

Combobox Design Tool in Visual Studio using Visual Basic

```

' Variables used within the subroutine are defined here. The combobox will have a start-up
' instruction for the operator.
    Dim combobox1_TitleText As String = "Select Project"
' The routine will also use two (2) collections, here defined outside the routine.
    Dim cb1_IDCollection As New Collection ' Dim defines the variable as used only in the form it is
declared in. Contrary, Public will permit the variable to be used in other forms within the project.
    Public cb1_NoCollection As New Collection
' Private is a term noting the subroutine (or routine) will only be accessible within the form it is
' defined in. If the routine was defined as Public, it could be used in other forms. Public assignments
' are useful as the routine would not have to be copied to another form and thusly making the program
' longer than necessary.
    Private Sub ComboBox1_Populate() ' The subroutine is named so it can be called when needed.
        ' Normalize variables on every call of the subroutine
        cb1_IDCollection.Clear() ' The collection is provide a name to identify it with the
        cb1_NoCollection.Clear() ' handshake interface tool, or the Design
        ' With project selection object, ComboBox1, begin to define its settings.
        With ComboBox1
            ' Set with title text
            .Text = combobox1_TitleText
            ' Clear any existing items from the list box - for refresh requests
            .Items.Clear()
            ' Fill the Combobox to display on the form tab
            .BeginUpdate()
            ' Establish the SQL command
            Dim cmd As New SqlCommand
            Dim sqlDataReader As SqlDataReader ' and the use of the SQL DataReader
            ' Establish the command text to collect Current Projects
            cmd.CommandText = "SELECT p.ProjectNoID, p.ProjectNo FROM dbo.ProjectDataSheet AS p WHERE
                               p.IsCurrent = 'True' ORDER BY p.ProjectNo DESC"
            ' Set the command text to the command type
            cmd.CommandType = CommandType.Text
            ' Establish the connection
            cmd.Connection = sqlConnR
            Try ' Attempt to open the connection to the database
                sqlConnR.Open()
                ' Execute the command AND retrieve the rows in the DataReader
                sqlDataReader = cmd.ExecuteReader()
                ' If the reader has found any rows, then
                If sqlDataReader.HasRows Then
                    ' read the rows.
                    Do While sqlDataReader.Read()
                        ' Add the row data Project Number to the Combobox
                        .Items.Add(ReturnSQLInteger(sqlDataReader, "ProjectNo"))
                        ' and add the Primary Key (PK) to a collection.
                        cb1_IDCollection.Add(ReturnSQLInteger(sqlDataReader, "ProjectNoID"))
                        ' Also, just for convenience, add the Project Number to a collection
                        cb1_NoCollection.Add(ReturnSQLInteger(sqlDataReader, "ProjectNo"))
                    Loop ' continue iterating the loop to the end of the returned records.
                End If ' End reading the row
            Catch ex As Exception ' Inform operator of error if the database cannot be accessed
                ' or a fault occurs reading the data.
                MessageBox(ex.Message.ToString)
            Finally ' Close the connection to the database
                sqlConnR.Close()
            End Try
            ' End the combobox update
            .EndUpdate()
        End With ' End the update to the ComboBox
    End Sub ' End the work with the subroutine.

```

Description of the Subroutine ComboBox1_Populate

This is a code-controlled reader, as shown with the “sqlConnR.Open()” and “sqlConnR.Close” lines. There are at least two other methods to access the database information, but the above was chosen as it eventually became the easiest and most streamlined to use.

The SQL connection will be discussed in the Programming menu item.