



AMIT ACADEMY

for Computer Education

**Subject – Desktop Application
Development (VB.NET)**

Unit-3

**Basic Tools for Desktop Application
Development**

**AMIT ACADEMY FOR COMPUTER EDUCATION
Nr Vardayini Mataji Temple, Rupal,
Gandhinagar-382630**

Email – amitacademy1117@gmail.com

Mobile No – 8460467193

YouTube Link –

<https://www.youtube.com/@ankitramijoinar>

Instagram Link –

<https://www.instagram.com/amitacademy17/>

Facebook Link –

<https://www.facebook.com/aramitacademy/>

Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193

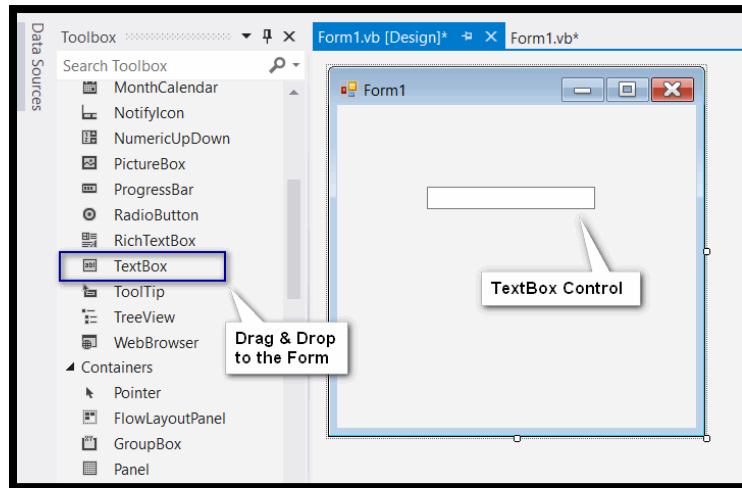


Data Entry Tools

- The tools using which data can be written into the computer software are called data entry tools.

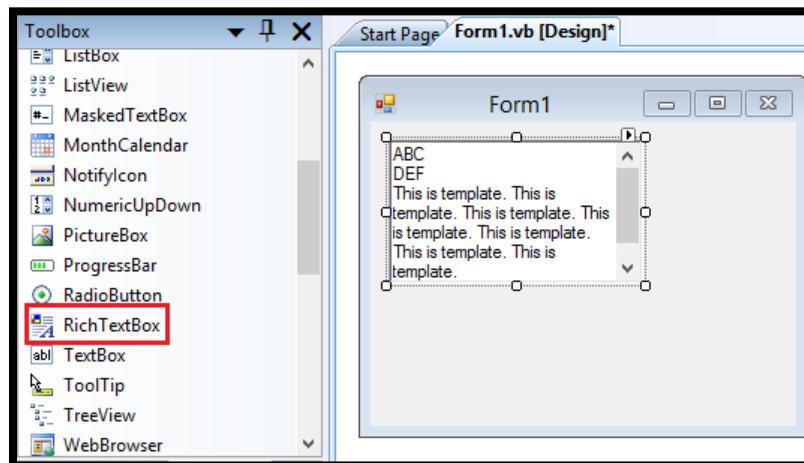
➤ Textbox

- Text box controls allow entering text on a form at runtime



➤ Rich Textbox

- The Rich Textbox control allows the user to display, enter, and edit text while also providing more advanced formatting features than the conventional Textbox control

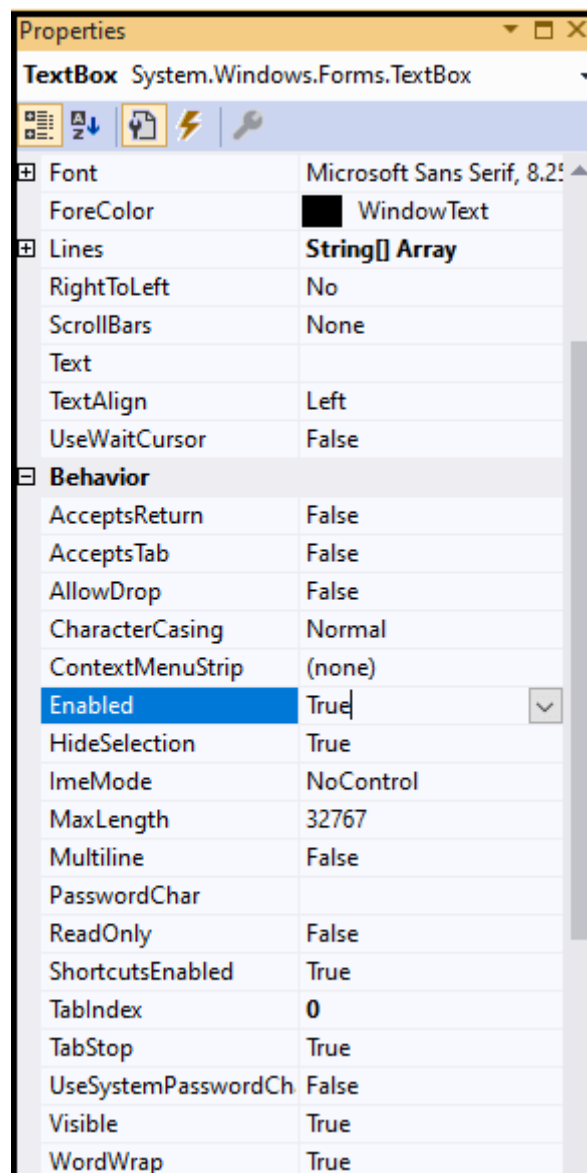




⇒ Textbox Control in VB.net

A Textbox control is used to display, accept the text from the user as an input, or a single line of text on a VB.NET Windows form at runtime. Furthermore, we can add multiple text and scroll bars in textbox control.

VB.NET Textbox Properties





Textbox Control Properties in VB.net

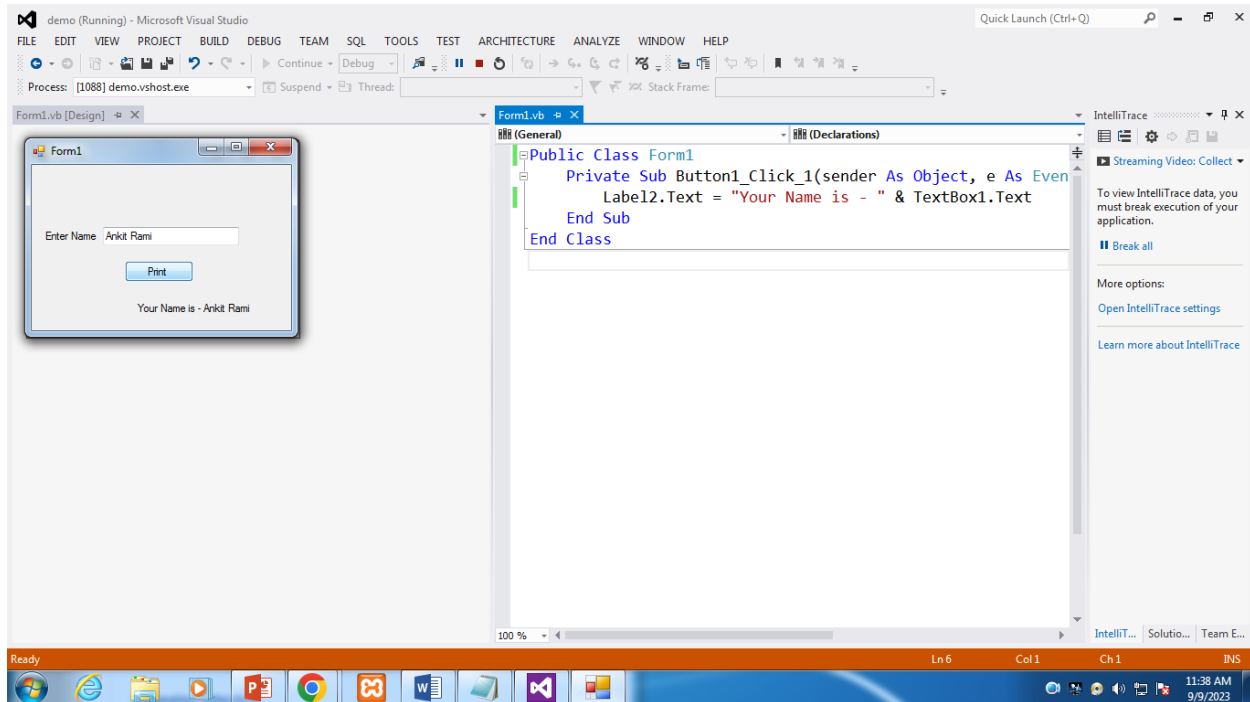
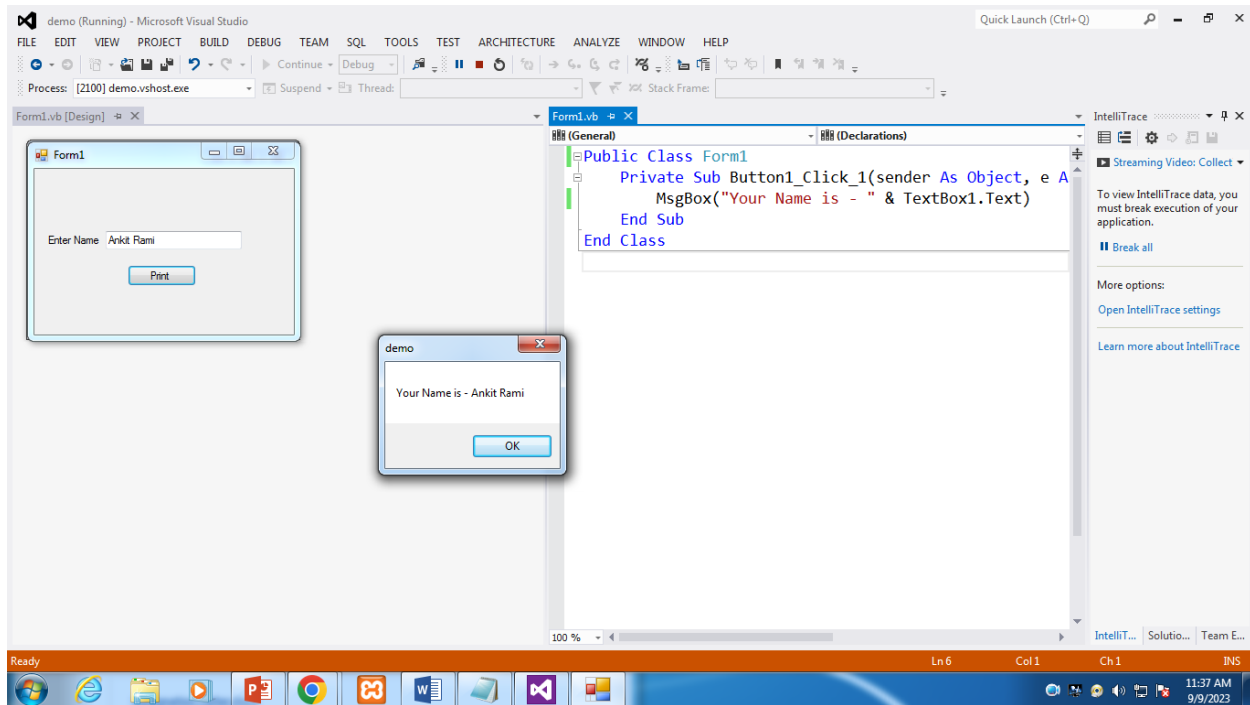
Properties	Description
AutoCompleteMode	It is used to get or set a value that indicates how the automatic completion works for the textbox control.
Font	It is used to set the font style of the text displayed on a Windows form.
Lines	It is used to set the number of lines in a TextBox control.
CharacterCasing	It is used to get or set a value representing whether the TextBox control can modify the character's case as they typed.
Multiline	It is used to enter more than one line in a TextBox control, by changing the Multiline property value from False to True.
AcceptsReturn	It is used to get or set a value that indicates whether pressing the enter button in a multiline textbox; it creates a new line of text in control.
PasswordChar	It is used to set the password character that can be a mask in a single line of a TextBox control.
PreferredHeight	It is used to set the preferred height of the textbox control in the window form.
ScrollBars	It is used to display a scrollbar on a multiline textbox by setting a value for a Textbox control.
Text	It is used to get or set the text associated with the textbox control.
Visible	The Visible property sets a value that indicates whether the textbox should be displayed on a Windows Form.
WordWrap	The WordWrap properties validate whether the multiline Textbox control automatically wraps words to the beginning of the next line when necessary.

