**IN BASDECLARES:**

Public DocTypeForGuide As String

Public isCM\_LeadEng\_Admin As String

Public isLeadEngineer As String

Public MDLNumberCreatedFormB As String

Public CreateOrModifyDocNum As String

Public RecordIdDocNumToModify As String

Public DocNumberDocNumToModify As String

Public DocTypeDocNumToModify As String

Public LocationDocNumToModify As String

Public SystemDocNumToModify As String

Public TitleDocNumToModify As String

Public ProgramRealDocNumToModify As String

Public ProgramDocNumToModify As string 'This actually will contain Project

Public ClassificationDocNumToModify As String

Public DocTypeSelectedLevelThree\_DocNumToModify As String

Public ResourceTypeDocNumToModify As String

Public NewReworkFormB As String

Public ClassificationFormB As String

Public PrefixForCheckingReset As String

Public DrafterFormB As String

Public ProjectSelected As String 'in Projects screen

**IN frmEngWorkflow:**

**DONE - Under frmEngWorkflow\_Load:**

'Check if it is CM/ADMIN/LeadEng and NewRecord is true. To use when showing or not the Number button

If IsDBNull(ds.Tables("USERAUTHENTICATE").Rows(0).Item("LeadEngineer")) Then

isLeadEngineer = "False"

ElseIf ds.Tables("USERAUTHENTICATE").Rows(0).Item("LeadEngineer") = "False" Then

isLeadEngineer = "False"

Else

isLeadEngineer = "True"

End If

If (CM = True Or ds.Tables("USERAUTHENTICATE").Rows(0).Item("Admin") = "T" Or isLeadEngineer = "True") And isLeadEngineer = "True" And isNewRecord = "True" Then

isCM\_LeadEng\_Admin = "True"

'btnAdd.Visible = False

Else

isCM\_LeadEng\_Admin = "False"

'btnNumber.Visible = False

End If

**DONE - Under txtMainInfoRecordID:**

'Check if document number has been confirmed in Form B. If it is hide the Number button (only for lead engineers, CMs and Admins)

'Also disable the Status drop-down as the user cannot move the status to anything else when it is at gate zero: record is set to gate zero in Form A and gate 1 in Form B. Record can be moved to Gate 1 only after signoff on Form B

IsDocNumConfirmed = ""

If IsDBNull(ds.Tables("USERAUTHENTICATE").Rows(0).Item("NewRecord")) Then

isNewRecord = "False"

ElseIf ds.Tables("USERAUTHENTICATE").Rows(0).Item("NewRecord") = 0 Then

isNewRecord = "False"

Else

isNewRecord = "True"

End If

'If (CM = True Or ds.Tables("USERAUTHENTICATE").Rows(0).Item("Admin") = "T" Or isLeadEngineer = "True") And isNewRecord = "True" Then

If isCM\_LeadEng\_Admin = "True" Then

da = New SqlClient.SqlDataAdapter("Select IsDocNumConfirmed, Status From tblDrawings where RecordID = '" & Val(Me.txtMainInfoRecordID.Text) & "'", SqlConnection1)

da.Fill(ds, "DocConfirmed")

If ds.Tables("DocConfirmed").Rows.Count > 0 Then

If IsDBNull(ds.Tables("DocConfirmed").Rows(0).Item("IsDocNumConfirmed")) And ds.Tables("DocConfirmed").Rows(0).Item("Status") <> "GATE 0: CM Needs To Assign DWG Number" Then

btnNumber.Visible = False

IsDocNumConfirmed = "Yes"

