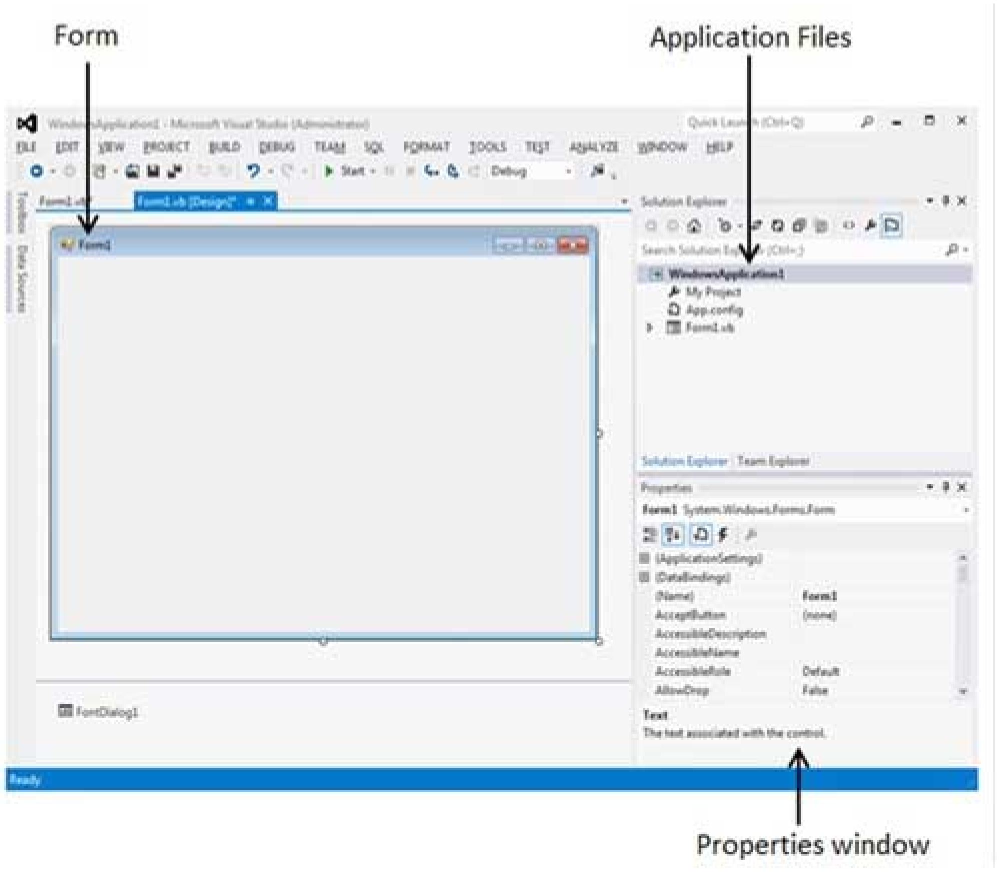
VB.NET - FORMS

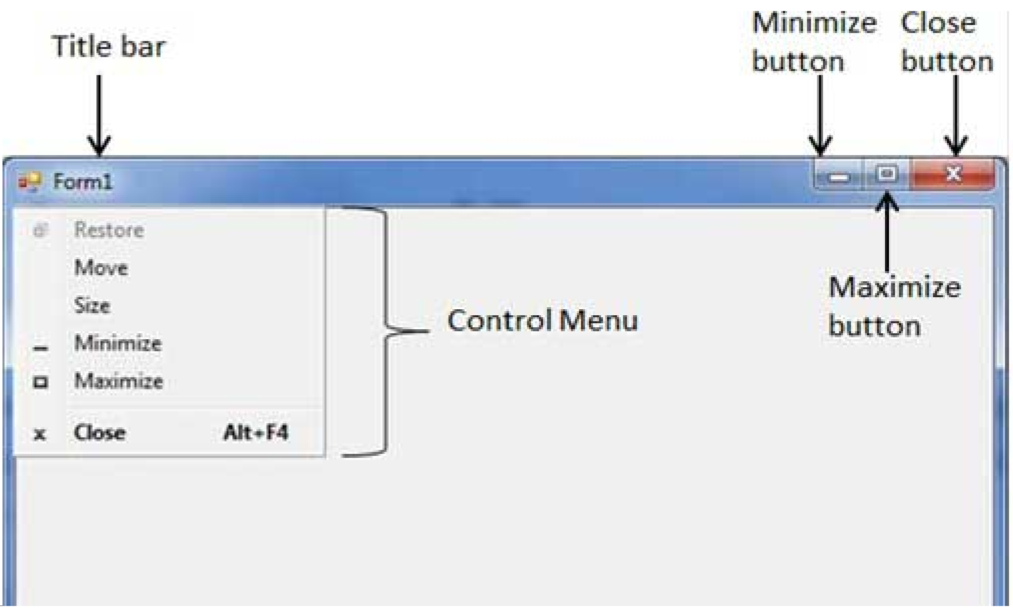
Let's start with creating a Window Forms Application by following the following steps in Microsoft Visual Studio: **File -> New Project -> Windows Forms Applications**