Textbox Control Events in VB.net

Events	Description
Click	When a textbox is clicked, a click event is called in the textbox control.
CausesValidationChanged	It occurs in the TextBox Control when the value of CauseValidation property is changed.
AcceptTabsChanged	It is found in the TextBox control when the property value of the AcceptTab is changed.
BackColorChanged	It is found in the TextBox Control when the property value of the BackColor is changed.
BorderStyleChanged	It is found in the TextBox Control when the value of the BorderStyle is changed.
ControlAdded	It is found when the new control is added to the Control.ControlCollection.
CursorChanged	It is found in TextBox, when the textbox control is removed from the Control.ControlCollection.
FontChanged	It occurs when the property of the Font is changed.
GetFocus	It is found in TextBox control to get the focus.
MouseClick	A MouseClick event occurs when the mouse clicks the control.
MultilineChanged	It is found in a textbox control when the value of multiline changes.



Ex- Textbox Control

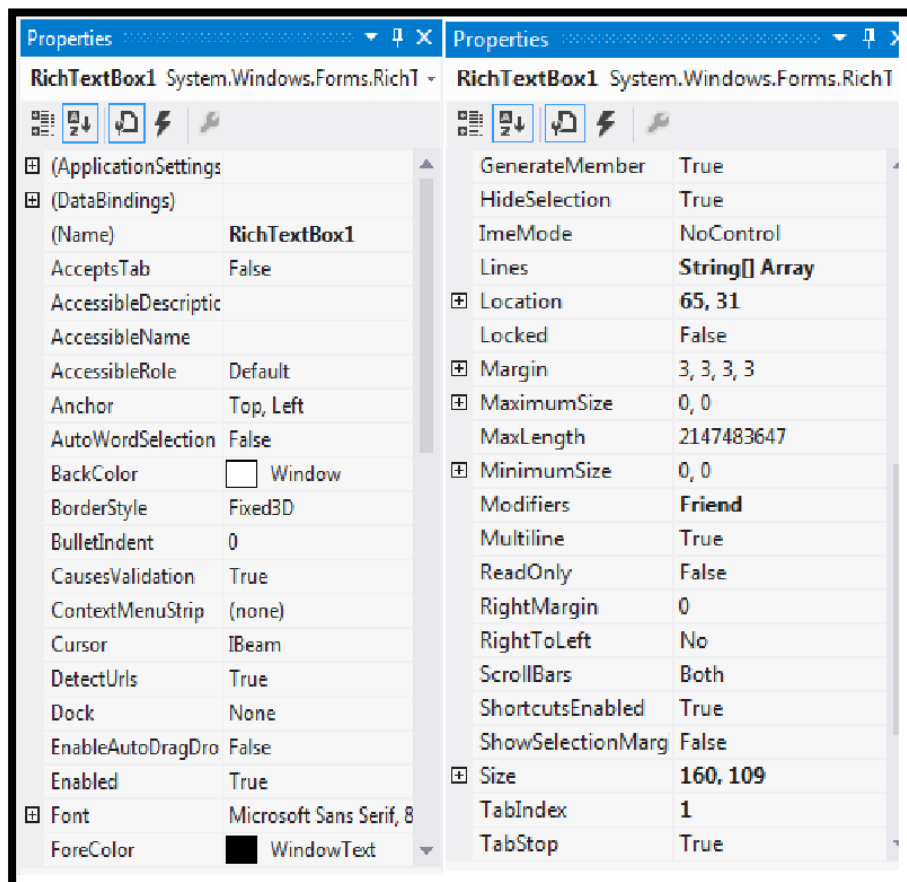




⇒ RichTextbox Control in VB.net

- The RichTextbox control allows the user to display, enter, and edit text while also providing more advanced formatting features.
- The RichTextbox control opens and saves files in both the RTF format and regular ASCII text format. You can use methods of the control (LoadFile and SaveFile) to directly read and write files, or use properties of the control such as SelRTF and TextRTF in conjunction with Visual Basic's file input/output statements.
- The RichTextbox control supports almost all of the properties, events and methods used with the standard TextBox control, such as MaxLength, MultiLine, ScrollBars, SelLength, SelStart, and SelText. Applications that already use TextBox controls can easily be adapted to make use of RichTextbox controls. However, the RichTextbox control doesn't have the same 64K character capacity limit of the conventional TextBox control.

RichTextbox Control Properties in VB.net





Methods of the RichTextBox Control:

Method

Description

LoadFile

Loads an .rtf file or text file into a RichTextBox control.

Syntax:

RichTextBox1.LoadFile *pathname*, *filetype*

Where *pathname* is a string defining the path and filename of the file to load into the control and *filetype* is an integer or constant that specifies the type of file loaded. Valid values for *filetype* are 0 (or the constant **rtfRTF**; default, specifies a valid .rtf file), or 1 (constant **rtfText**, specifies any text file).

When loading a file with the **LoadFile** method, the contents of the loaded file replaces the entire contents of the **RichTextBox** control. This will cause the values of the **Text** and **RTFText** properties to change.

SaveFile

Saves the contents of a RichTextBox control to a file.

Syntax:

RichTextBox1.SaveFile *pathname*, *filetype*

Where *pathname* is a string defining the path and filename of the file to receive the contents of the control and *filetype* is an integer or constant that specifies the type of file to be saved. Valid values for *filetype* are 0 (or the constant **rtfRTF**; default, saves contents as a .rtf file), or 1 (constant **rtfText**, saves contents as a text file).

Find

Searches the text in a **RichTextBox** control for a given string.

Syntax

object.Find(*string*, *start*, *end*, *options*)

The **Find** method syntax has these parts:

Part	Description
<i>object</i>	Required. The RTB control.
<i>string</i>	Required. A string expression you want to find in the control.
<i>start</i>	Optional. An integer character index that determines where to begin the search. Each character in the control has an integer index that uniquely identifies it. The first character of text in the control has an index of 0.
<i>end</i>	Optional. An integer character index that determines where to end the search.
<i>options</i>	Optional. One or more constants used to specify optional features, as described in Settings.

Settings

The setting for *options* can include:

Constant	Value	Description
rtfWholeWord	2	Determines if a match is based on a whole word or a fragment of a word.
rtfMatchCase	4	Determines if a match is based on the case of the specified string as well as the text of the string.
rtfNoHighlight	8	Determines if a match appears highlighted in the RichTextBox control.

You can combine multiple options by using the **Or** operator.

Remarks

If the text searched for is found, the **Find** method highlights the specified text and returns the index of the first character highlighted. If the specified text is not found, the **Find** method returns -1.

If you use the **Find** method without the **rtfNoHighlight** option while the **HideSelection** property is **True** and the **RichTextBox** control does not have the focus, the control still highlights the found text. Subsequent uses of the **Find** method will search only for the highlighted text until the insertion point moves.

The search behavior of the **Find** method varies based on the combination of values specified for the *start* and *end* arguments. This table describes the possible behaviors:

Start	End	Search Behavior
Specified	Specified	Searches from the specified start location to the specified end location.
Specified	Omitted	Searches from the specified start location to the end of the text in the control.
Omitted	Specified	Searches from the current insertion point to the specified end location.
Omitted	Omitted	Searches the current selection if text is selected or the entire contents of the control if no text is selected.

