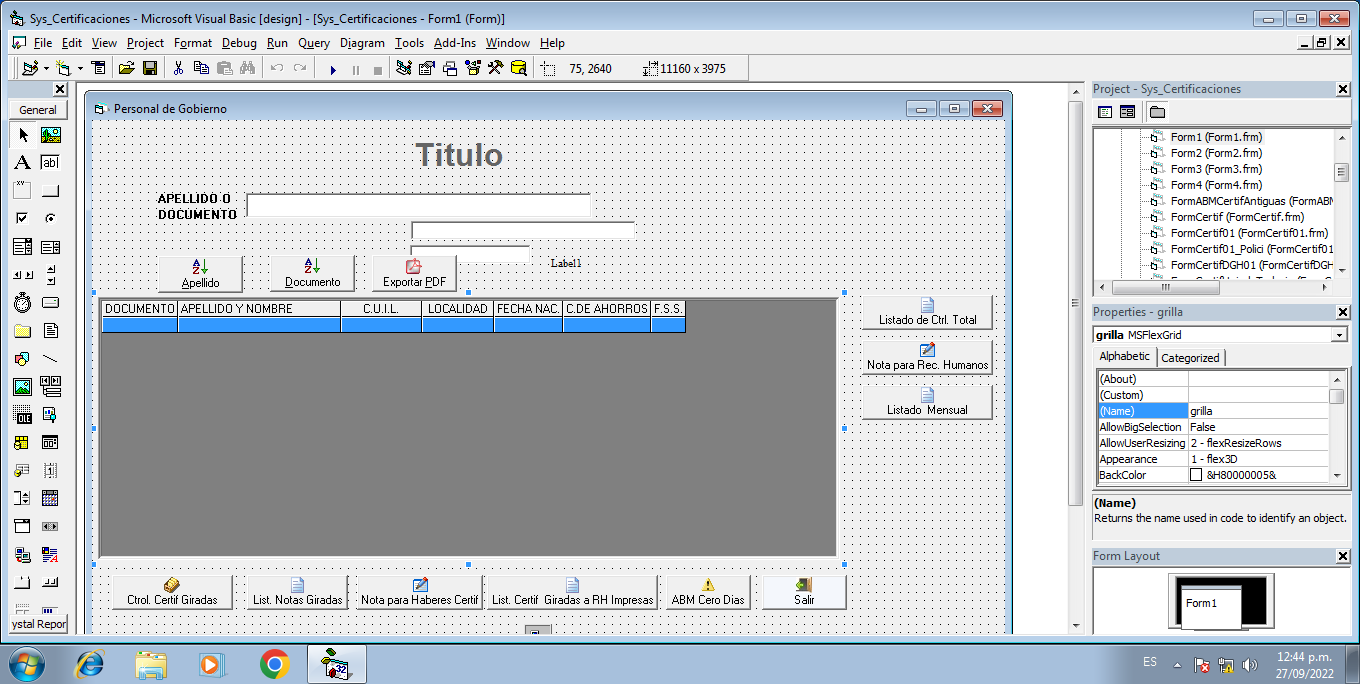
# FORMULARIOS:

Principal:

**Form1**

****

**Análisis de cada uno de los procedimientos del formulario**

**PROCEDIMIENTO**

Private Sub cmdabmcerodias\_Click()

