number of tabs.

Tab Control Event

Event	Means
SelectedIndexChanged	Occurs when the **SelectedIndex** property is changed.

- Code

Timer

- Timers are also very useful controls, because they let you create periodic events.
- Timers are components that cause periodic **Tick** events that you can use to execute code at specific intervals.

- Code

Timer Properties, Methods and Event

Property	Means
Enabled	Gets/sets whether the timer is running.
Interval	Gets/sets the time (in milliseconds) between timer ticks.

Method	Means
Start	Starts the timer.
Stop	Stops the timer.

Event	Means
Tick	Occurs when the timer interval has elapsed (and the timer is enabled).

Menu

- Menus are those controls that allow the user to make selections and also hide away those selections when they're not needed and saving space in Windows applications.
- the **MainMenu** control represents the container for the menu structure of a form;.
- Menus are made up of **MenuItem** objects that represent the individual parts of a menu
  - menu items can be a parent menu or a menu item in a menu.

Code

Menu Items

- Menus like File or Edit and the actual items in such menus are supported with the **MenuItem** class.

Properties	Meaning
Checked	To display a checkmark next to a menu item
Shortcut	keyboard combination (like Ctrl+O)
ShowShortcut	Set showShort cut property **True** to display that key combination in the menu item's caption.
Text	To set the caption of a menu or menu item

Context Menu

- Another popular type of menus is *context menus*.
- You use **ContextMenu** controls to give users access to frequently used menu commands, and bring them up by right-clicking another control.
- You associate context menus with other controls by setting the control's **ContextMenu** property to the **ContextMenu** control.
- Context menus are those handy menus that pop up over controls, usually when you right-click them. They're called context menus because they appear over specific controls

Contex Menu Properties

Property	Means
IsParent	Holds a value that is **True** if this menu contains any menu items.
MdiListItem	Holds a value that is **True** if the **MenuItem** is used to display a list of MDI child windows.
MenuItems	Holds the collection of **MenuItem** objects for this menu.

MDI

- Multiple Document Interface (MDI) programs can display multiple child windows inside them.
- Any windows can become an MDI parent, if you set the IsMdiContainer property to True.
- **IsMdiContainer = True**

- **Code**

MDI

- To arranging MDI child windows:
  - To automatically arrange MDI child windows, you can use the **LayoutMdi** method.
  - **MdiLayout** values that defines the layout of MDI child forms:
    - **ArrangeIcons—** All MDI child icons (which are displayed when you minimize an MDI child window) are arranged.
    - **Cascade—** All MDI child windows are cascaded.
    - **TileHorizontal—** All MDI child windows are tiled horizontally.
    - **TileVertical—** All MDI child windows are tiled vertically.

MDI vs. SDI

MDI	SDI
Multiple Document Interface (MDI) programs can display multiple child windows inside them.	Single document interface (SDI) applications, which can manipulate only one document at a time.
Visual Studio Environment is an example of Multiple Document Interface (MDI)	Notepad is an example of an SDI application, opening a document closes any previously opened document.
MDI is when your application consist of an MDI parent-form that contains all the other window/forms that the app consists of.	SDI is stand-alone, ordinary windows/forms that exists independently of each other.