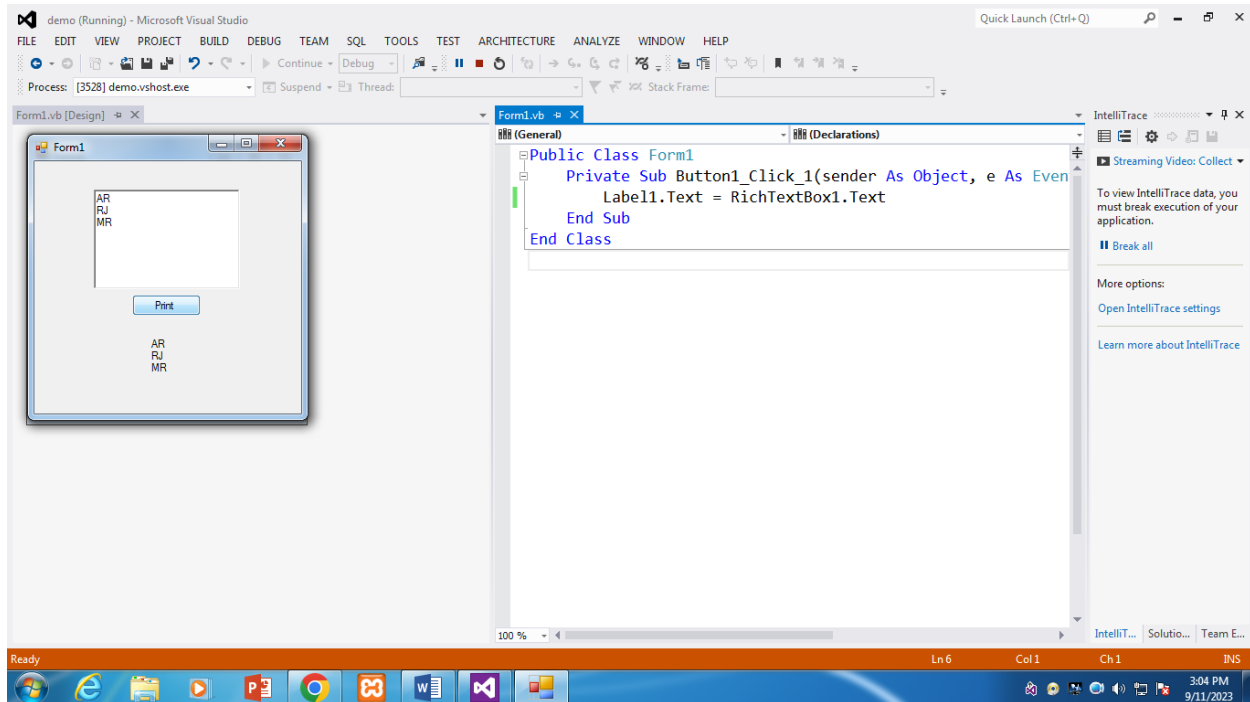
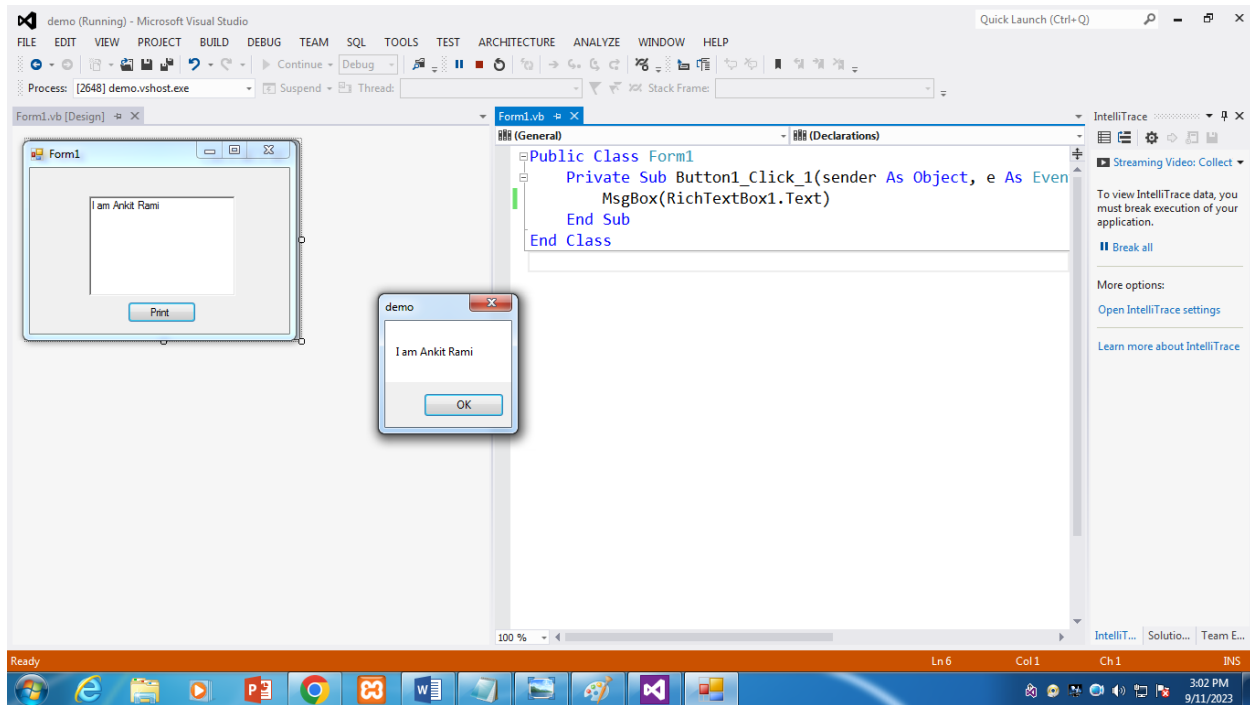
GetLineFromChar	<p>Returns the number of the line containing a specified character position in a RichTextBox control.</p> <p>Usage:</p> <pre>linenbr = RichTextBox1.GetLineFromChar(charpos)</pre> <p>Where <i>charpos</i> is a long integer that specifies the index of the character whose line you want to identify. The index of the first character in the RichTextBox control is 0.</p> <p>You use the GetLineFromChar method to find out which line in the text of a RichTextBox control contains a certain character position in the text. You might need to do this because the number of characters in each line of text can vary, making it very difficult to find out which line in the text contains a particular character, identified by its position in the text.</p>																																												
Span	<p>Selects text in a RichTextBox control based on a set of specified characters.</p> <p>Syntax</p> <p><i>object.Span characterset, forward, negate</i></p> <p>The Span method syntax has these parts:</p> <table border="1"> <thead> <tr> <th>Part</th><th>Description</th></tr> </thead> <tbody> <tr> <td><i>object</i></td><td>The RTB control.</td></tr> <tr> <td><i>characterset</i></td><td>Required. A string expression that specifies the set of characters to look for when extending the selection, based on the value of <i>negate</i>.</td></tr> <tr> <td><i>forward</i></td><td>Optional. A Boolean expression that determines which direction the insertion point moves, as described in Settings.</td></tr> <tr> <td><i>negate</i></td><td>Optional. A Boolean expression that determines whether the characters in <i>characterset</i> define the set of target characters or are excluded from the set of target characters, as described in Settings.</td></tr> </tbody> </table> <p>Settings</p> <p>The settings for <i>forward</i> are:</p> <table border="1"> <thead> <tr> <th>Setting</th><th>Description</th></tr> </thead> <tbody> <tr> <td>True</td><td>(Default) Selects text from the current insertion point or the beginning of the current selection forward, toward the end of the text.</td></tr> <tr> <td>False</td><td>Selects text from the current insertion point or the beginning of the current selection backward, toward the start of the text.</td></tr> </tbody> </table> <p>The settings for <i>negate</i> are:</p> <table border="1"> <thead> <tr> <th>Setting</th><th>Description</th></tr> </thead> <tbody> <tr> <td>True</td><td>The characters included in the selection are those which do not appear in the <i>characterset</i> argument. The selection stops at the first character found that appears in the <i>characterset</i> argument.</td></tr> <tr> <td>False</td><td>(Default) The characters included in the selection are those which appear in the <i>characterset</i> argument. The selection stops at the first character found that does not appear in the <i>characterset</i> argument.</td></tr> </tbody> </table> <p>Remarks</p> <p>The Span method is primarily used to easily select a word or sentence in the RichTextBox control.</p> <p>If the Span method cannot find the specified characters based on the values of the arguments, then the current insertion point or selection remains unchanged.</p> <p>The Span method does not return any data.</p> <p>Example:</p> <p>This example defines a pair of keyboard shortcuts that selects text in a RichTextBox control to the end of a sentence (CTRL+S) or the end of a word (CTRL+W). To try this example, put a RichTextBox control on a form. Paste this code into the KeyUp event of the RichTextBox control. Then run the example.</p> <pre>Private Sub RichTextBox1_KeyUp (KeyCode As Integer, Shift As Integer) If Shift = vbCtrlMask Then Select Case KeyCode ' If Ctrl+S: Case vbKeyS ' Select to the end of the sentence. RichTextBox1.Span ".?!:", True, True ' Extend selection to include punctuation. RichTextBox1.SelectionLength = RichTextBox1.SelectionLength + 1 ' If Ctrl+W: Case vbKeyW ' Select to the end of the word. RichTextBox1.Span ".?!:", True, True End Select End If End Sub</pre> <p>Moves the insertion point up to, but not including, the first character that is a member of the specified character set in a RichTextBox control.</p> <p>Syntax</p> <p><i>object.Upto(characterset, forward, negate)</i></p> <p>The Upto method syntax has these parts:</p> <table border="1"> <thead> <tr> <th>Part</th><th>Description</th></tr> </thead> <tbody> <tr> <td><i>object</i></td><td>Required. The RTB control.</td></tr> <tr> <td><i>characterset</i></td><td>Required. A string expression that specifies the set of characters to look for when moving the insertion point, based on the value of <i>negate</i>.</td></tr> <tr> <td><i>forward</i></td><td>Optional. A Boolean expression that determines which direction the insertion point moves, as described in Settings.</td></tr> <tr> <td><i>negate</i></td><td>Optional. A Boolean expression that determines whether the characters in <i>characterset</i> define the set of target characters or are excluded from the set of target characters, as described in Settings.</td></tr> </tbody> </table> <p>Settings</p> <p>The settings for <i>forward</i> are:</p> <table border="1"> <thead> <tr> <th>Setting</th><th>Description</th></tr> </thead> <tbody> <tr> <td>True</td><td>(Default) Moves the insertion point forward, toward the end of the text.</td></tr> <tr> <td>False</td><td>Moves the insertion point backward, toward the start of the text.</td></tr> </tbody> </table> <p>The settings for <i>negate</i> are:</p> <table border="1"> <thead> <tr> <th>Setting</th><th>Description</th></tr> </thead> <tbody> <tr> <td>True</td><td>The characters not specified in the <i>characterset</i> argument are used to move the insertion point.</td></tr> <tr> <td>False</td><td>(Default) The characters specified in the <i>characterset</i> argument are used to move the insertion point.</td></tr> </tbody> </table> <p>Upto Method Example</p> <p>This example defines a pair of keyboard shortcuts that moves the insertion point in a RichTextBox control to the end of a sentence (ALT+S) or the end of a word (ALT+W). To try this example, put a RichTextBox control on a form. Paste this code into the KeyUp event of the RichTextBox control. Then run the example.</p> <pre>Private Sub RichTextBox1_KeyUp (KeyCode As Integer, Shift As Integer) If Shift = vbAltMask Then Select Case KeyCode ' If Alt+S: Case vbKeyS ' Move insertion point to the end of the sentence. RichTextBox1.Upto ".?!:", True, False ' If Alt+W: Case vbKeyW ' Move insertion point to the end of the word. RichTextBox1.Upto ".?!:", True, False End Select End If End Sub</pre>	Part	Description	<i>object</i>	The RTB control.	<i>characterset</i>	Required. A string expression that specifies the set of characters to look for when extending the selection, based on the value of <i>negate</i> .	<i>forward</i>	Optional. A Boolean expression that determines which direction the insertion point moves, as described in Settings.	<i>negate</i>	Optional. A Boolean expression that determines whether the characters in <i>characterset</i> define the set of target characters or are excluded from the set of target characters, as described in Settings.	Setting	Description	True	(Default) Selects text from the current insertion point or the beginning of the current selection forward, toward the end of the text.	False	Selects text from the current insertion point or the beginning of the current selection backward, toward the start of the text.	Setting	Description	True	The characters included in the selection are those which do not appear in the <i>characterset</i> argument. The selection stops at the first character found that appears in the <i>characterset</i> argument.	False	(Default) The characters included in the selection are those which appear in the <i>characterset</i> argument. The selection stops at the first character found that does not appear in the <i>characterset</i> argument.	Part	Description	<i>object</i>	Required. The RTB control.	<i>characterset</i>	Required. A string expression that specifies the set of characters to look for when moving the insertion point, based on the value of <i>negate</i> .	<i>forward</i>	Optional. A Boolean expression that determines which direction the insertion point moves, as described in Settings.	<i>negate</i>	Optional. A Boolean expression that determines whether the characters in <i>characterset</i> define the set of target characters or are excluded from the set of target characters, as described in Settings.	Setting	Description	True	(Default) Moves the insertion point forward, toward the end of the text.	False	Moves the insertion point backward, toward the start of the text.	Setting	Description	True	The characters not specified in the <i>characterset</i> argument are used to move the insertion point.	False	(Default) The characters specified in the <i>characterset</i> argument are used to move the insertion point.
Part	Description																																												
<i>object</i>	The RTB control.																																												
<i>characterset</i>	Required. A string expression that specifies the set of characters to look for when extending the selection, based on the value of <i>negate</i> .																																												
<i>forward</i>	Optional. A Boolean expression that determines which direction the insertion point moves, as described in Settings.																																												
<i>negate</i>	Optional. A Boolean expression that determines whether the characters in <i>characterset</i> define the set of target characters or are excluded from the set of target characters, as described in Settings.																																												
Setting	Description																																												
True	(Default) Selects text from the current insertion point or the beginning of the current selection forward, toward the end of the text.																																												
False	Selects text from the current insertion point or the beginning of the current selection backward, toward the start of the text.																																												
Setting	Description																																												
True	The characters included in the selection are those which do not appear in the <i>characterset</i> argument. The selection stops at the first character found that appears in the <i>characterset</i> argument.																																												
False	(Default) The characters included in the selection are those which appear in the <i>characterset</i> argument. The selection stops at the first character found that does not appear in the <i>characterset</i> argument.																																												
Part	Description																																												
<i>object</i>	Required. The RTB control.																																												
<i>characterset</i>	Required. A string expression that specifies the set of characters to look for when moving the insertion point, based on the value of <i>negate</i> .																																												
<i>forward</i>	Optional. A Boolean expression that determines which direction the insertion point moves, as described in Settings.																																												
<i>negate</i>	Optional. A Boolean expression that determines whether the characters in <i>characterset</i> define the set of target characters or are excluded from the set of target characters, as described in Settings.																																												
Setting	Description																																												
True	(Default) Moves the insertion point forward, toward the end of the text.																																												
False	Moves the insertion point backward, toward the start of the text.																																												
Setting	Description																																												
True	The characters not specified in the <i>characterset</i> argument are used to move the insertion point.																																												
False	(Default) The characters specified in the <i>characterset</i> argument are used to move the insertion point.																																												
SelPrint	<p>Sends formatted text in a RichTextBox control to a device for printing.</p> <p>Syntax</p> <p><i>object.SelPrint(hdc)</i></p> <p>The SelPrint method syntax has these parts:</p> <table border="1"> <thead> <tr> <th>Part</th><th>Description</th></tr> </thead> <tbody> <tr> <td><i>object</i></td><td>The RTB control.</td></tr> <tr> <td><i>hdc</i></td><td>The device context of the device you plan to use to print the contents of the control.</td></tr> </tbody> </table> <p>Remarks</p> <p>If text is selected in the RichTextBox control, the SelPrint method sends only the selected text to the target device. If no text is selected, the entire contents of the RichTextBox are sent to the target device.</p> <p>The SelPrint method does not print text from the RichTextBox control. Rather, it sends a copy of formatted text to a device which can print the text. For example, you can send the text to the Printer object using code as follows:</p> <pre>RichTextBox1.SelPrint(Printer.hDC)</pre> <p>Notice that the hDC property of the Printer object is used to specify the device context argument of the SelPrint method.</p> <p>Note If you use the Printer object as the destination of the text from the RichTextBox control, you must first initialize the device context of the Printer object by printing something like a zero-length string.</p>	Part	Description	<i>object</i>	The RTB control.	<i>hdc</i>	The device context of the device you plan to use to print the contents of the control.																																						
Part	Description																																												
<i>object</i>	The RTB control.																																												
<i>hdc</i>	The device context of the device you plan to use to print the contents of the control.																																												

Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



Ex- RichTextbox Control





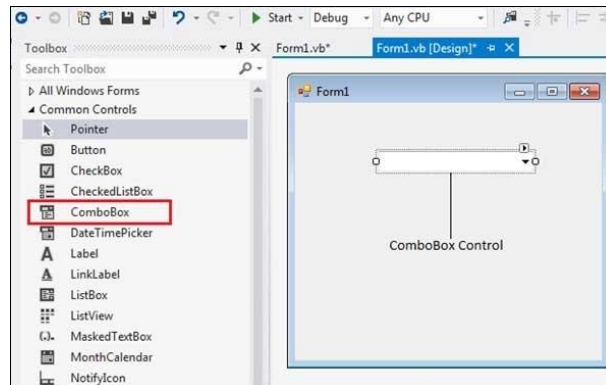
Data Selection Tools

- These are tools that can be used to select data from a specific list called data selection tools.
- **Combo Box**
 - It is used to display a drop-down list of various items.
- **List Box**
 - ListBox control is used to display a list of items in Windows form. It allows the user to select one or more items from the ListBox Control
- **Radio Button**
 - RadioButton is used to select one option from the number of choices.
 - RadioButton control is used to provide a set of mutually exclusive options.
- **Check Box**
 - CheckBox control is a control that allows the user to select or deselect options from the available options.
 - CheckBox control allows the user to set true/false or yes/no type options.



⇒ ComboBox Control in VB.net

- ComboBox control is used to display a drop-down list of various items.
- It is a combination of a text box in which the user enters an item and a drop-down list from which the user selects an item.