Unload Me

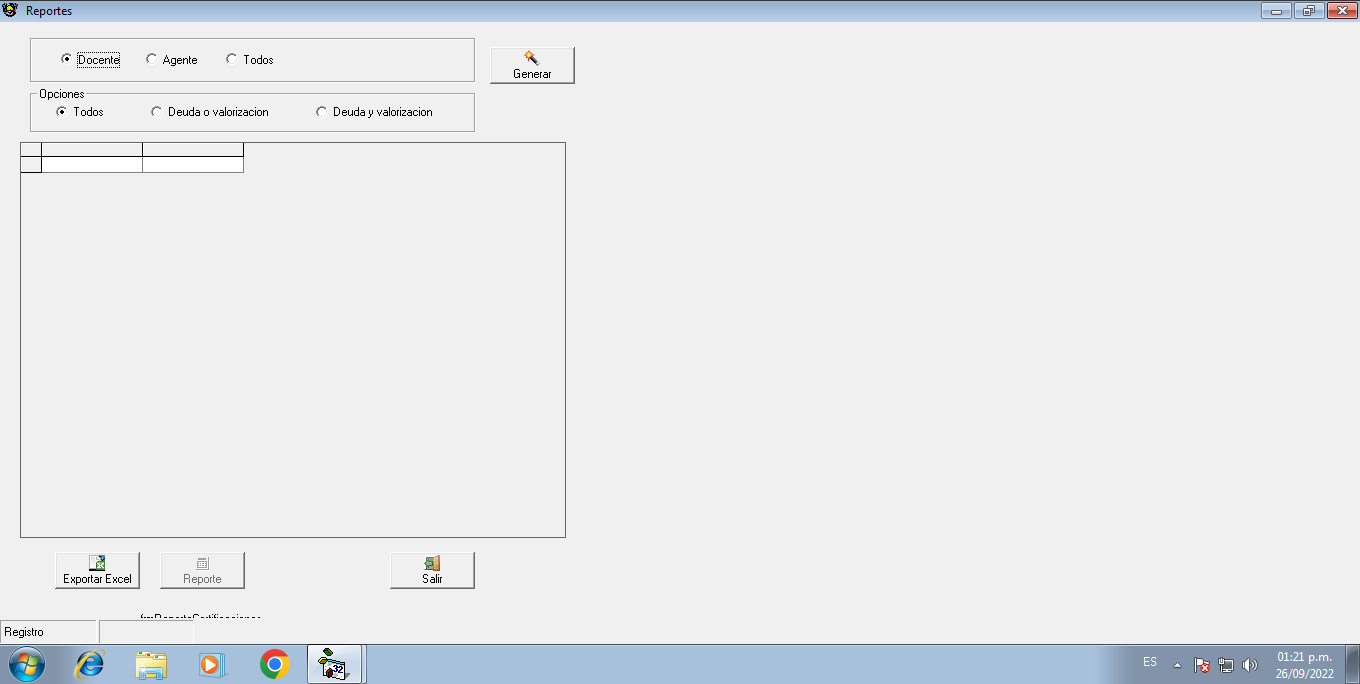
Form\_carga\_cerodias.Show

End Sub

**Muestra el formulario CERO DIAS**

**PROCEDIMIENTO**

**Control de certificaciones giradas – Inicia REPORTE CERFITIFACIONES**

****

Private Sub cmdcertifGiradasImpresas\_Click()

Dim a As String

If Valor\_Dep <> "HA" Then **COMPRUEBA SI EL DEPARTAMENTO (VALOR\_DEP) ES HABERES “HA” O RECURSOS HUMANOS “RH”**

MsgBox "NO TIENE ACCESO"

Exit Sub

End If

a = InputBox("Fecha de Pase a RH ", "Fecha de Pase a RH", (Date))

If a = "" Then **COMPRUEBA SI ESTA VACÍO “a”**

Exit Sub

Else

If IsDate(a) Then **COMPRUEBA SI “a” ES FECHA**

fecha\_giro = a **ASIGNA “a” A “fecha\_giro”**

Else

MsgBox "Ingreso una Fecha Invalida"

Exit Sub

End If

End If

DB.Execute ("VB\_Temp\_List\_Certif\_Giradas\_DGH '" & fecha\_giro & "' ")

**EJECUTA “**VB\_Temp\_List\_Certif\_Giradas\_DGH” **DE LA BASE DE DATOS:**

CREATE procedure [dbo].[VB\_Temp\_List\_Certif\_Giradas\_DGH]

@FechaPaseDGH datetime

as

IF OBJECT\_ID ('Temp\_List\_Certif\_Giradas') is not null

drop table Temp\_List\_Certif\_Giradas

select ce.\*,pe.Apellido, Empresa2=

CASE

WHEN CE.Empresa ='DOCENT' THEN 'DOCENTE'