Finally, select OK, Microsoft Visual Studio creates your project and displays following window Form with a name **Form1**.



Visual Basic Form is the container for all the controls that make up the user interface. Every window you see in a running visual basic application is a form, thus the terms form and window describe the same entity. Visual Studio creates a default form for you when you create a **Windows Forms Application**.

Every form will have title bar on which the form's caption is displayed and there will be buttons to close, maximize and minimize the form shown below:



If you click the icon on the top left corner, it opens the control menu, which contains the various commands to control the form like to move control from one place to another place, to maximize or minimize the form or to close the form.

Form Properties

Following table lists down various important properties related to a form. These properties can be set or read during application execution. You can refer to Microsoft documentation for a complete list of properties associated with a Form control:

S.N	Properties	Description
1	**AcceptButton**	The button that's automatically activated when you press Enter, no matter which control has the focus at the time. Usually the OK button on a form is set as AcceptButton for a form.
2	**CancelButton**	The button that's automatically activated when you hit the Esc key. Usually, the Cancel button on a form is set as CancelButton for a form.
3	**AutoScale**	This Boolean property determines whether the controls you place on the form are automatically scaled to the height of the current font. The default value of this property is True. This is a property of the form, but it affects the controls on the form.
4	**AutoScroll**	This Boolean property indicates whether scroll bars will be automatically attached to the form if it is resized to a point that not all its controls are visible.
5	**AutoScrollMinSize**	This property lets you specify the minimum size of the form, before the scroll bars are attached.
6	**AutoScrollPosition**	The AutoScrollPosition is the number of pixels by which the two scroll bars were displaced from their initial locations.
7	**BackColor**	Sets the form background color.
8	**BorderStyle**	The BorderStyle property determines the style of the form's border and the appearance of the form: • **None**: Borderless window that can't be resized. • **Sizable**: This is default value and will be used for resizable window that's used for displaying regular forms. • **Fixed3D**: Window with a visible border, "raised" relative to the main area. In this case, windows can't be resized. • **FixedDialog**: A fixed window, used to create dialog boxes. • **FixedSingle**: A fixed window with a single line border. • **FixedToolWindow**: A fixed window with a Close button only. It looks like the toolbar displayed by the drawing and

		imaging applications. • **SizableToolWindow**: Same as the FixedToolWindow but resizable. In addition, its caption font is smaller than the usual.
9	**ControlBox**	By default, this property is True and you can set it to False to hide the icon and disable the Control menu.
10	**Enabled**	If True, allows the form to respond to mouse and keyboard events; if False, disables form.
11	**Font**	This property specify font type, style, size
12	**HelpButton**	Determines whether a Help button should be displayed in the caption box of the form.
13	**Height**	This is the height of the Form in pixels.
14	**MinimizeBox**	By default, this property is True and you can set it to False to hide the Minimize button on the title bar.
15	**MaximizeBox**	By default, this property is True and you can set it to False to hide the Maximize button on the title bar.
16	**MinimumSize**	This specifies the minimum height and width of the window you can minimize.
17	**MaximumSize**	This specifies the maximum height and width of the window you maximize.
18	**Name**	This is the actual name of the form.
19	**StartPosition**	This property determines the initial position of the form when it's first displayed. It will have any of the following values: • **CenterParent:** The form is centered in the area of its parent form. • **CenterScreen:** The form is centered on the monitor. • **Manual:** The location and size of the form will determine its starting position. • **WindowsDefaultBounds:** The form is positioned at the default location and size determined by Windows. • **WindowsDefaultLocation:** The form is positioned at the Windows default location and has the dimensions you've set at design time.
20	**Text**	The text, which will appear at the title bar of the form.
21	**Top, Left**	These two properties set or return the coordinates of the form's top-left corner in pixels.
22	**TopMost**	This property is a True/False value that lets you specify whether the form will remain on top of all other forms in your application. Its default property is False.
23	**Width**	This is the width of the form in pixel.