ComboBox Control Properties in VB.net

Sr.No.	Property & Description
1	AllowSelection Gets a value indicating whether the list enables selection of list items.
2	AutoCompleteCustomSource Gets or sets a custom System.Collections.Specialized.StringCollection to use when the AutoCompleteSourceproperty is set to CustomSource.
3	AutoCompleteMode Gets or sets an option that controls how automatic completion works for the ComboBox.
4	AutoCompleteSource Gets or sets a value specifying the source of complete strings used for automatic completion.
5	DataBindings Gets the data bindings for the control.
6	DataManager Gets the CurrencyManager associated with this control.
7	DataSource Gets or sets the data source for this ComboBox.
8	DropDownHeight Gets or sets the height in pixels of the drop-down portion of the ComboBox.
9	DropDownStyle Gets or sets a value specifying the style of the combo box.
10	DropDownWidth Gets or sets the width of the of the drop-down portion of a combo box.
11	DroppedDown Gets or sets a value indicating whether the combo box is displaying its drop-down portion.
12	FlatStyle Gets or sets the appearance of the ComboBox.

13	ItemHeight Gets or sets the height of an item in the combo box.
14	Items Gets an object representing the collection of the items contained in this ComboBox.
15	MaxDropDownItems Gets or sets the maximum number of items to be displayed in the drop-down part of the combo box.
16	MaxLength Gets or sets the maximum number of characters a user can enter in the editable area of the combo box.
17	SelectedIndex Gets or sets the index specifying the currently selected item.
18	SelectedItem Gets or sets currently selected item in the ComboBox.
19	SelectedText Gets or sets the text that is selected in the editable portion of a ComboBox.
20	SelectedValue Gets or sets the value of the member property specified by the ValueMember property.
21	SelectionLength Gets or sets the number of characters selected in the editable portion of the combo box.
22	SelectionStart Gets or sets the starting index of text selected in the combo box.
23	Sorted Gets or sets a value indicating whether the items in the combo box are sorted.
24	Text Gets or sets the text associated with this control.

Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



Combobox Control Methods in VB.net

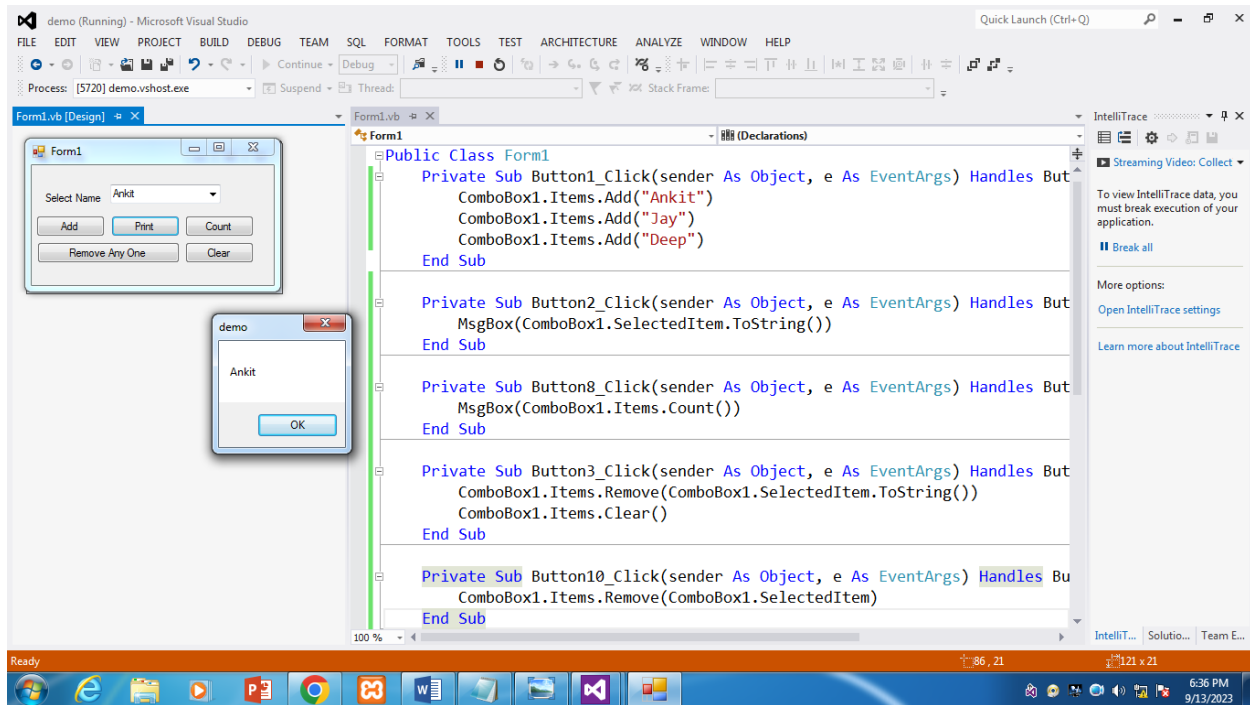
Sr.No.	Method Name & Description
1	BeginUpdate Prevents the control from drawing until the EndUpdate method is called, while items are added to the combo box one at a time.
2	EndUpdate Resumes drawing of a combo box, after it was turned off by the BeginUpdate method.
3	FindString Finds the first item in the combo box that starts with the string specified as an argument.
4	FindStringExact Finds the first item in the combo box that exactly matches the specified string.
5	SelectAll Selects all the text in the editable area of the combo box.

Combobox Control Events in VB.net

Sr.No.	Event & Description
1	DropDown Occurs when the drop-down portion of a combo box is displayed.
2	DropDownClosed Occurs when the drop-down portion of a combo box is no longer visible.
3	DropDownStyleChanged Occurs when the DropDownStyle property of the ComboBox has changed.
4	SelectedIndexChanged Occurs when the SelectedIndex property of a ComboBox control has changed.
5	SelectionChangeCommitted Occurs when the selected item has changed and the change appears in the combo box.

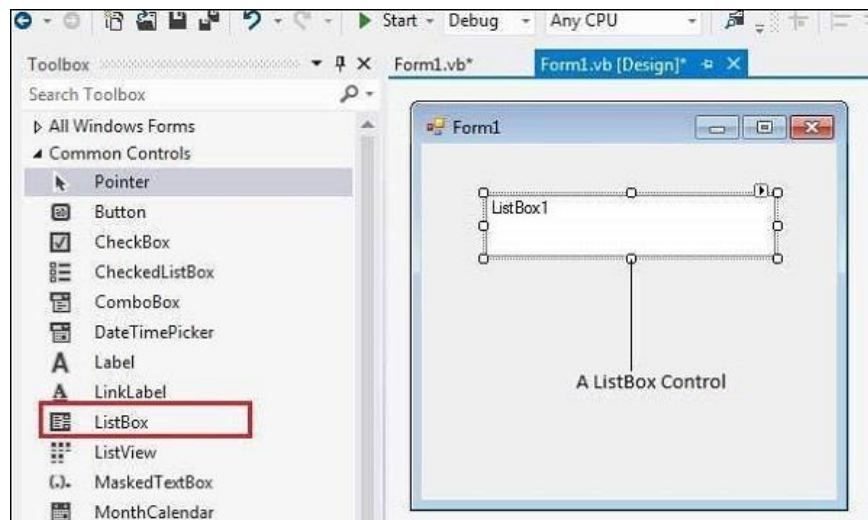


Ex- ComboBox Control



⇒ List Box Control in VB.net

- ListBox represents a Windows control to display a list of items to a user.
- A user can select an item from the list. It allows the programmer to add items at design time by using the properties window or at the runtime.





List Box Control Properties in VB.net

Sr.No.	Property & Description
1	AllowSelection Gets a value indicating whether the ListBox currently enables selection of list items.
2	BorderStyle Gets or sets the type of border drawn around the list box.
3	ColumnWidth Gets or sets the width of columns in a multicolumn list box.
4	HorizontalExtent Gets or sets the horizontal scrolling area of a list box.
5	HorizontalScrollBar Gets or sets the value indicating whether a horizontal scrollbar is displayed in the list box.
6	ItemHeight Gets or sets the height of an item in the list box.
7	Items Gets the items of the list box.
8	MultiColumn Gets or sets a value indicating whether the list box supports multiple columns.
9	ScrollAlwaysVisible Gets or sets a value indicating whether the vertical scroll bar is shown at all times.
10	SelectedIndex Gets or sets the zero-based index of the currently selected item in a list box.
11	SelectedIndices Gets a collection that contains the zero-based indexes of all currently selected items in the list box.
12	SelectedItem Gets or sets the currently selected item in the list box.
13	SelectedItems Gets a collection containing the currently selected items in the list box.
14	SelectedValue Gets or sets the value of the member property specified by the ValueMember property.
15	SelectionMode Gets or sets the method in which items are selected in the list box. This property has values - <ul style="list-style-type: none">• None• One• MultiSimple• MultiExtended
16	Sorted Gets or sets a value indicating whether the items in the list box are sorted alphabetically.
17	Text Gets or searches for the text of the currently selected item in the list box.
18	TopIndex Gets or sets the index of the first visible item of a list box.

List Box Control Method in VB.net

Sr.No.	Method Name & Description
1	BeginUpdate Prevents the control from drawing until the EndUpdate method is called, while items are added to the ListBox one at a time.
2	ClearSelected Unselects all items in the ListBox.
3	EndUpdate Resumes drawing of a list box after it was turned off by the BeginUpdate method.
4	FindString Finds the first item in the ListBox that starts with the string specified as an argument.
5	FindStringExact Finds the first item in the ListBox that exactly matches the specified string.
6	GetSelected Returns a value indicating whether the specified item is selected.
7	SetSelected Selects or clears the selection for the specified item in a ListBox.
8	OnSelectedIndexChanged Raises the SelectedIndexChanged event.
8	OnSelectedValueChanged Raises the SelectedValueChanged event.

Design and Develop by Assistant Professor Ankit Rami

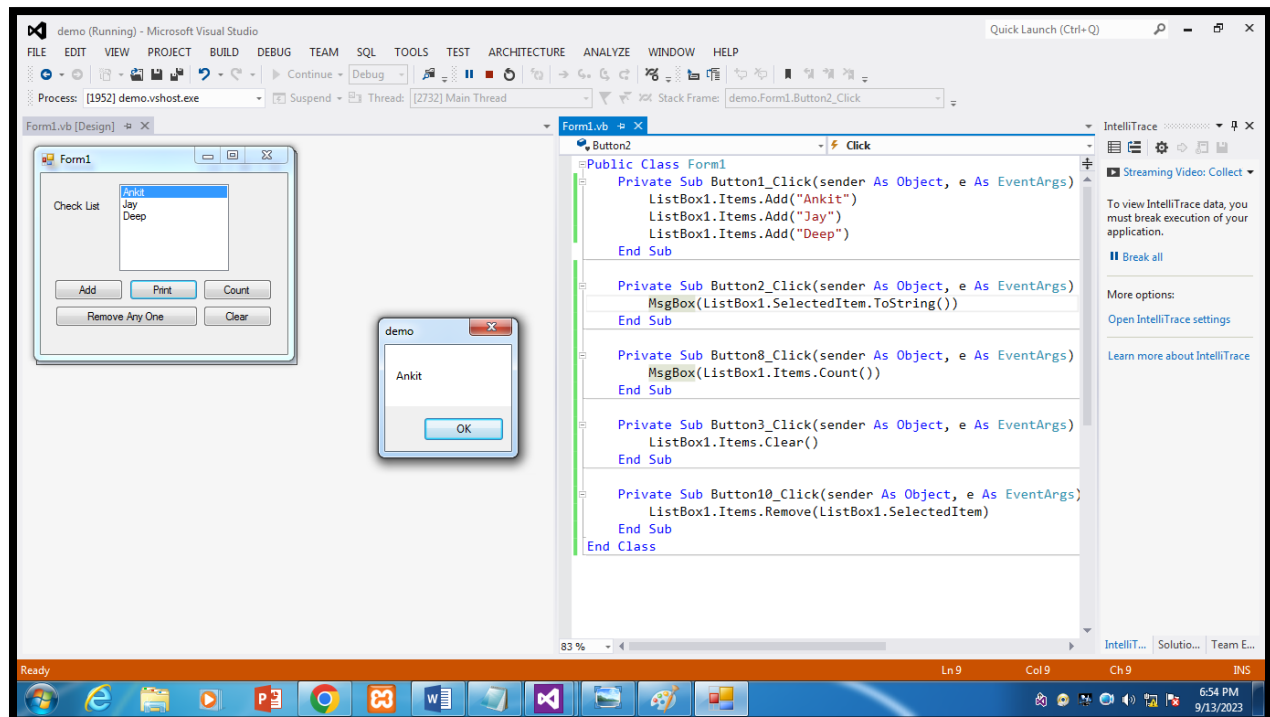
Any Query Contact - 8460467193



List Box Control Events in VB.Net

Sr.No.	Event & Description
1	Click Occurs when a list box is selected.
2	SelectedIndexChanged Occurs when the SelectedIndex property of a list box is changed.