ElseIf IsDBNull(ds.Tables("DocConfirmed").Rows(0).Item("IsDocNumConfirmed")) And ds.Tables("DocConfirmed").Rows(0).Item("Status") = "GATE 0: CM Needs To Assign DWG Number" Then

btnNumber.Visible = True

IsDocNumConfirmed = "No"

cboMainStatus.Enabled = False

ElseIf ds.Tables("DocConfirmed").Rows(0).Item("IsDocNumConfirmed") = False And ds.Tables("DocConfirmed").Rows(0).Item("Status") = "GATE 0: CM Needs To Assign DWG Number" Then

btnNumber.Visible = True

IsDocNumConfirmed = "No"

cboMainStatus.Enabled = False

ElseIf ds.Tables("DocConfirmed").Rows(0).Item("IsDocNumConfirmed") = False And ds.Tables("DocConfirmed").Rows(0).Item("Status") <> "GATE 0: CM Needs To Assign DWG Number" Then

btnNumber.Visible = False

IsDocNumConfirmed = "Yes"

ElseIf ds.Tables("DocConfirmed").Rows(0).Item("IsDocNumConfirmed") = True Then

btnNumber.Visible = False

IsDocNumConfirmed = "Yes"

Else

btnNumber.Visible = True

IsDocNumConfirmed = "No"

End If

End If

ds.Tables("DocConfirmed").Clear()

ds.Tables("DocConfirmed").Dispose()

Else

btnNumber.Visible = False

End If

**DONE - Under the btnAdd:**

'If allowed user and is New Program then open Form B, otherwise Form A

If isCM\_LeadEng\_Admin = "True" Then 'And IsDocNumConfirmed = "No"

CreateOrModifyDocNum = "Create"

Dim frmEngWorkflowAddFormB As New frmEngWorkflowAddFormB

frmEngWorkflowAddFormB.ShowDialog()

Else

Dim frmEngWorkflowAdd As New frmEngWorkflowAdd

frmEngWorkflowAdd.ShowDialog()

End If

Search()

**DONE - Under the btnNumber:**

CreateOrModifyDocNum = "Modify"

'GET RECORD ID, LOCATION, DOCUMENT TYPE AND CLASSIFICATION OF THE RECORD THAT WILL BE MODIFIED IN FORM B

Dim a As Integer = 0

Dim selectedRowsForUsingInFormB() As Integer = GridView2.GetSelectedRows()

For Each rowHandleTwo As Integer In selectedRowsForUsingInFormB

If rowHandleTwo >= 0 Then

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colRecordID)) Then

RecordIdDocNumToModify = ""

Else

RecordIdDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colRecordID)

End If

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colDocumentType)) Then

DocTypeDocNumToModify = ""

Else

DocTypeDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colDocumentType)

End If

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colLocation)) Then

LocationDocNumToModify = ""

Else

LocationDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colLocation)

End If

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colSystemDescription)) Then

SystemDocNumToModify = ""

Else

SystemDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colSystemDescription)

End If

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colDrawingTitle)) Then

TitleDocNumToModify = ""

Else

TitleDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colDrawingTitle)

End If

If IsDBNull(GridView3.GetRowCellValue(rowHandleTwo, colProgram)) Then

ProgramDocNumToModify = ""

Else

ProgramDocNumToModify = GridView3.GetRowCellValue(rowHandleTwo, colProgram) 'This will actually be Project(contain project)

End If

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colNewRework)) Then

ClassificationDocNumToModify = ""

Else

ClassificationDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colNewRework)

End If

If ClassificationDocNumToModify = "Rework Drawing" Then

NewReworkFormB = "New Drawing"

ClassificationDocNumToModify = "Drawing"

ElseIf ClassificationDocNumToModify = "New Drawing" Then

NewReworkFormB = "New Drawing"

ClassificationDocNumToModify = "Drawing"

ElseIf ClassificationDocNumToModify = "Rework Report" Then

NewReworkFormB = "New Report"

ClassificationDocNumToModify = "Report"

ElseIf ClassificationDocNumToModify = "New Report" Then

NewReworkFormB = "New Report"

ClassificationDocNumToModify = "Report"

End If

If IsDBNull(GridView2.GetRowCellValue(rowHandleTwo, colDrawingNumber)) Then

DocNumberDocNumToModify = ""