WHEN CE.Empresa ='DOCPRI' THEN 'DOCENTE'

WHEN CE.Empresa ='DOCSEC' THEN 'DOCENTE'

WHEN CE.Empresa<>'DOCENT' THEN 'ADMINISTRACION'

END

INTO Temp\_List\_Certif\_Giradas

from certificaciones ce join Personas pe

on ce.Documento =pe.Documento

where Control='I' AND [Fecha Pase DGH] =@FechaPaseDGH

**CREA LA TABLA TEMPORAL Temp\_List\_Certif\_Giradas**

numero\_reporte = 171

Form\_Report.Show **EJECUTA y MUESTRA FORM\_REPORT**

End Sub

**PROCEDIMIENTO**

Private Sub cmdfonsocAudt\_Click()

DBLocal.Execute ("vb\_carga\_temp\_FondoSocialAuditor") **NO EXISTE EL PROCEDIMIENTO EN LA BASE DE DATOS**

numero\_reporte = 101

Form\_Report.Show

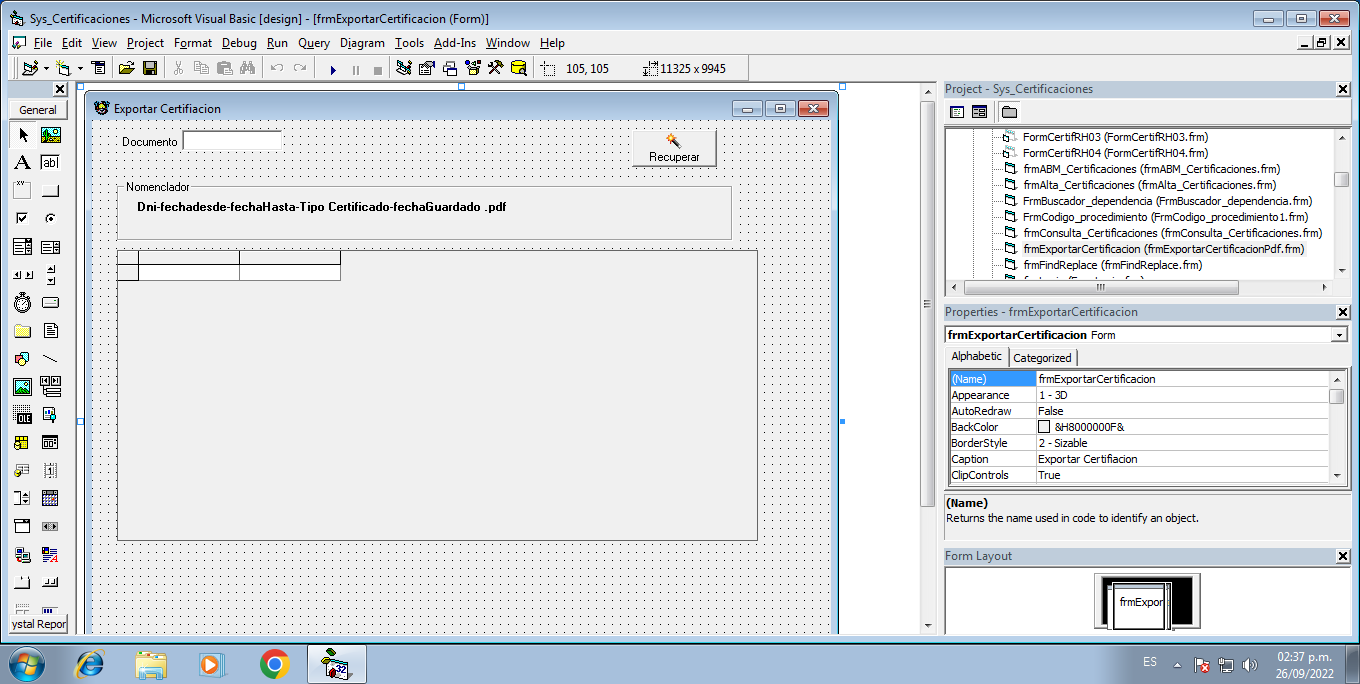
End Sub

**PROCEDIMIENTO**

Private Sub CmdGeneraPdf\_Click()