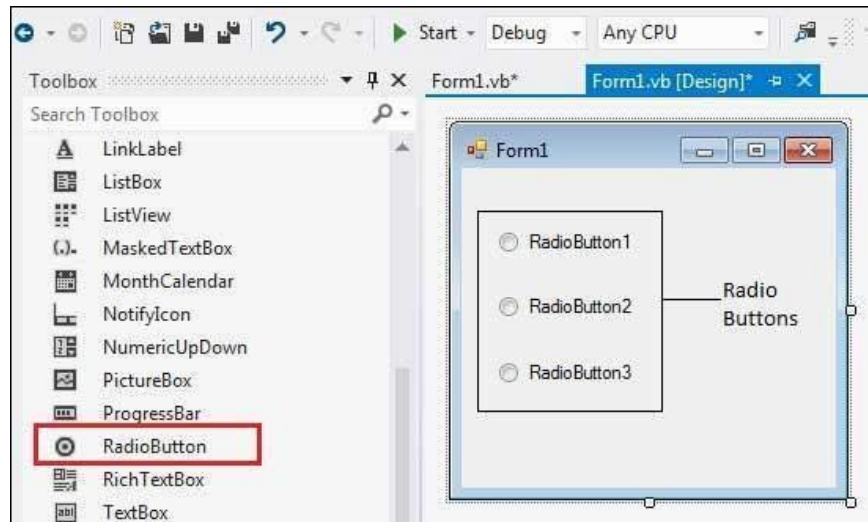
Ex- List Box Control





⇒ Radio Button Control in VB.net

- Radio Button control is used to provide a set of mutually exclusive options. The user can select one radio button in a group.
- If you need to place more than one group of radio buttons in the same form, you should place them in different container controls like a Group Box control.



Radio Button Control Properties in VB.net

Sr.No.	Property & Description
1	Appearance Gets or sets a value determining the appearance of the radio button.
2	AutoCheck Gets or sets a value indicating whether the Checked value and the appearance of the control automatically change when the control is clicked.
3	CheckAlign Gets or sets the location of the check box portion of the radio button.
4	Checked Gets or sets a value indicating whether the control is checked.
5	Text Gets or sets the caption for a radio button.
6	TabStop Gets or sets a value indicating whether a user can give focus to the RadioButton control using the TAB key.

Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



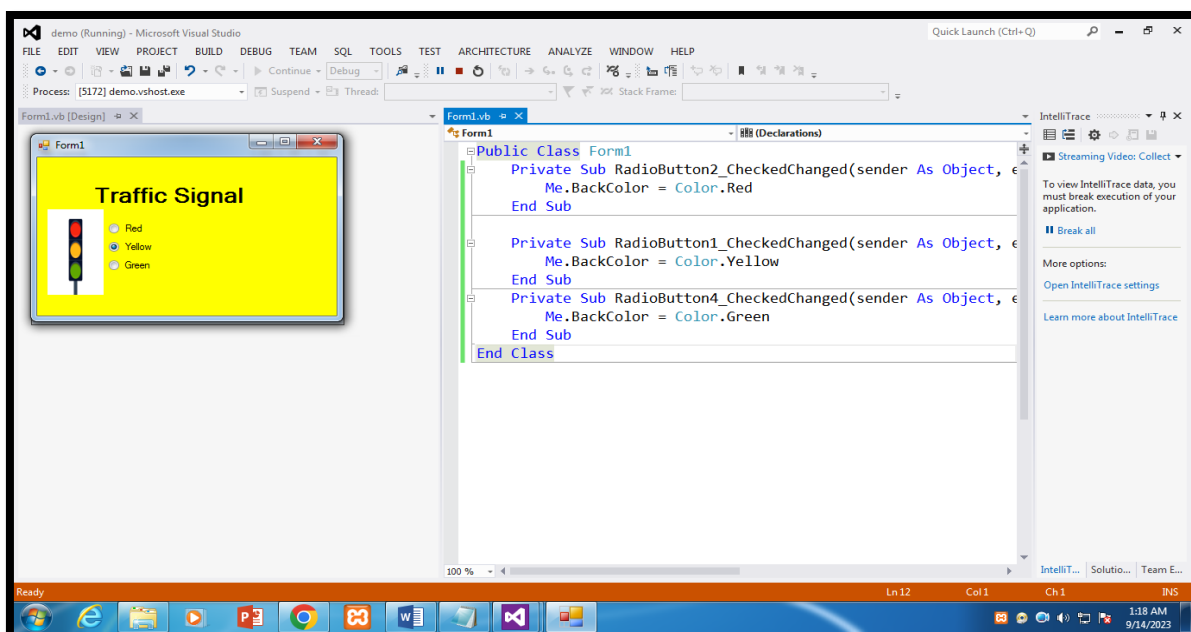
Radio Button Control Method in VB.net

Sr.No.	Method Name & Description
1	PerformClick Generates a Click event for the control, simulating a click by a user.

Radio Button Control Events in VB.net

Sr.No	Event & Description
1	AppearanceChanged Occurs when the value of the Appearance property of the RadioButton control is changed.
2	CheckedChanged Occurs when the value of the Checked property of the RadioButton control is changed.

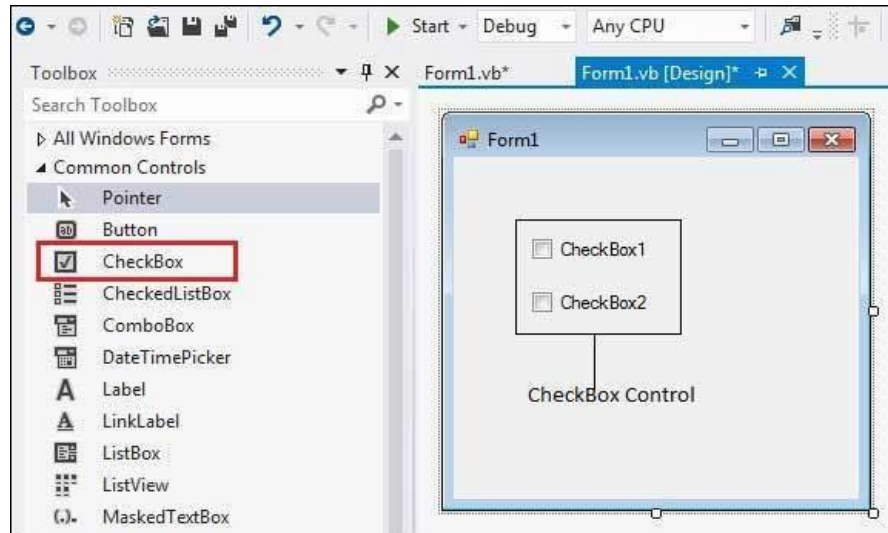
Ex- Radio Button Control





Check Box Control in VB.net

- Checkbox control allows the user to set true/false or yes/no type options.
- The user can select or deselect it. When a check box is selected it has the value True, and when it is cleared, it holds the value False.



Check Box Control Properties in VB.net

Sr.No.	Property & Description
1	Appearance Gets or sets a value determining the appearance of the check box.
2	AutoCheck Gets or sets a value indicating whether the Checked or CheckState value and the appearance of the control automatically change when the check box is selected.
3	CheckAlign Gets or sets the horizontal and vertical alignment of the check mark on the check box.
4	Checked Gets or sets a value indicating whether the check box is selected.
5	CheckState Gets or sets the state of a check box.
6	Text Gets or sets the caption of a check box.
7	ThreeState Gets or sets a value indicating whether or not a check box should allow three check states rather than two.

Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



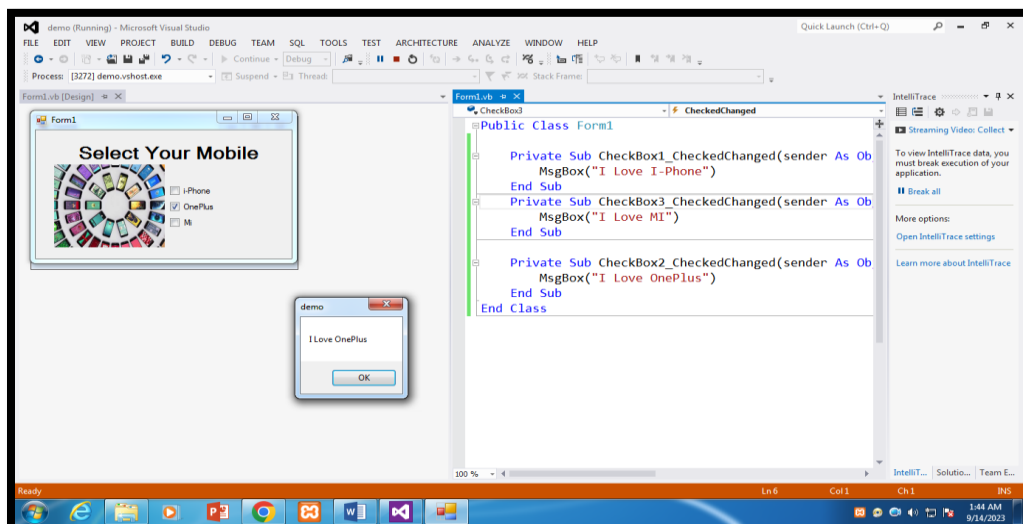
Check Box Control Methods in VB.net

Sr.No.	Method Name & Description
1	OnCheckedChanged Raises the CheckedChanged event.
2	OnCheckStateChanged Raises the CheckStateChanged event.
3	OnClick Raises the OnClick event.

Check Box Control Events in VB.net

Sr.No.	Event & Description
1	AppearanceChanged Occurs when the value of the Appearance property of the check box is changed.
2	CheckedChanged Occurs when the value of the Checked property of the CheckBox control is changed.
3	CheckStateChanged Occurs when the value of the CheckState property of the CheckBox control is changed.

Ex- Check Box Control



Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



Data Display Tools

- These are tools used to view pictures, give a notice or instruction, and scroll or Track data.

➤ **Label**

- It is used to display some informative text on GUI.

➤ **Picture Box**

- It is used to display images on the form

➤ **Scroll Bar**

- The ScrollBar controls display vertical and horizontal scroll bars on the form.
- This is used for navigating through large amount of information.

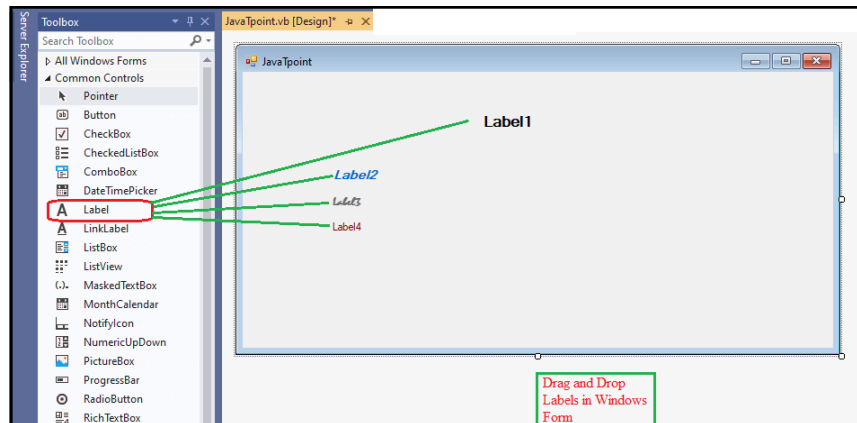
➤ **Track Bar**

- The TrackBar control is similar to the ScrollBar control, but it doesn't cover a continuous range of values.
- The TrackBar control has a fixed number of tick marks, which the developer can label. Users can place the slider's indicator to be desired value.



Label Control in VB.net

- It is generally used to display some informative text on the GUI which is not changed during runtime.
- In VB.NET, a label control is used to display descriptive text for the form in control.
- It does not participate in user input or keyboard or mouse events.
- The labels are defined in the class System.Windows.Forms namespace.



Label Control Properties in VB.net

Sr.No.	Property & Description
1	AutoSize Gets or sets a value specifying if the control should be automatically resized to display all its contents.
2	BorderStyle Gets or sets the border style for the control.
3	FlatStyle Gets or sets the flat style appearance of the Label control
4	Font Gets or sets the font of the text displayed by the control.
5	FontHeight Gets or sets the height of the font of the control.
6	ForeColor Gets or sets the foreground color of the control.

7	PreferredHeight Gets the preferred height of the control.
8	PreferredWidth Gets the preferred width of the control.
9	TabStop Gets or sets a value indicating whether the user can tab to the Label. This property is not used by this class.
10	Text Gets or sets the text associated with this control.
11	TextAlign Gets or sets the alignment of text in the label.