Form Methods

The following are some of the commonly used methods of the Form class.You can refer to Microsoft documentation for a complete list of methods associated with forms control:

S.N.	Method Name & Description
1	**Activate** Activates the form and gives it focus.
2	**ActivateMdiChild** Activates the MDI child of a form.
3	**AddOwnedForm** Adds an owned form to this form.
4	**BringToFront** Brings the control to the front of the z-order.
5	**CenterToParent** Centers the position of the form within the bounds of the parent form.
6	**CenterToScreen** Centers the form on the current screen.
7	**Close** Closes the form.
8	**Contains** Retrieves a value indicating whether the specified control is a child of the control.
9	**Focus** Sets input focus to the control.
10	**Hide** Conceals the control from the user.
11	**Refresh** Forces the control to invalidate its client area and immediately redraw itself and any child controls.

12	**Scale*SizeF*** Scales the control and all child controls by the specified scaling factor.
13	**ScaleControl** Scales the location, size, padding, and margin of a control.
14	**ScaleCore** Performs scaling of the form.
15	**Select** Activates the control.
16	**SendToBack** Sends the control to the back of the z-order.
17	**SetAutoScrollMargin** Sets the size of the auto-scroll margins.
18	**SetDesktopBounds** Sets the bounds of the form in desktop coordinates.
19	**SetDesktopLocation** Sets the location of the form in desktop coordinates.
20	**SetDisplayRectLocation** Positions the display window to the specified value.
21	**Show** Displays the control to the user.
22	**ShowDialog** Shows the form as a modal dialog box.

Form Events

Following table lists down various important events related to a form. You can refer to Microsoft documentation for a complete list of events associated with forms control:

S.N	Event	Description
1	**Activated**	Occurs when the form is activated in code or by the user.
2	**Click**	Occurs when the form is clicked.
3	**Closed**	Occurs before the form is closed.
4	**Closing**	Occurs when the form is closing.
5	**DoubleClick**	Occurs when the form control is double-clicked.
6	**DragDrop**	Occurs when a drag-and-drop operation is completed.
7	**Enter**	Occurs when the form is entered.
8	**GotFocus**	Occurs when the form control receives focus.
9	**HelpButtonClicked**	Occurs when the **Help** button is clicked.
10	**KeyDown**	Occurs when a key is pressed while the form has focus.
11	**KeyPress**	Occurs when a key is pressed while the form has focus.
12	**KeyUp**	Occurs when a key is released while the form has focus.
13	**Load**	Occurs before a form is displayed for the first time.
14	**LostFocus**	Occurs when the form loses focus.
15	**MouseDown**	Occurs when the mouse pointer is over the form and a mouse button is pressed.
16	**MouseEnter**	Occurs when the mouse pointer enters the form.
17	**MouseHover**	Occurs when the mouse pointer rests on the form.
18	**MouseLeave**	Occurs when the mouse pointer leaves the form.
19	**MouseMove**	Occurs when the mouse pointer is moved over the form.
20	**MouseUp**	Occurs when the mouse pointer is over the form and a mouse button is released.
21	**MouseWheel**	Occurs when the mouse wheel moves while the control has focus.
22	**Move**	Occurs when the form is moved.
23	**Resize**	Occurs when the control is resized.
24	**Scroll**	Occurs when the user or code scrolls through the client area.
25	**Shown**	Occurs whenever the form is first displayed.
26	**VisibleChanged**	Occurs when the Visible property value changes.