Else

DocNumberDocNumToModify = GridView2.GetRowCellValue(rowHandleTwo, colDrawingNumber)

End If

'Get Prefix. To use in the ResetSequence in Form B

If ClassificationDocNumToModify = "Drawing" And (LocationDocNumToModify = "Calgary" Or LocationDocNumToModify = "Toronto") Then

PrefixForCheckingReset = DocNumberDocNumToModify.Substring(DocNumberDocNumToModify.Length - 5)

Else

PrefixForCheckingReset = DocNumberDocNumToModify.Substring(DocNumberDocNumToModify.Length - 4)

''Get only numbers

'Dim nums As StringBuilder = New StringBuilder()

'For Each c As Char In PrefixForCheckingReset

' If (Char.IsNumber(c)) Then

' nums.Append(c)

' End If

'Next

'PrefixForCheckingReset = nums.ToString

End If

End If

Next rowHandleTwo

'Get the selection that was made by the user for Level 3. To use in the ResetOrNot function. Also get the Resource Type to use in Form B in case modifying an existing doc number

da = New SqlClient.SqlDataAdapter("Select DocTypeSelectedLevelThree, Type From tblDrawings Where RecordID = '" & RecordIdDocNumToModify & "'", SqlConnection1)

da.Fill(ds, "DocTypeSelectedLevelThree")

If ds.Tables("DocTypeSelectedLevelThree").Rows.Count > 0 Then

DocTypeSelectedLevelThree\_DocNumToModify = ds.Tables("DocTypeSelectedLevelThree").Rows(0).Item("DocTypeSelectedLevelThree")

ResourceTypeDocNumToModify = ds.Tables("DocTypeSelectedLevelThree").Rows(0).Item("Type")

End If

ds.Tables("DocTypeSelectedLevelThree").Clear()

ds.Tables("DocTypeSelectedLevelThree").Dispose()

'Only allow to modify the selected record if user has permission And record has not been confirmed yet

If isCM\_LeadEng\_Admin = "True" And IsDocNumConfirmed = "No" Then

CreateOrModifyDocNum = "Modify"

Dim frmEngWorkflowAddFormB As New frmEngWorkflowAddFormB

frmEngWorkflowAddFormB.ShowDialog()

Search()

End If

**DONE - UNDER FrmEngWorkflow.vb FILE:**

Private Sub btnShowTree\_Click(sender As Object, e As EventArgs) Handles btnShowTree.Click

Dim frmEngWorkflowTree As New frmEngWorkflowTree

frmEngWorkflowTree.ShowDialog()

End Sub

**DONE - ALL THE CODE UNDER THESE TWO NEW FORMS:**

DONE - frmEngWorkflowAdd

DONE - frmEngWorkflowAddFormB

**DONE - UNDER THE TWO MENU ITEMS IN ADMIN/LOV MAINTENANCE:**

Private Sub JASCCodesToolStripMenuItem\_Click(sender As Object, e As EventArgs) Handles JASCCodesToolStripMenuItem.Click

Dim frmEngWorkflowJASCCodes As New frmEngWorkflowJASCCodes

frmEngWorkflowJASCCodes.ShowDialog()

End Sub

Private Sub Level3DocTypeToolStripMenuItem\_Click(sender As Object, e As EventArgs) Handles Level3DocTypeToolStripMenuItem.Click

Dim frmEngWorkflowLevelThreeDocType As New frmEngWorkflowLevelThreeDocType

frmEngWorkflowLevelThreeDocType.ShowDialog()

End Sub

**DONE - IN frmEngWorkflowGuide: (THIS IS A NEW FORM)**

Dim da As SqlClient.SqlDataAdapter

Dim ds As New DataSet

Private Sub frmEngWorkflowGuide\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

SqlConnection1.ConnectionString = sqlString

If DocTypeForGuide = "" Then

da = New SqlClient.SqlDataAdapter("Select RecordID, DocumentType, Classification, Notes From tblDocumentType Order By Classification", SqlConnection1)