Label Control Methods in VB.net

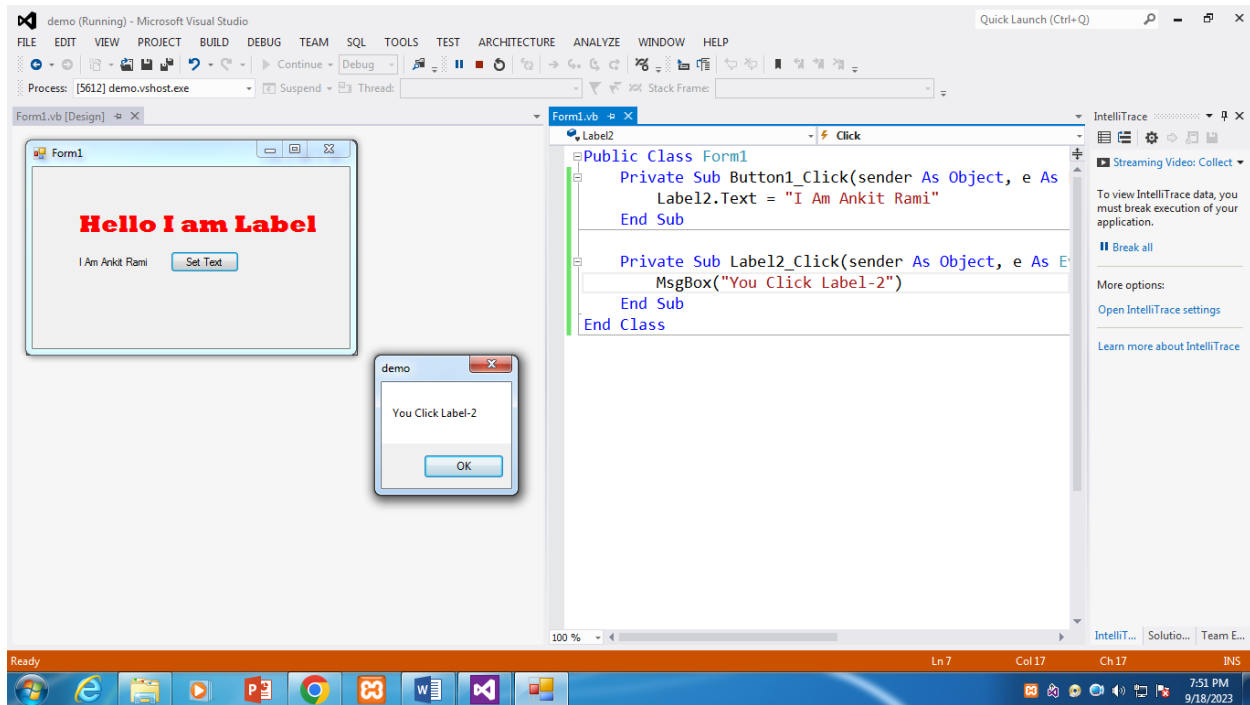
Sr.No.	Method Name & Description
1	GetPreferredSize Retrieves the size of a rectangular area into which a control can be fitted.
2	Refresh Forces the control to invalidate its client area and immediately redraw itself and any child controls.
3	Select Activates the control.
4	Show Displays the control to the user.
5	ToString Returns a String that contains the name of the control.

Label Control Events in VB.net

Sr.No.	Event & Description
1	AutoSizeChanged Occurs when the value of the AutoSize property changes.
2	Click Occurs when the control is clicked.
3	DoubleClick Occurs when the control is double-clicked.
4	GotFocus Occurs when the control receives focus.
5	Leave Occurs when the input focus leaves the control.
6	LostFocus Occurs when the control loses focus.
7	TabIndexChanged Occurs when the TabIndex property value changes.
8	TabStopChanged Occurs when the TabStop property changes.
9	TextChanged Occurs when the Text property value changes.

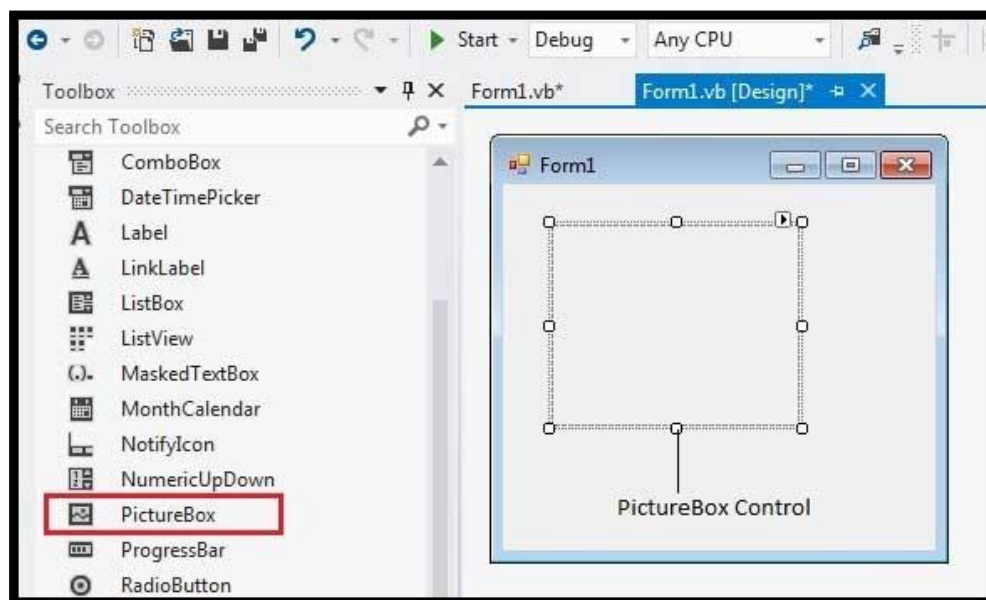


Ex- Label Control



Picture Box Control in VB.net

- PictureBox control is used to display the images on Windows Form.
- PictureBox control has an image property that allows the user to set the image at runtime or design time.



Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



Picture Box Control Properties in VB.net

Sr.No.	Property & Description
1	AllowDrop Specifies whether the picture box accepts data that a user drags on it.
2	ErrorImage Gets or specifies an image to be displayed when an error occurs during the image-loading process or if the image load is cancelled.
3	Image Gets or sets the image that is displayed in the control.
4	ImageLocation Gets or sets the path or the URL for the image displayed in the control.
5	InitialImage Gets or sets the image displayed in the control when the main image is loaded.
6	SizeMode Determines the size of the image to be displayed in the control. This property takes its value from the PictureBoxSizeMode enumeration, which has values – <ul style="list-style-type: none">• Normal – the upper left corner of the image is placed at upper left part of the picture box• StretchImage – allows stretching of the image• AutoSize – allows resizing the picture box to the size of the image• CenterImage – allows centering the image in the picture box• Zoom – allows increasing or decreasing the image size to maintain the size ratio.
7	TabIndex Gets or sets the tab index value.
8	TabStop Specifies whether the user will be able to focus on the picture box by using the TAB key.
9	Text Gets or sets the text for the picture box.
10	WaitOnLoad Specifies whether or not an image is loaded synchronously.



Picture Box Control Events in VB.net

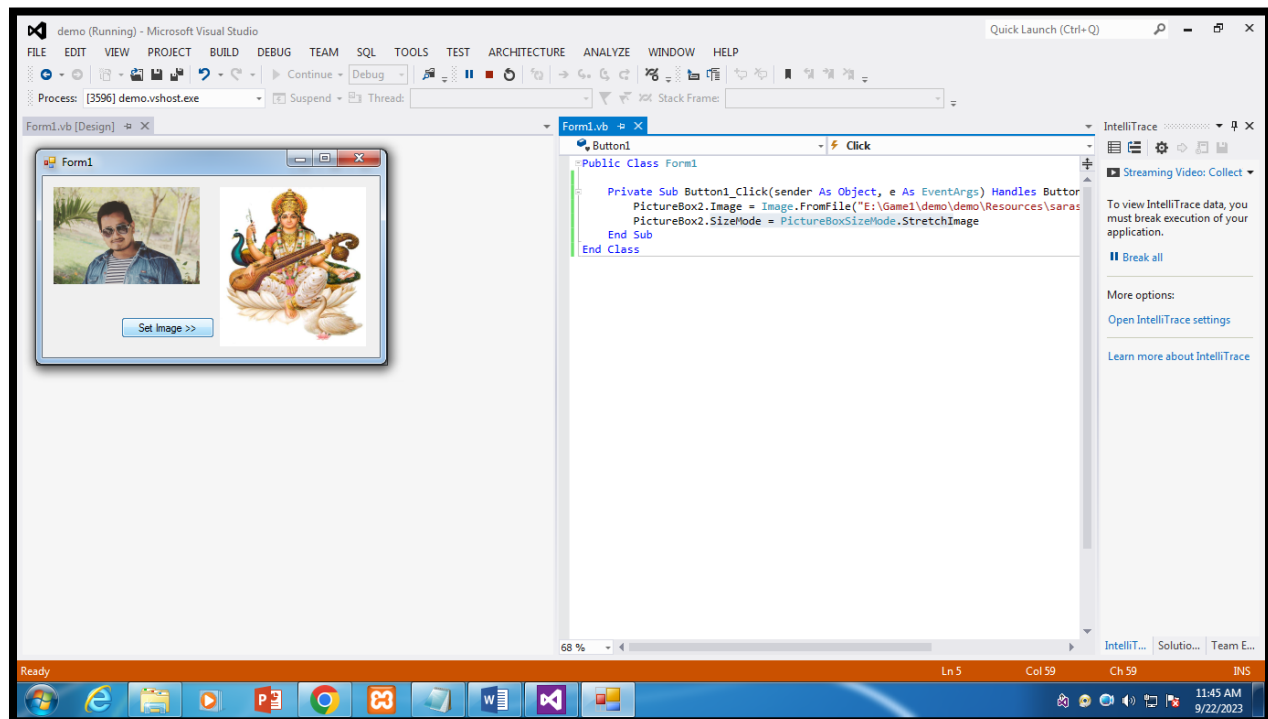
Sr.No.	Event & Description
1	CausesValidationChanged Overrides the Control.CausesValidationChanged property.
2	Click Occurs when the control is clicked.
3	Enter Overrides the Control.Enter property.
4	FontChanged Occurs when the value of the Font property changes.
5	ForeColorChanged Occurs when the value of the ForeColor property changes.
6	KeyDown Occurs when a key is pressed when the control has focus.
7	KeyPress Occurs when a key is pressed when the control has focus.
8	KeyUp Occurs when a key is released when the control has focus.
9	Leave Occurs when input focus leaves the PictureBox.
10	LoadCompleted Occurs when the asynchronous image-load operation is completed, been canceled, or raised an exception.
11	LoadProgressChanged Occurs when the progress of an asynchronous image-loading operation has changed.
12	Resize Occurs when the control is resized.
13	RightToLeftChanged Occurs when the value of the RightToLeft property changes.
14	SizeChanged Occurs when the Size property value changes.
15	SizeModeChanged Occurs when SizeMode changes.
16	TabIndexChanged Occurs when the value of the TabIndex property changes.
17	TabStopChanged Occurs when the value of the TabStop property changes.
18	TextChanged Occurs when the value of the Text property changes.



Picture Box Control Methods in VB.net

Sr.No.	Method Name & Description
1	CancelAsync Cancels an asynchronous image load.
2	Load Displays an image in the picture box
3	LoadAsync Loads image asynchronously.
4	ToString Returns the string that represents the current picture box.

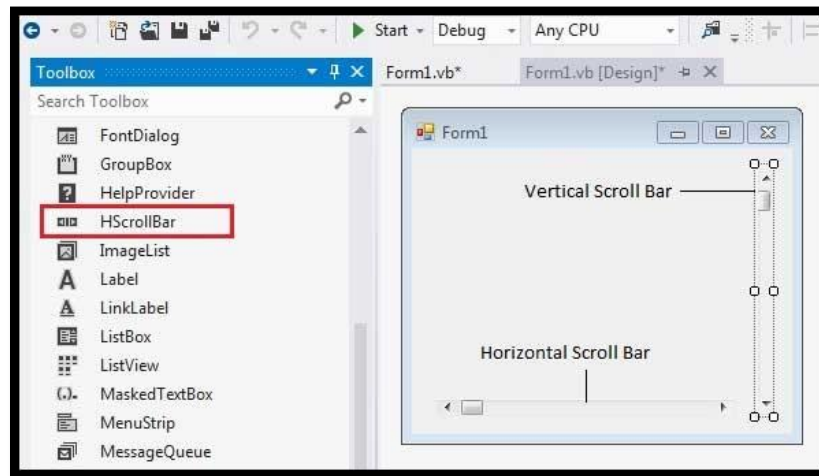
Ex- Picture Box Control





Scroll Bar Control in VB.net

- ScrollBar control is used to create and display vertical and horizontal scroll bars on the Windows form
- It is used when we have large information in a form, and we are unable to see all the data. Therefore, we used VB.NET ScrollBar control.
- ScrollBar is of two types: HScrollBar for displaying scroll bars and VScrollBar for displaying Vertical Scroll bars.



