

# Connection Example

Document ID : Connection\_Example.PDF  
Author : Michele Harris  
Version : 1.1  
Date : 2009-06-19

## *Table of Contents*

---

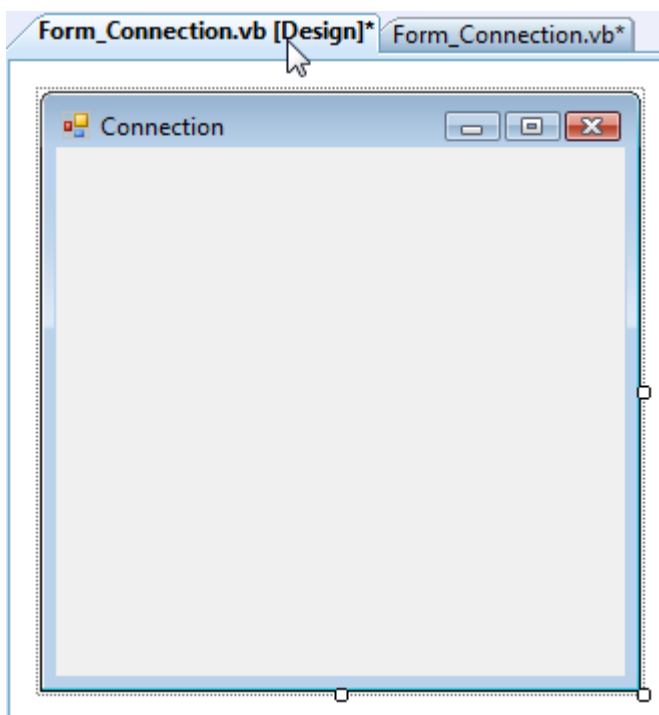
Connection Example.....	2
Adding the Code.....	2
Quick Watch myConnection .....	5
ConnectionString.....	6
ConnectionTimeout.....	6
DataSource.....	6
Provider .....	7
ServerVersion.....	7
State.....	7
Troubleshooting.....	7

---

## Connection Example

Open Visual Studio, click File, and select “New Project.” Visual Studio will prompt you to select a type of project. You’ll want to choose “Windows Form” and name it “Form\_Connection.” Under properties, change your form name to “Connection.”

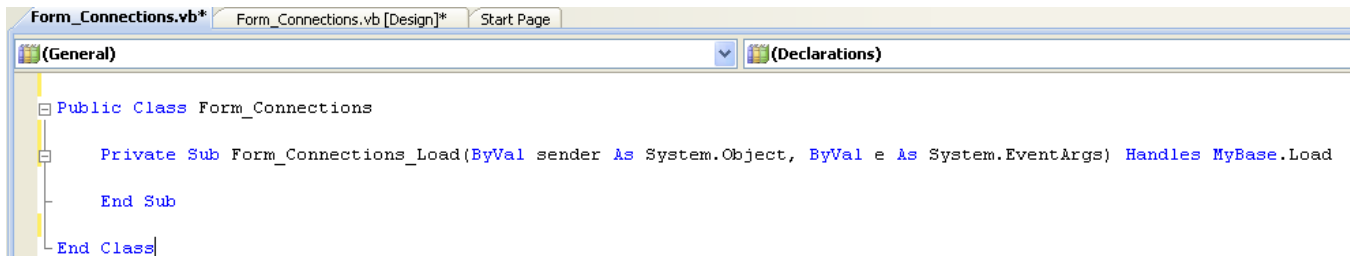
You should begin with a form similar in appearance to the one below.



## Adding the Code

Next, you’ll want to enter the code for this project. To do so, double click the form you created, which will take you to the Code Explorer screen. You’ll notice some

code already there— Next, you'll want to enter the code for this project. To do so, double click the form you created, which will take you to the Code Explorer screen. You'll notice the following code already there.



```
Form_Connections.vb*  Form_Connections.vb [Design]*  Start Page
(General)  (Declarations)
Public Class Form_Connections
    Private Sub Form_Connections_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    End Sub
End Class
```

Take note of this code, because you'll want to integrate yours into the existing framework. I've highlighted the existing code for your convenience. Type the following into the Code Explorer, making sure you reproduce it character for character. Again, the highlighted code should already appear on the screen.

```
Imports System.Data.OleDb

Public Class Form_adapter

    Dim myConnection_string As String

    Dim myAdapter As OleDbDataAdapter

    Dim live_oledb_connection As OleDbConnection

    Private Sub Form_adapter_Load(ByVal sender As System.Object,
                                   ByVal e As System.EventArgs)
                                   Handles MyBase.Load
```

---

```
        live_oledb_connection =  
open_connection_to_access_database()
```

```
End Sub
```

```
-----
```