ElseIf DocTypeForGuide = "Drawing" Then

da = New SqlClient.SqlDataAdapter("Select RecordID, DocumentType, Classification, Notes From tblDocumentType Where Classification = 'Drawing'", SqlConnection1)

ElseIf DocTypeForGuide = "Report" Then

da = New SqlClient.SqlDataAdapter("Select RecordID, DocumentType, Classification, Notes From tblDocumentType Where Classification = 'Report'", SqlConnection1)

End If

da.Fill(ds, "Guide")

Me.bsGuide.DataSource = ds.Tables("Guide")

GridControl1.DataSource = Me.bsGuide

End Sub

**DONE - IN frmEngWorkflowJASCCodes: (THIS IS A NEW FORM)**

Imports DevExpress.XtraGrid.Views.Base

Public Class frmEngWorkflowJASCCodes

'Dim da As SqlClient.SqlDataAdapter

'Dim ds As New DataSet

Private Sub frmEngWorkflowJASCCodes\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

'SqlConnection1.ConnectionString = sqlString

'da = New SqlClient.SqlDataAdapter("Select RecordID, SectionNumAndDesc, JASCAndDesc, JASCPrefix From tblDocTypeJASC\_Series", SqlConnection1)

'da.Fill(ds, "JascCodes")

'Me.bsJASCCodes.DataSource = ds.Tables("JascCodes")

'GridControl1.DataSource = Me.bsJASCCodes

SqlConnection1.ConnectionString = sqlString

Search()

End Sub

Public Function Search()

Dim SqlString As String

Me.DsJASCCodes1.tblDocTypeJASC\_Series.Clear()

Me.txtRecordID.DataBindings.Clear()

Dim cmdSave As String

cmdSave = Me.SqlDataAdapter1.SelectCommand.CommandText

Me.SqlDataAdapter1.SelectCommand.CommandText() += SqlString

Try

Me.SqlDataAdapter1.Fill(DsJASCCodes1.tblDocTypeJASC\_Series)

Catch ex As Exception

MsgBox(ex.ToString)

End Try

Me.SqlDataAdapter1.SelectCommand.CommandText = cmdSave

If DsJASCCodes1.Tables("tblDocTypeJASC\_Series").Rows.Count <> 0 Then

Search = True

'Create DataBindings

Me.txtRecordID.DataBindings.Add("Text",

Me.bsJASCCodes, "RecordID")

End If

End Function

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles btnSave.Click

Me.SqlDataAdapter1.Update(DsJASCCodes1.tblDocTypeJASC\_Series)

Search()

End Sub

Private Sub btnDelete\_Click(sender As Object, e As EventArgs) Handles btnDelete.Click

u.ExecuteSQL(SqlConnection1, "Delete from tblDocTypeJASC\_Series Where RecordID = '" & Val(Me.txtRecordID.Text) & "'")

Search()

End Sub

Private Sub GridView1\_BeforeLeaveRow(sender As Object, e As RowAllowEventArgs) Handles GridView1.BeforeLeaveRow

Me.SqlDataAdapter1.Update(DsJASCCodes1.tblDocTypeJASC\_Series)

Search()

End Sub

End Class

**DONE - IN frmEngWorkflowLevelThreeDocType: (THIS IS A NEW FORM)**

Imports DevExpress.XtraGrid.Views.Base

Public Class frmEngWorkflowLevelThreeDocType

Private Sub frmEngWorkflowLevelThreeDocType\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

SqlConnection1.ConnectionString = sqlString

Search()

End Sub

Public Function Search()

Dim SqlString As String

DsLevelThree1.tblDocTypePrefixLevelThree.Clear()

Me.txtRecordID.DataBindings.Clear()

Dim cmdSave As String

cmdSave = Me.SqlDataAdapter1.SelectCommand.CommandText

Me.SqlDataAdapter1.SelectCommand.CommandText() += SqlString

Try

Me.SqlDataAdapter1.Fill(DsLevelThree1.tblDocTypePrefixLevelThree)