Scroll Bar Control Properties in VB.net

Sr.No.	Property & Description
1	AutoSize Gets or sets a value indicating whether the ScrollBar is automatically resized to fit its contents.
2	BackColor Gets or sets the background color for the control.
3	ForeColor Gets or sets the foreground color of the scroll bar control.
4	ImeMode Gets or sets the Input Method Editor (IME) mode supported by this control.
5	LargeChange Gets or sets a value to be added to or subtracted from the Value property when the scroll box is moved a large distance.

6	Maximum Gets or sets the upper limit of values of the scrollable range.
7	Minimum Gets or sets the lower limit of values of the scrollable range.
8	SmallChange Gets or sets the value to be added to or subtracted from the Value property when the scroll box is moved a small distance.
9	Value Gets or sets a numeric value that represents the current position of the scroll box on the scroll bar control.



Scroll Bar Control Methods in VB.net

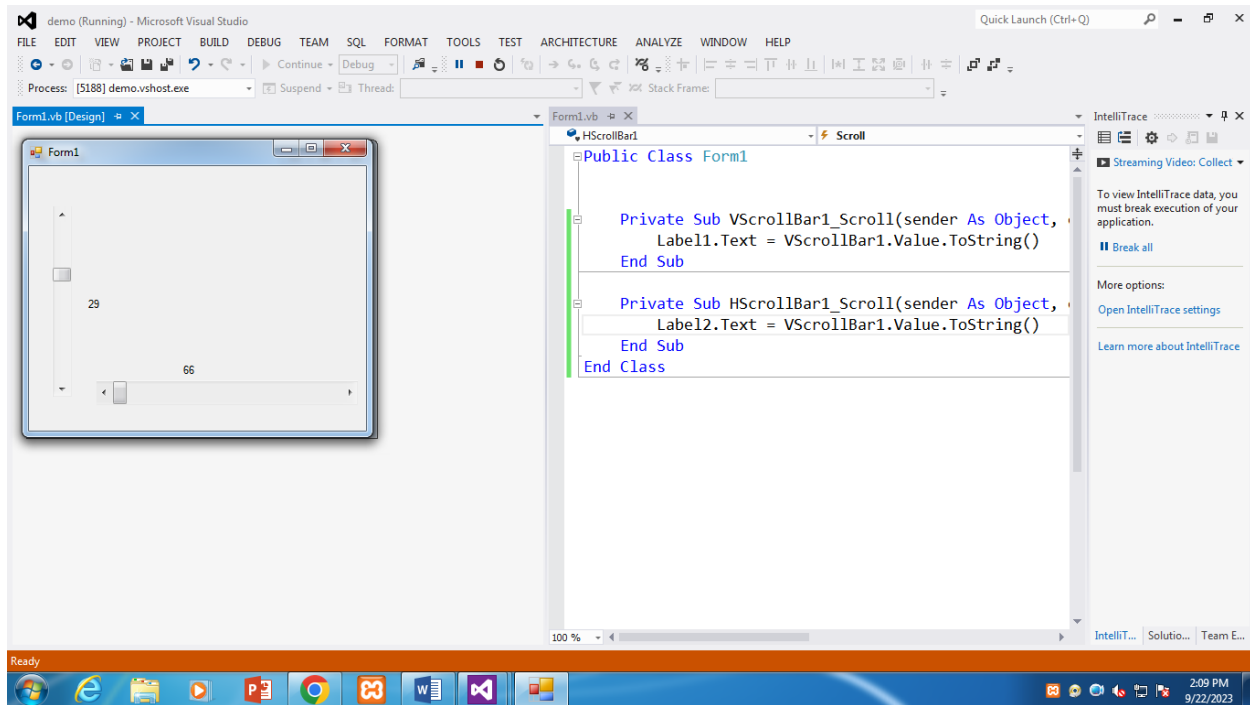
Sr.No.	Method Name & Description
1	OnClick Generates the Click event.
2	Select Activates the control.

Scroll Bar Control Events in VB.net

Sr.No.	Event & Description
1	Click Occurs when the control is clicked.
2	DoubleClick Occurs when the user double-clicks the control.
3	Scroll Occurs when the control is moved.
4	ValueChanged Occurs when the Value property changes, either by handling the Scroll event or programmatically.

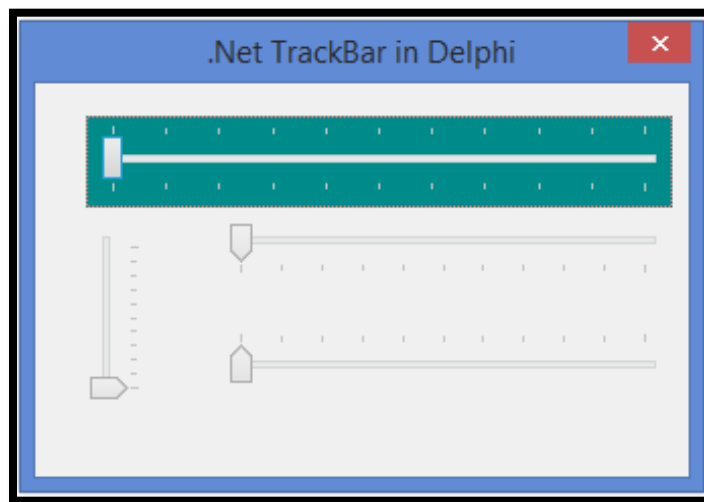


Ex- Scroll Bar Control



Track Bar Control in VB.net

- TrackBar control is similar to the ScrollBar control, but it doesn't cover a continuous range of values.
- TrackBar control has a fixed number of tick marks, which the developer can label. Users can place the slider's indicator to be desired value.





Track Bar Control Properties in VB.net

Sr.No.	Property & Description
1	AutoSize Gets or sets a value indicating whether the ScrollBar is automatically resized to fit its contents.
2	BackColor Gets or sets the background color for the control.
3	ForeColor Gets or sets the foreground color of the scroll bar control.
4	ImeMode Gets or sets the Input Method Editor (IME) mode supported by this control.
5	LargeChange Gets or sets a value to be added to or subtracted from the Value property when the scroll box is moved a large distance.

6	Maximum Gets or sets the upper limit of values of the scrollable range.
7	Minimum Gets or sets the lower limit of values of the scrollable range.
8	SmallChange Gets or sets the value to be added to or subtracted from the Value property when the scroll box is moved a small distance.
9	Value Gets or sets a numeric value that represents the current position of the scroll box on the scroll bar control.

Track Bar Control Methods in VB.net

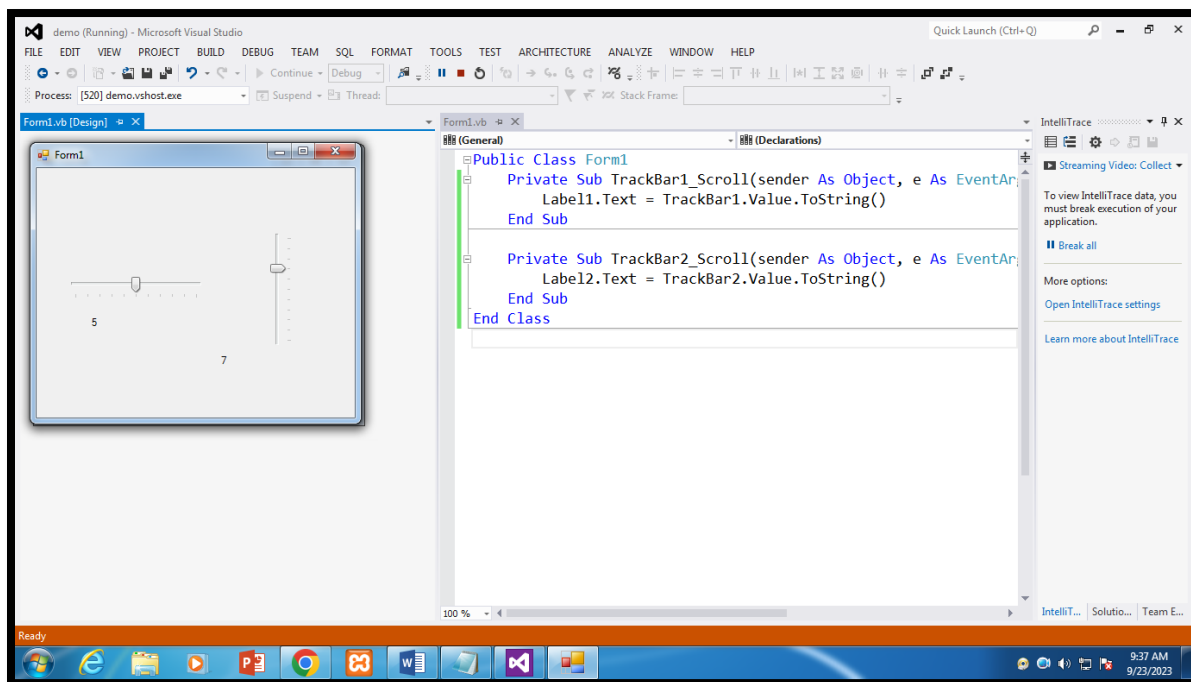
Sr.No.	Method Name & Description
1	OnClick Generates the Click event.
2	Select Activates the control.



Track Bar Control Events in VB.net

Sr.No.	Event & Description
1	Click Occurs when the control is clicked.
2	DoubleClick Occurs when the user double-clicks the control.
3	Scroll Occurs when the control is moved.
4	ValueChanged Occurs when the Value property changes, either by handling the Scroll event or programmatically.

Ex- Track Bar Control



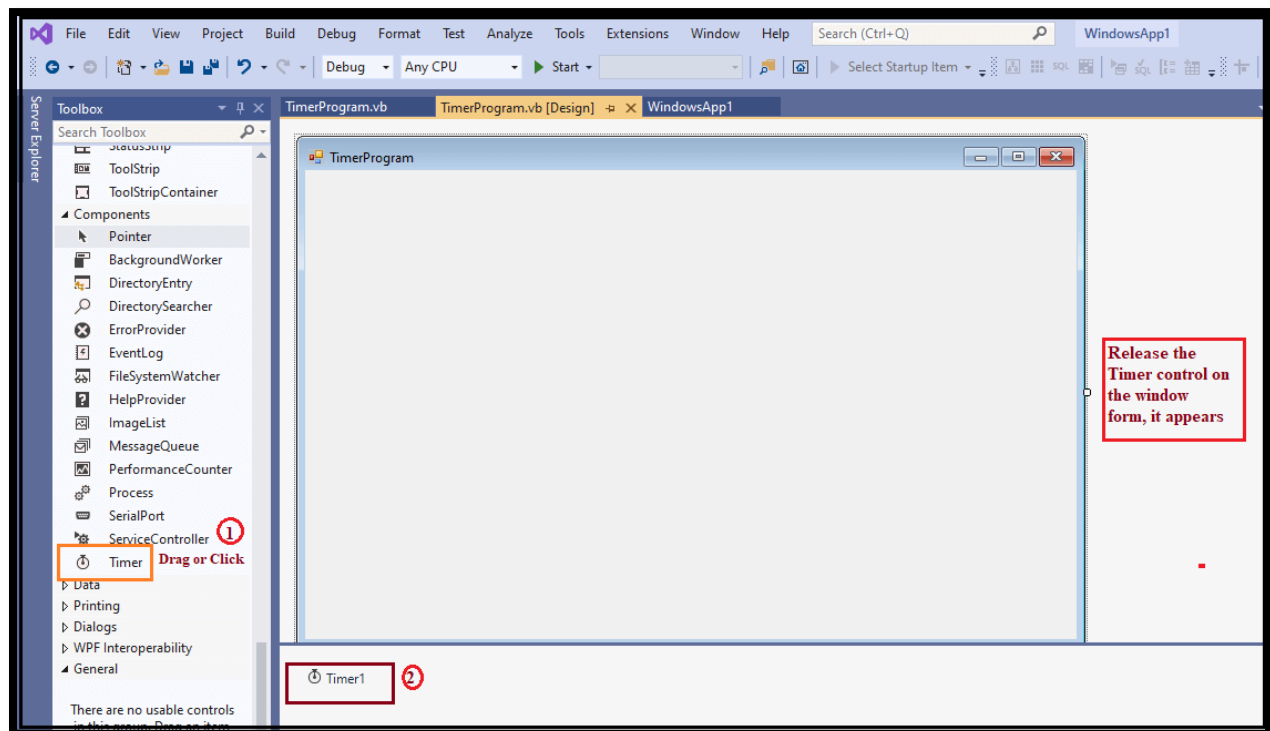


Auto Execution Tool

- These are tools used like timers.

