

Chapter 38: Working with Windows Forms

Section 38.1: Using the default Form instance

VB.NET offers default Form instances. The developer does not need to create the instance as it is created behind the scenes. However, *it is **not** preferable* to use the default instance all but the simplest programs.

```
Public Class Form1

    Public Sub Foo()
        MessageBox.Show("Bar")
    End Sub

End Class

Module Module1

    Public Sub Main()
        ' Default instance
        Form1.Foo()
        ' New instance
        Dim myForm1 As Form1 = New Form1()
        myForm1.Foo()

    End Sub

End Module
```

See also:

- [Do you have to explicitly create instance of form in VB.NET?](#)
- [Why is there a default instance of every form in VB.Net but not in C#?](#)

Section 38.2: Passing Data From One Form To Another

Sometimes you might want to pass information that has been generated in one form, to another form for additional use. This is useful for forms that display a search tool, or a settings page among many other uses.

Let's say you want to pass a `DataTable` between a form that is already open (*MainForm*) and a new form (*NewForm*):

In The MainForm:

```
Private Sub Open_New_Form()
    Dim newInstanceOfForm As New NewForm(DataTable1)
    newInstanceOfForm.ShowDialog()
End Sub
```

In The NewForm

```
Public Class NewForm
    Dim newDataTable as Datatable

    Public Sub New(PassedDataTable As Datatable)
        InitializeComponent()
        newDataTable= PassedDataTable
    End Sub
```

End Class

Now when the *NewForm* is opened, it is passed `DataTable1` from *MainForm* and stored as `NewDataTable` in *NewForm* for use by that form.

This can be extremely useful when trying to pass large amounts of information between forms, especially when combining all of the information in to a single `ArrayList` and passing the `ArrayList` to the new form.