Catch ex As Exception

MsgBox(ex.ToString)

End Try

Me.SqlDataAdapter1.SelectCommand.CommandText = cmdSave

If DsLevelThree1.Tables("tblDocTypePrefixLevelThree").Rows.Count <> 0 Then

Search = True

'Create DataBindings

Me.txtRecordID.DataBindings.Add("Text",

bsLevelThree, "RecordID")

End If

End Function

Private Sub btnSave\_Click(sender As Object, e As EventArgs) Handles btnSave.Click

Me.SqlDataAdapter1.Update(DsLevelThree1.tblDocTypePrefixLevelThree)

Search()

End Sub

Private Sub btnDelete\_Click(sender As Object, e As EventArgs) Handles btnDelete.Click

u.ExecuteSQL(SqlConnection1, "Delete from tblDocTypePrefixLevelThree Where RecordID = '" & Val(Me.txtRecordID.Text) & "'")

Search()

End Sub

Private Sub GridView1\_BeforeLeaveRow(sender As Object, e As RowAllowEventArgs) Handles GridView1.BeforeLeaveRow

Me.SqlDataAdapter1.Update(DsLevelThree1.tblDocTypePrefixLevelThree)

Search()

End Sub

End Class

**DONE - IN frmEngWorkflowSignOff:**

ElseIf Gate = 7 Then 'Planner, CM Administrator and Lead Engineer

da = New SqlClient.SqlDataAdapter("Select [User] from tblNotificationsGroups Where (Gate0 = 1 And (Role = 'Configuration Management' Or Role = 'Lead Engineer' Or Role = 'Planner') And Program = '" & ProgramName & "' And Active = 1) Or (Gate0 = 1 And Program = '" & ProgramName & "' And Active = 1)", SqlConnection1)

Else

GoTo GETEMAILADDRESSES

End If

**DONE - And this under the same form(**frmEngWorkflowSignOff):

ElseIf Gate = 7 Then 'Planner, CM Administrator and Lead Engineer

da = New SqlClient.SqlDataAdapter("Select Email from tblUsers Where (ID IN(" & ListOfUserIDs & "))", SqlConnection1)

EmailBody = "You have a record to sign off " & DrawingNumForEmails & " " & RevisionForEmails 'Document

EmailSubject = "Sign Off Pending " & DrawingNumForEmails & " " & RevisionForEmails

ElseIf Gate = 8 Then 'Drafter email for Form B, when creating a new document

da = New SqlClient.SqlDataAdapter("Select Email from tblUsers Where ID = '" & DrafterFormB & "' And Drafter = 'T'", SqlConnection1)

EmailBody = "You have a record to sign off " & DrawingNumForEmails & " " & RevisionForEmails 'Document

EmailSubject = "Sign Off Pending " & DrawingNumForEmails & " " & RevisionForEmails

**IN Programs:**

**DONE - Under Programs\_Load Sub this is the entire code:**

AddNew = "No"

SqlConnection1.ConnectionString = sqlString

SqlConnection2.ConnectionString = sqlString

Search()

**DONE - Under the AddNew button, this is the entire code:**

Dim ProductGroupAdd As String = ""

If CM = True Or ds.Tables("USERAUTHENTICATE").Rows(0).Item("Admin") = "T" Then

'Nothing

Else

MsgBox("You are not an authorized user")

Exit Sub

End If

AddNew = "Yes"

If cbAddDocNumbering.Text = "Field Aerospace (Canada)" Then

If cbAddDocNumbering.Text = "" Or txtAddProjNo.Text = "" Then

MsgBox("Select Doc Numbering and Project Number")

Exit Sub

End If

ProductGroupAdd = ""

Else

If cbAddDocNumbering.Text = "" Or cbAddProductGroup.Text = "" Or txtAddProjNo.Text = "" Then

MsgBox("Select Doc Numbering, Product Group and Project Number")