Timer Control in VB.net

- Timer control is a looping control used to repeat any task in a given time interval. It is an important control used in Client-side and Server-side programming, also in Windows Services.
- Furthermore, if we want to execute an application after a specific amount of time, we can use the Timer Control.
- The timer is enabled, it generates a tick event handler to perform any defined task in its time interval property. It starts when the start () method of timer control is called, and it repeats the defined task continuously until the timer stops.





Timer Control Properties in VB.net

Properties	Description
Name	The Name property is used to set the name of the control.
Enabled	The Enables property is used to enable or disable the timer control. By default, it is True.
Interval	An Interval property is used to set or obtain the iteration interval in milliseconds to raise the timer control's elapsed event. According to the interval, a timer repeats the task.
AutoReset	The AutoReset property is used to obtain or set a Boolean value that determines whether the timer raises the elapsed event only once.
Events	Events property are used to get the list of event handler that is associated with Event Component.
CanRaiseEvents	It is used to get a value that represents whether the component can raise an event.

Timer Control Methods in VB.net

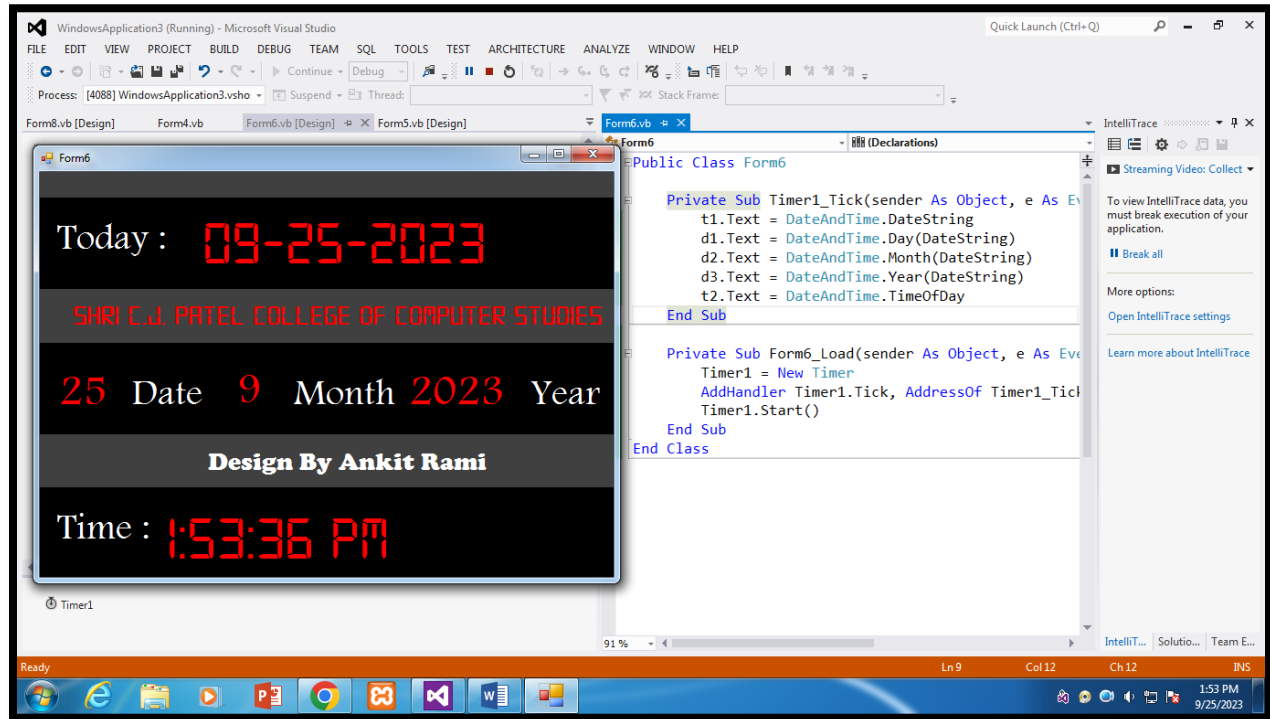
Methods	Description
BeginInt()	The BeginInt() method is used to start run time initialization of a timer control used on a form or by another component.
Dispose()	The Dispose() method is used to free all resources used by the Timer Control or component.
Dispose(Boolean)	It is used to release all resources used by the current Timer control.
Close()	The Close() method is used to release the resource used by the Timer Control.
Start()	The Start() method is used to begin the Timer control's elapsed event by setting the Enabled property to true.
EndInt()	The EndInt() method is used to end the run time initialization of timer control that is used on a form or by another component.
Stop()	The Stop() method is used to stop the timer control's elapsed event by setting Enabled property to false.

Timer Control Events in VB.net

Events	Description
Disposed	When control or component is terminated by calling the Dispose method, a Dispose event occurs.
Elapsed	When the interval elapses in timer control, the Elapsed event has occurred.
Tick	A tick event is used to repeat the task according to the time set in the Interval property. It is the default event of a timer control that repeats the task between the Start() and Stop() methods.

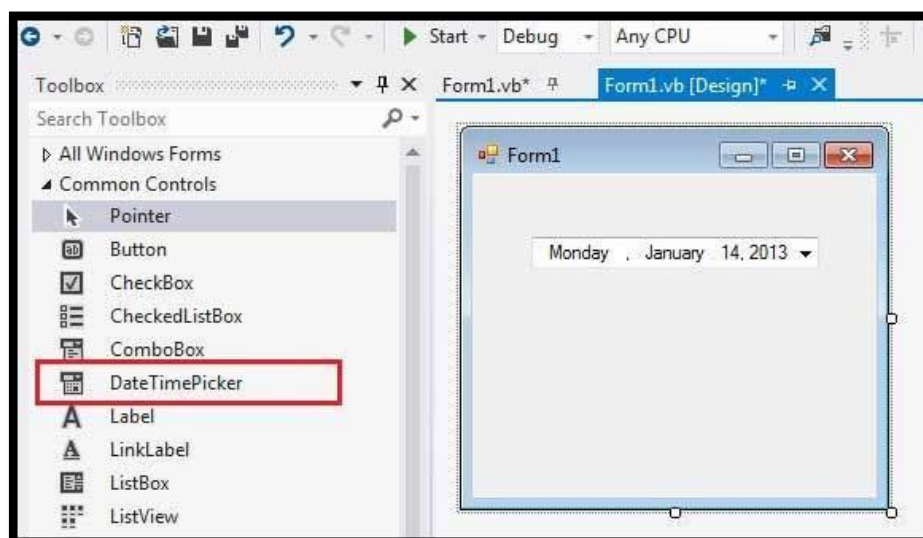


Ex- Timer Control



Date Time Picker Control in VB.net

- DateTimePicker control allows selecting a date and time by editing the displayed values in the control. If you click the arrow in the DateTimePicker control, it displays a month calendar, like a combo box control.



Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193



DateTimePicker Control Properties in VB.net

Sr.No.	Property & Description		
1	BackColor Gets or sets a value indicating the background color of the DateTimePicker control.	13	ForeColor Gets or sets the foreground color of the DateTimePicker control.
2	BackgroundImage Gets or sets the background image for the control.	14	Format Gets or sets the format of the date and time displayed in the control.
3	BackgroundImageLayout Gets or sets the layout of the background image of the DateTimePicker control.	15	MaxDate Gets or sets the maximum date and time that can be selected in the control.
4	CalendarFont Gets or sets the font style applied to the calendar.	16	MaximumDateTime Gets the maximum date value allowed for the DateTimePicker control.
5	CalendarForeColor Gets or sets the foreground color of the calendar.	17	MinDate Gets or sets the minimum date and time that can be selected in the control.
6	CalendarMonthBackground Gets or sets the background color of the calendar month.	18	MinimumDateTime Gets the minimum date value allowed for the DateTimePicker control.
7	CalendarTitleBackColor Gets or sets the background color of the calendar title.	19	PreferredHeight Gets the preferred height of the DateTimePicker control.
8	CalendarTitleForeColor Gets or sets the foreground color of the calendar title.	20	RightToLeftLayout Gets or sets whether the contents of the DateTimePicker are laid out from right to left.
9	CalendarTrailingForeColor Gets or sets the foreground color of the calendar trailing dates.	21	ShowCheckBox Gets or sets a value indicating whether a check box is displayed to the left of the selected date.
10	Checked Gets or sets a value indicating whether the Value property has been set with a valid date/time value and the displayed value is able to be updated.	22	ShowUpDown Gets or sets a value indicating whether a spin button control (also known as an up-down control) is used to adjust the date/time value.
11	CustomFormat Gets or sets the custom date/time format string.	23	Text Gets or sets the text associated with this control.
12	DropDownAlign Gets or sets the alignment of the drop-down calendar on the DateTimePicker control.	24	Value Gets or sets the date/time value assigned to the control.



DateTimePicker Control Methods in VB.net

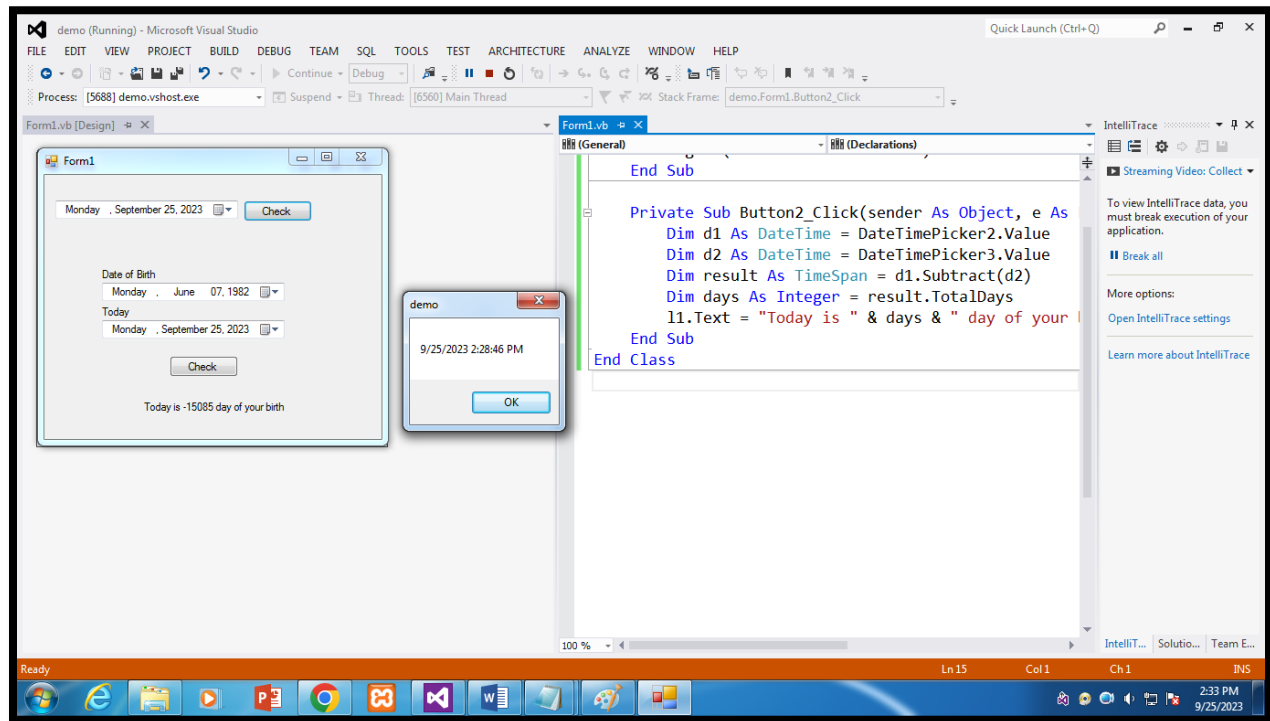
Sr.No.	Method Name & Description
1	ToString Returns the string representing the control.

DateTimePicker Control Events in VB.net

Sr.No.	Event & Description
1	BackColorChanged Occurs when the value of the BackColor property changes.
2	BackgroundImageChanged Occurs when the value of the BackgroundImage property changes.
3	BackgroundImageLayoutChanged Occurs when the value of the BackgroundImageLayout property changes.
4	Click Occurs when the control is clicked.
5	CloseUp Occurs when the drop-down calendar is dismissed and disappears.
6	DoubleClick Occurs when the control is double-clicked.
7	DragDrop Occurs when a drag-and-drop operation is completed.
8	ForeColorChanged Occurs when the value of the ForeColor property changes.
9	FormatChanged Occurs when the Format property value has changed.
10	MouseClick Occurs when the control is clicked with the mouse.
11	MouseDoubleClick Occurs when the control is double-clicked with the mouse.
12	PaddingChanged Occurs when the value of the Padding property changes.
13	Paint Occurs when the control is redrawn.
14	RightToLeftLayoutChanged Occurs when the RightToLeftLayout property changes.
15	TextChanged Occurs when the value of the Text property changes.
16	ValueChanged Occurs when the Value property changes.



Ex- DateTimePicker Control



Design and Develop by Assistant Professor Ankit Rami

Any Query Contact - 8460467193