frmExportarCertificacion.Show **EJECUTA y MUESTRA** frmExportarCertificacion

End Sub

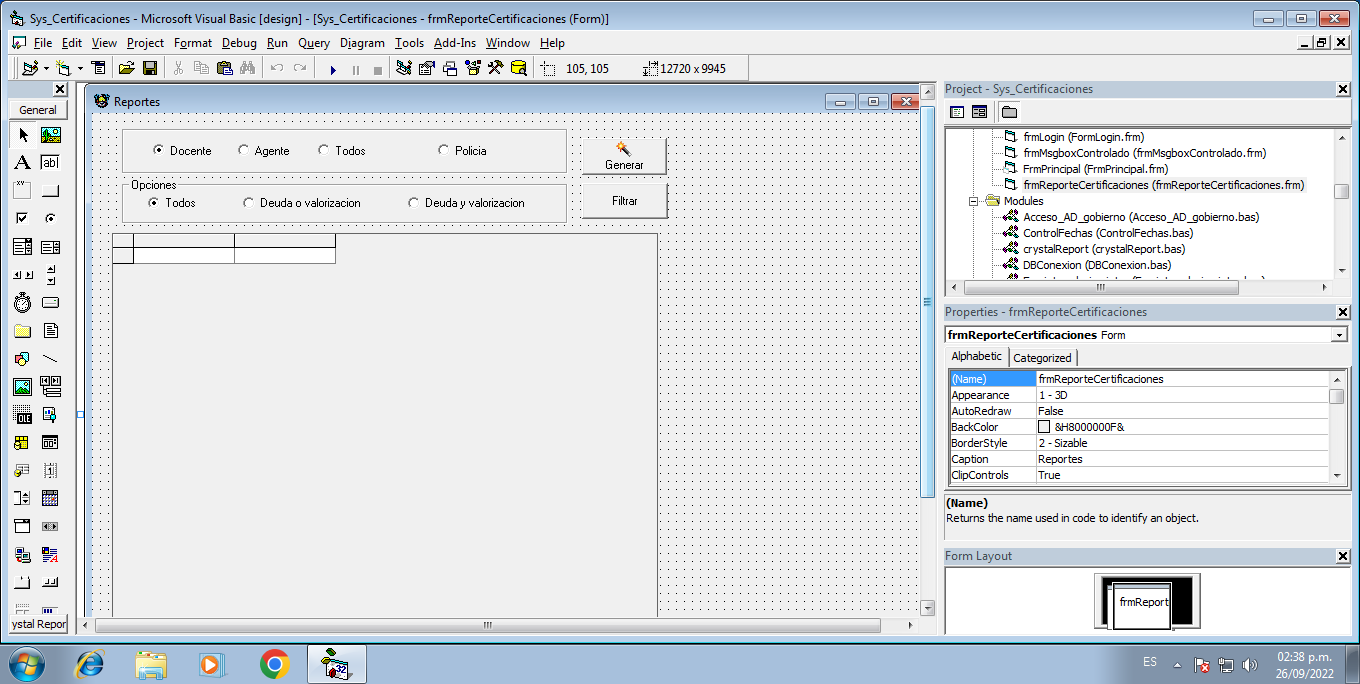


**PROCEDIMIENTO**

Private Sub cmdlistcert\_Click()

frmReporteCertificaciones.Show vbModal

End Sub



**PROCEDIMIENTO**

Private Sub cmdlistfondo\_Click()

DBLocal.Execute ("VB\_Crea\_y\_Carga\_FondoSocial")