Dialog Box

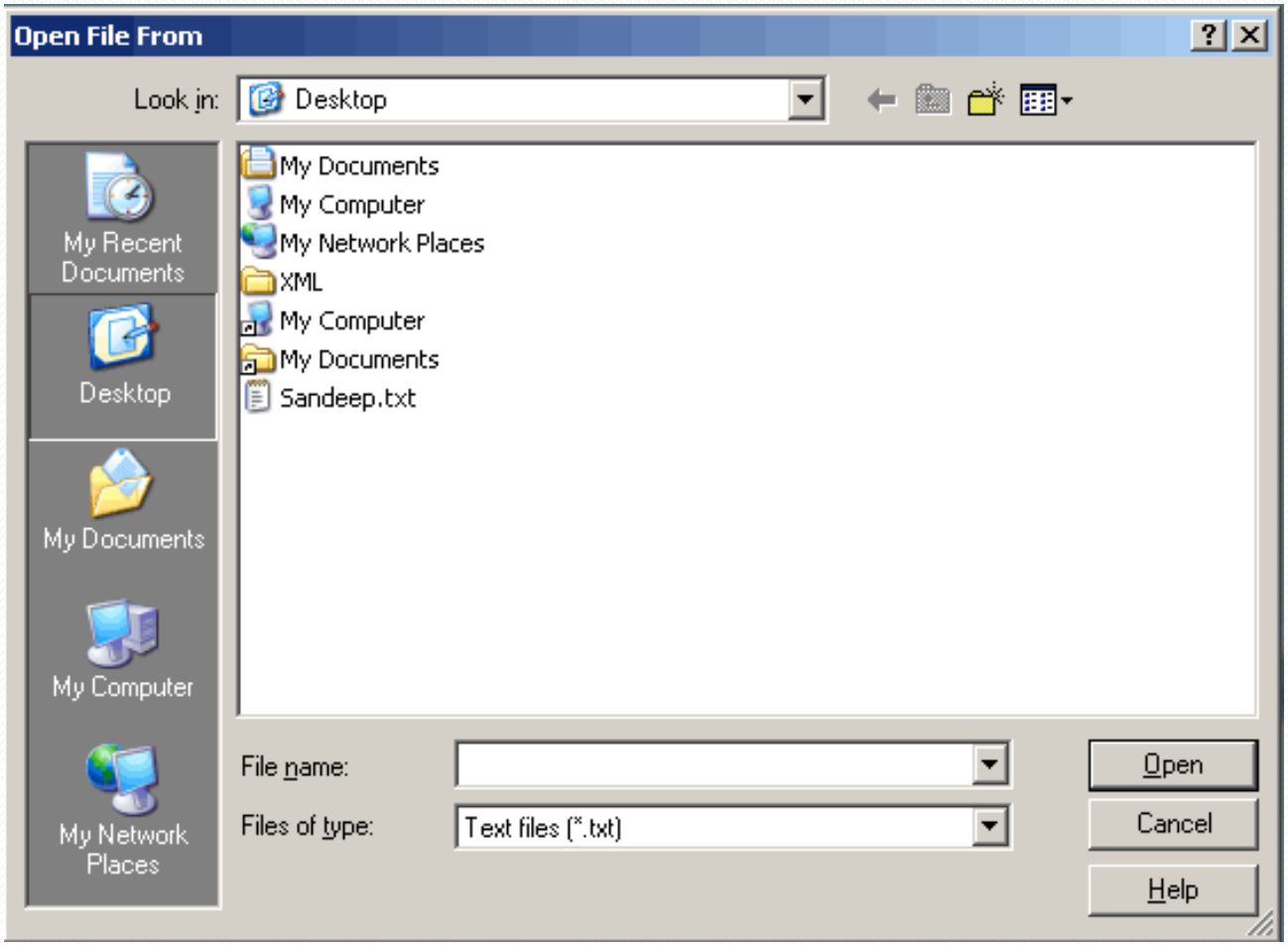
- Visual Basic .NET comes with built-in dialog boxes which allow us to create our own File Open, File Save, Font and Color dialogs.
- **Dialog Box :**
  - OpenFileDialog
  - SaveFileDialog
  - ColorDialog
  - FontDialog
  - PrintDialog,
  - PrintPreviewDialog
  - PageSetupDialog.

Dialog Box

- To make a dialog box visible at run time we use the dialog box's *ShowDialog* method.
- The return values of all the above said dialog boxes which will determine which selection a user makes are:
  - Abort
  - Cancel
  - Ignore
  - No
  - None
  - OK
  - Return
  - Retry
  - Yes.

OpenFileDialog Box

- Open File Dialog's are supported by the OpenFileDialog class .
- They allow us to select a file to be opened.



OpenFileDialog Box

Code

Properties	Mean
AddExtension	Gets/Sets if the dialog box adds extension to file names if the user doesn't supply the extension.
CheckFileEixsts	Checks whether the specified file exists before returning from the dialog.
CheckPathExists	Checks whether the specified path exists before returning from the dialog.
DefaultExt	Allows you to set the default file extension
FileName	Gets/Sets file name selected in the file dialog box
FileNames	Gets the file names of all selected files.
Filter	Gets/Sets the current file name filter string, which sets the choices that appear in the "Files of Type" box.