Exit Sub

End If

ProductGroupAdd = cbAddProductGroup.Text

End If

u.ExecuteSQL(SqlConnection1, "Insert into tblProgram (ProgramName, Active, Standard, Legacy, ProjectNumber, ActNo, PDFFolder, ShortForm, DocNumbering, ProductGroup) Values ('" & u.FTM(Me.txtAddProjDesc.Text) & "', 'True', '" & optStd.Checked & "', '" & optLegacy.Checked & "', '" & Val(Me.txtAddProjNo.Text) & "', '', '', '', '" & cbAddDocNumbering.Text & "', '" & ProductGroupAdd & "')")

u.ExecuteSQL(SqlConnection1, "Insert into tblProgramCertification (Program, FAAProjectNo, LeadCertEngineer) Values ('" & u.FTM(Me.txtAddProjDesc.Text) & "', '', '')")

Search()

AddNew = "No"

txtAddProjDesc.Text = ""

txtAddProjNo.Text = ""

cbAddDocNumbering.SelectedIndex = -1

cbAddProductGroup.SelectedIndex = -1

**DONE - Under btnSaveProject:**

Dim ProjectNum As String = ""

Dim ACTNum As String = ""

Dim ShortForm As String = ""

Dim PDFFolder As String = ""

Dim Active As String = ""

If CM = True Or ds.Tables("USERAUTHENTICATE").Rows(0).Item("Admin") = "T" Then

'Nothing

Else

MsgBox("You are not an authorized user")

Exit Sub

End If

'WHEN SAVING GET THE ROW SELECTED FROM THE PROJECTS GRID, SO THAT I CAN HAVE IT SELECTED AFTER ALL THE SAVE CODE IS EXECUTED

RowSelectedProjectsGrid = GridView1.FocusedRowHandle

If txtProjectNumber.Text = "" Then

MsgBox("Please enter the Project Number")

Exit Sub

End If

ProjectNum = txtProjectNumber.Text

ACTNum = txtActNumber.Text

ShortForm = txtShortForm.Text

PDFFolder = txtPDFFolder.Text

Active = cboActive.Text

'Updates to the Project Number fields

u.ExecuteSQL(SqlConnection1, "Update tblProgram Set ProjectNumber = '" & ProjectNum & "', ACTNo = '" & ACTNum & "', ShortForm = '" & ShortForm & "', PDFFolder = '" & PDFFolder & "', Active = '" & Active & "', DocNumbering = '" & cboDocNumberingSave.Text & "', ProductGroup = '" & cboProductGroupSave.Text & "' Where RecordID = '" & txtRecordID.Text & "'")

AddNew = "Yes"

Search()

AddNew = "No"

'Put the focus on the row that was selected, for the Projects grid, before hitting the save button

GridView1.FocusedRowHandle = RowSelectedProjectsGrid

GridView1.SelectRow(RowSelectedProjectsGrid)

**DONE - Under Search()**

Me.cboDocNumberingSave.DataBindings.Clear()

Me.cboProductGroupSave.DataBindings.Clear()

Me.cbAddDocNumbering.DataBindings.Clear()

Me.cbAddProductGroup.DataBindings.Clear()

Me.cboDocNumberingSave.DataBindings.Add("Text", Me.bsPrograms, "DocNumbering")

Me.cboProductGroupSave.DataBindings.Add("Text", Me.bsPrograms, "ProductGroup")

Me.cbAddDocNumbering.DataBindings.Add("Text", Me.bsPrograms, "DocNumbering")

Me.cbAddProductGroup.DataBindings.Add("Text", Me.bsPrograms, "ProductGroup")

**DONE - Add this (same form, Programs):**

Private Sub cbAddDocNumbering\_TextChanged(sender As Object, e As EventArgs) Handles cbAddDocNumbering.TextChanged