Example:

Following is an example, which shows how we create two buttons at the time of form load event and different properties are being set at the same time.

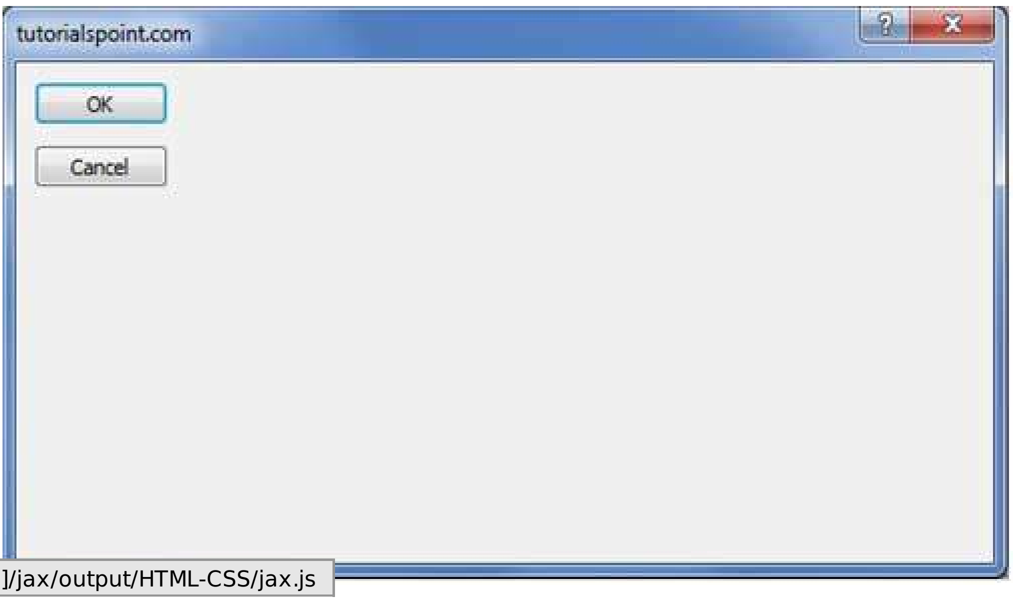
Because **Form1** is being referenced within its own event handler, so it will be written as **Me** instead of using its name, but if we access the same form inside any other control's event handler, then it will be accessed using its name **Form1**.

Let's double click on the Form and put the follow code in the opened window.

```
Public Class Form1

   Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      ' Create two buttons to use as the accept and cancel buttons.
      Dim button1 As New Button()
      Dim button2 As New Button()
      ' Set the text of button1 to "OK".
      button1.Text = "OK"
      ' Set the position of the button on the form.
      button1.Location = New Point(10, 10)
      ' Set the text of button2 to "Cancel".
      button2.Text = "Cancel"
      ' Set the position of the button based on the location of button1.
      button2.Location = _
         New Point(button1.Left, button1.Height + button1.Top + 10)
      ' Set the caption bar text of the form.
      Me.Text = "tutorialspoint.com"
      ' Display a help button on the form.
      Me.HelpButton = True
     ' Define the border style of the form to a dialog box.
      Me.FormBorderStyle = FormBorderStyle.FixedDialog
      ' Set the MaximizeBox to false to remove the maximize box.
      Me.MaximizeBox = False
      ' Set the MinimizeBox to false to remove the minimize box.
      Me.MinimizeBox = False
      ' Set the accept button of the form to button1.
      Me.AcceptButton = button1
      ' Set the cancel button of the form to button2.
      Me.CancelButton = button2
      ' Set the start position of the form to the center of the screen.
      Me.StartPosition = FormStartPosition.CenterScreen
      ' Set window width and height
      Me.Height = 300
      Me.Width = 560
      ' Add button1 to the form.
      Me.Controls.Add(button1)
      ' Add button2 to the form.
      Me.Controls.Add(button2)
   End Sub
End Class
```

When the above code is executed and run using **Start** button available at the Microsoft Visual Studio tool bar, it will show the following window:


tutorialspoint.com
OK
Cancel