OpenFileDialog Box Properties

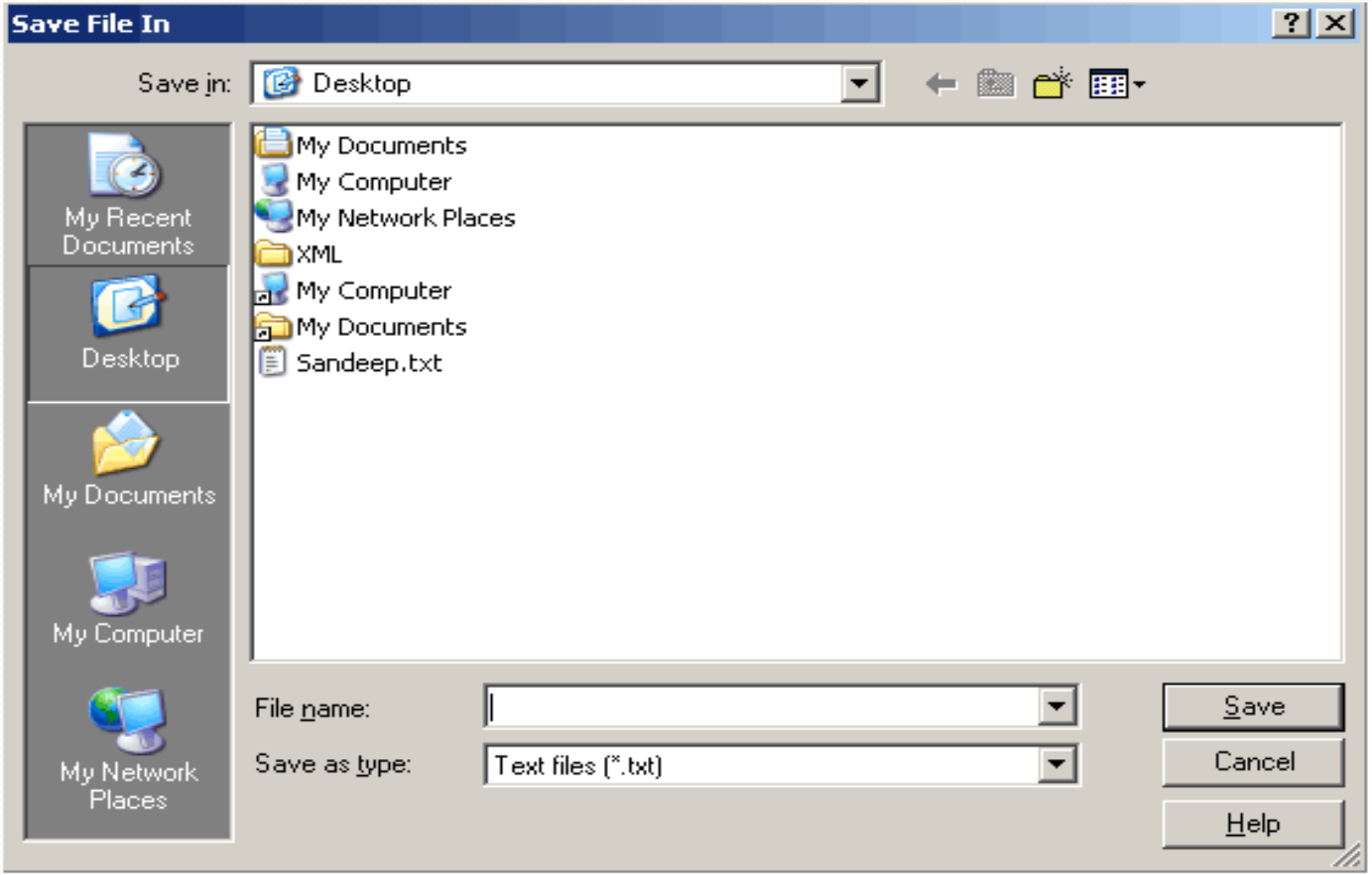
Properties	Mean
FilterIndex	Gets/Sets the index of the filter selected in the file dialog box.
InitialDirectory	This property allows to set the initial directory which should open when you use the OpenFileDialog.
MultiSelect	This property when set to True allows to select multiple file extensions.
ShowHelp	Gets/Sets whether the help button should be displayed.
Title	This property allows to set a title for the file dialog box.

SaveFileDialog

- Save File Dialog's are supported by the SaveFileDialog class and they allow us to save the file in a specified location.
- Properties of the Save File Dialog are the same as that of the Open File Dialog.
- Notable property of Save File dialog is the OverwritePromopt property which displays a warning if we choose to save to a name that already exists.