'Doc Numbering and Product Group is not visible when Canada

If cbAddDocNumbering.Text = "Field Aerospace (Canada)" Then

cbAddProductGroup.Visible = False

lblProductGroupAdd.Visible = False

cboProductGroupSave.Visible = False

lblProductGroupSave.Visible = False

Else

cbAddProductGroup.Visible = True

lblProductGroupAdd.Visible = True

cboProductGroupSave.Visible = True

lblProductGroupSave.Visible = True

End If

End Sub

'This is so that the user cannot free type text in the combobox

Private Sub cboDocNumberingSave\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles cboDocNumberingSave.KeyPress

e.Handled = True

End Sub

'This is so that the user cannot free type text in the combobox

Private Sub cboProductGroupSave\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles cboProductGroupSave.KeyPress

e.Handled = True

End Sub

Private Sub cboProductGroupSave\_Click(sender As Object, e As EventArgs) Handles cboProductGroupSave.Click

Me.cboProductGroupSave.Items.Clear()

'Populate Product Group drop-down for saving

da = New SqlClient.SqlDataAdapter("Select \* from tblProductGroup Order By ProductGroup", SqlConnection1)

da.Fill(ds, "ProductGroup")

Dim p As Integer = 0

While p <= ds.Tables("ProductGroup").Rows.Count - 1

Me.cboProductGroupSave.Items.Add(ds.Tables("ProductGroup").Rows(p).Item("ProductGroup"))

p = p + 1

End While

ds.Tables("ProductGroup").Clear()

ds.Tables("ProductGroup").Dispose()

End Sub

Private Sub cbAddProductGroup\_Click(sender As Object, e As EventArgs) Handles cbAddProductGroup.Click

Me.cbAddProductGroup.Items.Clear()

'Populate Product Group drop-down for saving

da = New SqlClient.SqlDataAdapter("Select \* from tblProductGroup Order By ProductGroup", SqlConnection1)

da.Fill(ds, "ProductGroup")

Dim p As Integer = 0

While p <= ds.Tables("ProductGroup").Rows.Count - 1

Me.cbAddProductGroup.Items.Add(ds.Tables("ProductGroup").Rows(p).Item("ProductGroup"))

p = p + 1

End While

ds.Tables("ProductGroup").Clear()

ds.Tables("ProductGroup").Dispose()

End Sub

**DONE - IN frmEngWorkflowProductGroup: (THIS IS A NEW FORM)**

Private Sub frmEngWorkflowProductGroup\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

SqlConnection1.ConnectionString = sqlString

Search()

End Sub

Public Function Search()

Dim SqlString As String

DsProductGroup1.tblProductGroup.Clear()

Me.txtRecordID.DataBindings.Clear()

Dim cmdSave As String

cmdSave = Me.SqlDataAdapter1.SelectCommand.CommandText

Me.SqlDataAdapter1.SelectCommand.CommandText() += SqlString

Try

Me.SqlDataAdapter1.Fill(DsProductGroup1.tblProductGroup)

Catch ex As Exception

MsgBox(ex.ToString)

End Try

Me.SqlDataAdapter1.SelectCommand.CommandText = cmdSave

If DsProductGroup1.Tables("tblProductGroup").Rows.Count <> 0 Then

Search = True

'Create DataBindings

Me.txtRecordID.DataBindings.Add("Text", bsProductGroup, "RecordID")

End If

End Function

Private Sub btnSave\_Click(sender As Object, e As EventArgs) Handles btnSave.Click

Me.SqlDataAdapter1.Update(DsProductGroup1.tblProductGroup)

Search()

End Sub

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

u.ExecuteSQL(SqlConnection1, "Delete from tblProductGroup Where RecordID = '" & Val(Me.txtRecordID.Text) & "'")

Search()

End Sub

Private Sub GridView1\_BeforeLeaveRow(sender As Object, e As RowAllowEventArgs) Handles GridView1.BeforeLeaveRow

Me.SqlDataAdapter1.Update(DsProductGroup1.tblProductGroup)

Search()

End Sub