**EJECUTA “**VB\_Crea\_y\_Carga\_FondoSocial” **DE LA BASE DE DATOS: NO EXISTE ESE PROCEDIMIENTO**

numero\_reporte = 100

Form\_Report.Show

End Sub

**PROCEDIMIENTO**

Private Sub cmdlistGirados\_Click()

Dim DepCam As String

Dim sSql As String

Dim a As String

a = InputBox("Fecha de Giro ", "Fecha de Giro", (Date))

If a = "" Then **COMPRUEBA SI ESTA VACÍO “a”**

Exit Sub

Else

If IsDate(a) Then **COMPRUEBA SI “a” ES FECHA**

fecha\_giro = a **ASIGNA “a” A “fecha\_giro”**

Else

MsgBox "Ingreso una Fecha Invalida"

Exit Sub

End If

End If

If Valor\_Dep = "HA" Or Valor\_Dep = "SL" Then

DepCam = "RH" 'cambiamos el valor de la dependencia como si fuera RH

Else

DepCam = "HA"

End If

**No se entiende. Si la dependencia es HA o SL. Asigna a DepCam: RH**

**Pero todo lo que no sea HA o SL también asigna a DepCam: HA.**

DBLocal.Execute ("VB\_carga\_TempListadGirados '" & fecha\_giro & "','" & DepCam & "'")

**NO EXISTE EL PROCEDIMIENTO**

sSql = "select \* from TempListadGirados"

**NO EXISTE LA TABLA**

'Set rs = New ADODB.Recordset

'rs.Open sSQL, DBLocal, adOpenForwardOnly, adLockReadOnly

' Set rsl = DBLocal.OpenResultset(sSQL, rdOpenKeyset, rdConcurReadOnly)

Set rsl = New ADODB.Recordset

rsl.Open sSql, DBLocal, adOpenKeyset, adLockReadOnly

If Not rsl.EOF Then

numero\_reporte = 117

Form\_Report.Show (1)

Else

MsgBox " No hay certificaciones giradas con esta fecha"

Exit Sub

End If

rsl.Close

End Sub

**PROCEDIMIENTO**

Private Sub Form\_Load()

Dim strSQL As String

Dim rs As ADODB.Recordset

' tarea = TareaLlamada

lblversion.Caption = "Version " & App.Major & "." & App.Minor & "." & App.Revision

' opcion = 6

' tarea = "2"

Me.Height = 9400

Me.Width = 13600

Me.Left = 0

Me.Top = 0

Lblformulario = Me.Name

' cmdartaumentos.Visible = False

**CARGA EL FORM**

If opcion = 1 Then

text\_titulo.Caption = "TERCERA COPIA"

lblFecCar.Visible = False

**DE ACUERDO A LA OPCIÓN CAMBIA EL TÍTULO DEL FORMULARIO**

End If

If opcion = 6 Then

Posic = 9

Seguridad **LLAMA AL MODULO DE SEGURIDAD**

strSQL = "SELECT sector FROM seguridad WHERE usuario='" & susuario & "'"

**BUSCA AL USUARIO PARA LEER SU SECTOR**

Set rs = New ADODB.Recordset

rs.Open strSQL, DB, adOpenForwardOnly, adLockReadOnly

If Not rs.EOF Then

sect = rs!sector

**ASIGNA EL SECTOR DEL USUARIO A LA VARIABLE sect**

End If

rs.Close

strSQL = "SELECT leyenda,[Fecha Maestro] FROM parametros"

**DE PARAMETROS SACA LA LEYENDA, POR EJEMPLO:**

**"2022 - 40° ANIVERSARIO DE LA GESTA HEROICA DE MALVINAS"**

**Y FECHA MAESTRO**

Set rs = New ADODB.Recordset

rs.Open strSQL, DB, adOpenForwardOnly, adLockReadOnly

If Not rs.EOF Then

leyenda = rs!leyenda **ASIGNA LA LEYENDA DEL RECORDSET A LA VARIABLE LEYENDA**

Fecha\_Maestro = rs![Fecha Maestro] **ASIGNA LA FECHA MAESTRO DEL RECORDSET A LA VARIABLE Fecha\_Maestro**