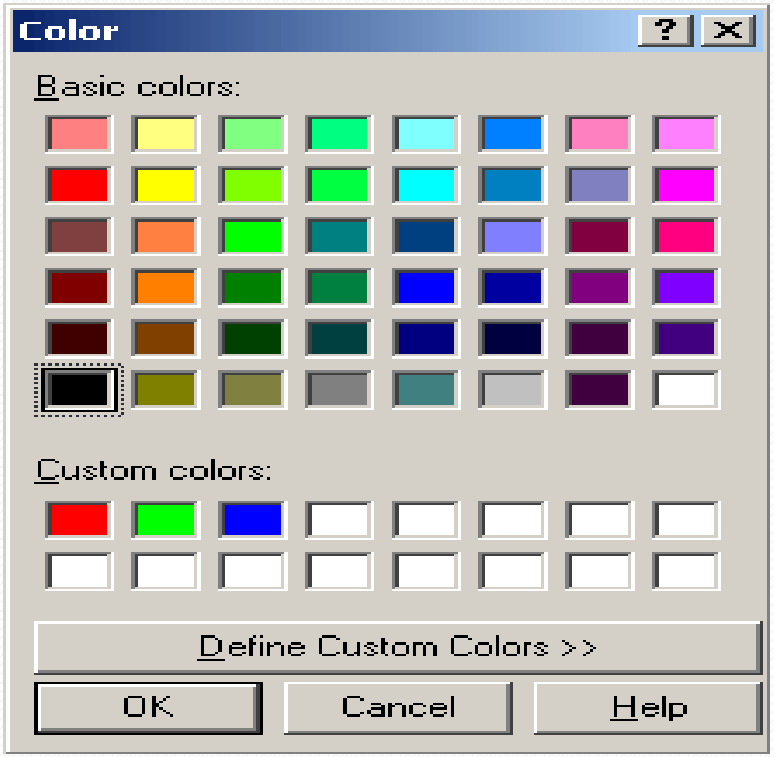
SaveFileDialog

Code



ColorDialog

- Color Dialog's are supported by the ColorDialog Class and they allow us to select a color. The image below displays a color dialog.