The above code Imports System.Data.OleDb for use in our program, declares three public variables, and runs a private sub routine that takes place during the form's loading. Next, we'll want to enter our first function. Type the following code right below the line "End Sub".

```
Function open_connection_to_access_database()  
    Dim myConnection As OleDbConnection  
    myConnection_string = make_connection_string()  
    myConnection = New OleDbConnection(myConnection_string)  
    myConnection.Open()  
    Return myConnection  
End Function
```

```
-----
```

Finally, add the following code, which will complete your program. Again, be sure to type everything character for character, and not to retype any highlighted code.

```
Function make_connection_string()
```

---

```
Dim connection_string As String
Dim provider As String
Dim data_source As String
Dim security As String
provider = "Provider=Microsoft.jet.oledb.4.0;"
data_source = "Data Source=c:\slm.mdb;"
security = "Persist Security Info=false;"
connection_string = provider & data_source & security
Return connection_string
End Function
End Class
```

---

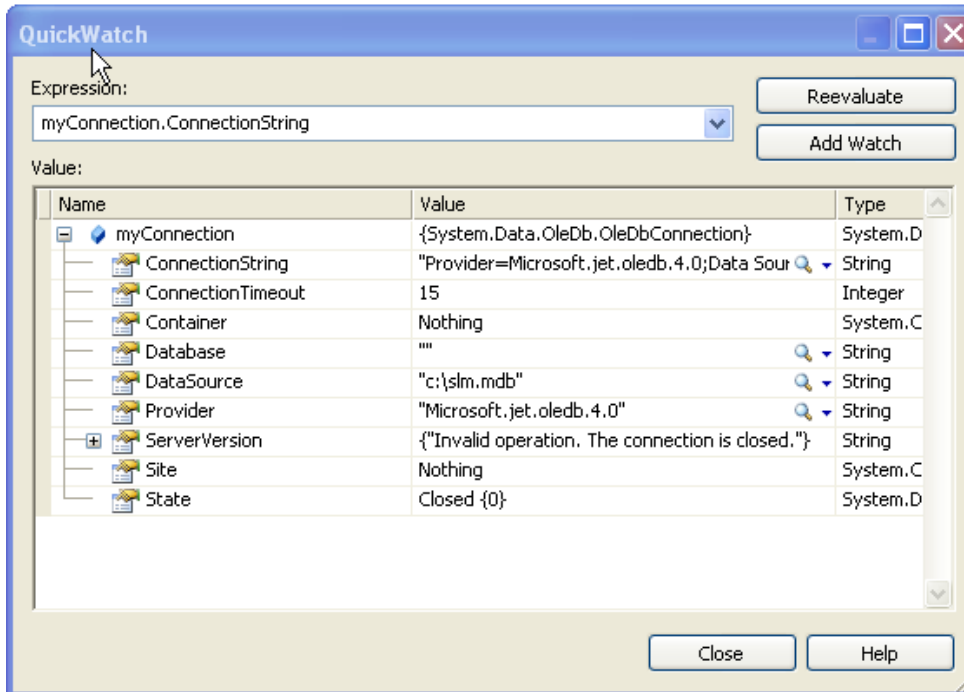
If after typing this code you encounter any errors that prevent you from debugging, compare each line of code you wrote to those above, and fix any discrepancies. If you're still unable to debug and are sure your code is correct, see **Troubleshooting**.

## Quick Watch myConnection

Once you've finished programming your form, press F8 to step through each line of code. In particular, take the opportunity to "Quick Watch" the "myConnection" variable. Highlight "myConnection" as it's declared in the first function, right click it, and select "Quick Watch." Once the Quick Watch

---

form displays, expand “myConnection” by clicking the “+” sign to the left of its name. These actions should produce the screen you see below.



## ConnectionString

The connection string is what links the Access database to our VB.NET Windows program, by means of a Connection Object.

## ConnectionTimeout

The connection timeout is no misnomer—it’s merely the amount of time in seconds until a connection would “time out.”

## DataSource

The Data Source is merely the location of our database—in this case, it’s C:\slm.mdb.

## Provider

This argument is the Ole.DB provider used for Access databases.

## ServerVersion

Again, no misnomer here—it's merely the version of the SQL server we're using.

## State

State let's us know whether our database connection is opened or closed.

## Troubleshooting

You might encounter a few error messages while trying to Debug Form\_Connection.

If so, review the code you typed above to ensure that the error isn't the result of a simple misspelling or omission. If after doing this your error still persists—and the error is linked to the myConnection.Open procedure, as shown below—double check to make sure the **slm.mdb** file has been copied to your C drive. If it hasn't, retrieve **slm.mdb** from [\\lynx\c](http://lynx/c) and copy it to your computer's C drive.

---