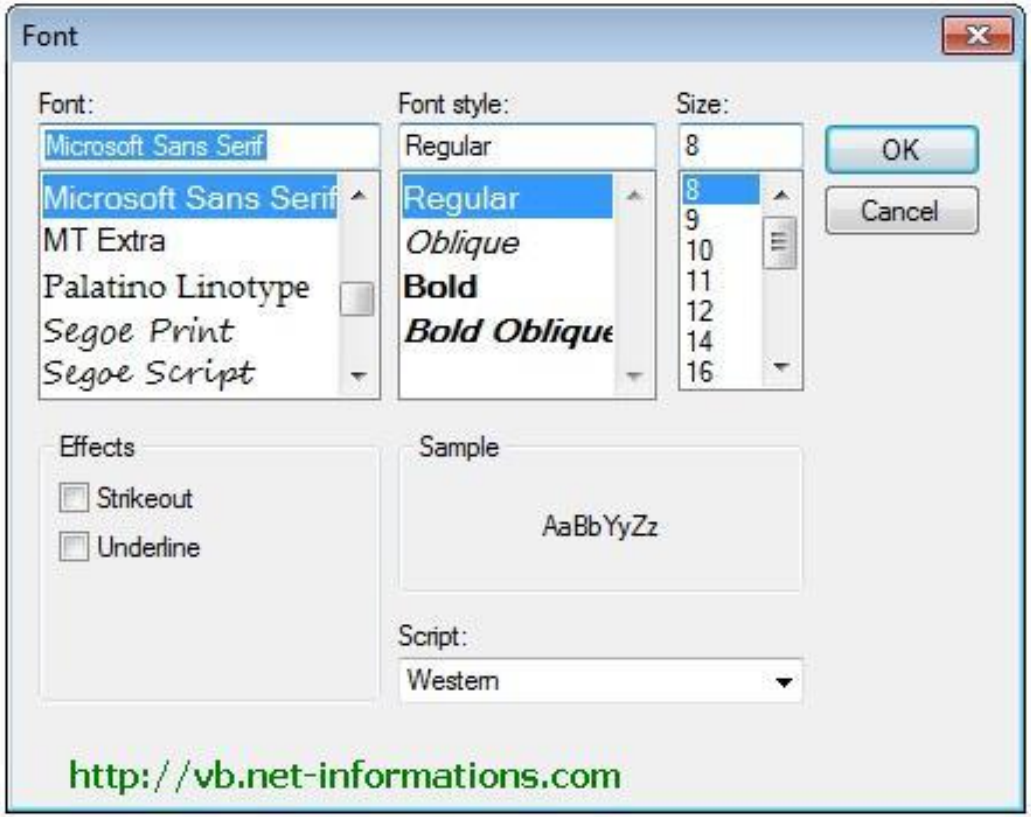
ColorDialog box properties

Code

Property	Meaning
AllowFullOpen:	Gets/Sets whether the user can use the dialog box to define custom colors.
AnyColor:	Gets/Sets whether thedialog box displays all the available colors in the set of basic colors.
Color:	Gets/Sets the color selected by the user.
CustomColors:	Gets/Sets the set of custom colors shown in the dialog box.
ShowHelp:	Gets/Sets whether the dialog box displays a help button.
SolidColorOnly:	Gets/Sets whether the dialog box will restrict users to selecting solid colors only.

FontDialog Box

- The Font dialog box lets the user choose attributes for a logical font,
  - font family, font style, point size, effects , and a script .



FontDialog Box

Code

Properties	Meaning
AllowVerticalFonts:	Gets/Sets whether the dialog box displays both vertical and horizontal fonts or only horizontal fonts. Color: Gets/Sets selected font color.
Font:	Gets/Sets the selected font.
FontMustExist:	Gets/Sets whether the dialog box specifies an error condition if the user attempts to select a font or size that doesn't exist.
MaxSize:	Gets/Sets the maximum point size the user can select.
MinSize:	Gets/Sets the mainimum point size the user can select.
ShowColors:	Gets/Sets whether the dialog box displays the color choice.
ShowEffects	Gets/Sets whether the dialog box contains controls that allow the user to specify to specify strikethrough, underline and text color options.
ShowHelp:	Gets/Sets whether the dialog box displays a help button.

InputBox and MsgBox

- Message Boxes : Used to display message or some other helpful information.
- Input boxes : are great for getting user input.

MsgBox

Code

The Message Box has these parts:

MsgBox(prompt[, buttons] [, title] [, helpfile, context])

- *Prompt* is the message of the Message Box
- *Buttons* is the only tricky part of the message box.
- *Title* is the title of the message box.
- *Helpfile* and *Context* is where at in a helpfile the user can find help on that message box.

MsgBox

- Message box can look like 4 icons,
  - Critical, Question, Exclamation, and Information.
- It can have a variety of buttons for the user to click on which include:
  - OK, OK Cancel, Abort Retry Ignore, Yes No Cancel, Yes No, and Retry Cancel.
  - You can also select which of the buttons is the default.

Value of Buttons

Constant	Value	Description
OKOnly	0	Shows OK button only.
OKCancel	1	Shows OK and Cancel buttons.
AbortRetryIgnore	2	Shows Abort, Retry, and Ignore buttons.
YesNoCancel	3	Shows Yes, No, and Cancel buttons.
YesNo	4	Shows Yes and No buttons.
RetryCancel	5	Shows Retry and Cancel buttons.
Critical	16	Shows Critical Message icon.
Question	32	Shows Warning Query icon.
Exclamation	48	Shows Warning Message icon.
Information	64	Shows Information Message icon.

InputBox

- An InputBox( ) function will display a message box where the user can enter a value or a message in the form of text. The format is
- If you need a password, name, or other info., this is a very easy way of getting it.

InputBox

- The syntax for an InputBox is:

  myMessage=InputBox(Prompt, Title, default_text, *Xpos, ypos*)

- myMessage : is a string variable which accept the message input by the user.
- *Prompt :* is the message that the input box has.
  - This tells the user what the program wants them to input. (ex: "Please enter your password.").
- *Title :* is the title of the input box, which is the part that appears in the blue. (ex: "Password").

InputBox

- *Default_text :* is an option you may want to use.
  - It puts text in the text box part of the InputBox as a default.
- *Xpos* and *ypos* : are the horizontal and vertical positions of the input box.
  - If you leave these out, the default puts the input box horizontally centered and about one third of the way down the screen.
- *Helpfile* and *context :* tell where in a help file the user can get help on that Input Box.