End If

rs.Close

'Consulta para traer cargo en caso de que tenga 14/09/2016

strSQL = "SELECT \* FROM [User Home] WHERE usuario='" & susuario & "'"

**DE User Home SACA nombre de usuario para asignarlo a nombre, cargo y ministerio.**

Set rs = New ADODB.Recordset

rs.Open strSQL, DB, adOpenForwardOnly, adLockReadOnly

If Not rs.EOF Then

If rs![nombre y apellido] <> Null Then

NombreCargo = rs![nombre y apellido]

DesccCargo = rs!cargo

MinistCargo = rs!ministerio

End If

End If

rs.Close

**DE ACUERDO A LA VARIABLE TareaLlamada QUE SE ASIGNA AL INICIALIZAR EL PROGRAMA SE MUESTRA EL TÍTULO EN EL FORMULARIO**

Select Case TareaLlamada

Case "1"

text\_titulo.Caption = "CERTIFICACIONES"

Case "2"

volver = ""

text\_titulo.Caption = "CERTIFICACIONES DE SERVICIOS"

Case "3"

text\_titulo.Caption = "CERTIFICACIONES ANTERIORES AL 2005"

End Select

**SE MUESTRA EL ELEMENTO LISTADO DE CERTIFICACIONES**

cmdlistcert.Visible = True

lblFecCar.Visible = False

**SE MUESTRA EL ELEMENTO LISTADO DE GIRADOS**

cmdlistGirados.Visible = True

**SI VALOR\_DEP (DEPENDENCIA) ES RH Y SI EL SECTOR ES 2 MUESTRA GIRO**

**SECTOR 2 A ESTE MOMENTO Y POR LA INTERPRETACIÓN, CORRESPONDE A “EMPRESAS” DOCENTES: DOCENT, DOCSEC, DOCPRI, DOCSAL**

If Valor\_Dep = "RH" Then

If (sect = 2) Then

Cmd\_Giro.Visible = False

Else

Cmd\_Giro.Visible = True

End If

End If

**SI VALOR\_DEP (DEPENDENCIA) ES “HA” (HABERES) MUESTRA EL BOTÓN DEL FORMULARIO DE CERO DIAS Y DE CERTIFICACIONES GIRADAS IMPRESAS**

If Valor\_Dep = "HA" Then

cmdabmcerodias.Visible = True

cmdcertifGiradasImpresas.Visible = True

End If

Posic = 9

Seguridad

End If

**COMPRUEBA EN SEGURIDAD SI PUEDE IMPRIMIR**

'Verifica si se puede Imprimir

' If Valor\_Seg < "2" Or Seg\_POLI = 1 Then

' Sel\_Listados.Enabled = False

' Sel\_Listados.Visible = False

' End If

**AHORA VERIFICA SI LA VARIABLE sdocum ES DIFERENTE DE CERO. RASTREANDO LA VARIABLE:**

Variables globales:

' usado para saber cual es la persona actual sobre la cual estoy trabajando

Public **sdocum** As Long

Public **sapellido** As String

' determina el legajo actual sobre el cual se esta trabajando

Public **slegajo** As Double

If sdocum <> 0 Then

If sapellido = "" Then

txtapedni = sdocum

Else

txtapedni = Trim(sapellido)

End If

orden = " ORDER BY DOCUMENTO "

ordenSQL = " documento "

Consultar

End If

**VERIFICA SI sapellido ESTA VACÍA, EN ESE CASO ASIGNA A txtepedni LA VARIABLE sdocum EN CASO CONTRARIO, QUE APELLIDO NO ESTÉ VACIO, ASIGNA A txtapedni LA VARIABLE sapellido**

**ORDENA EL LISTADO POR DOCUMENTO Y EJECUTA LA SENTENCIA DOCUMENTO, LLAMANDO AL PROCEDIMIENTO “CONSULTAR”**

**PROCEDIMIENTO**

Private Sub Consultar()

On Error GoTo Error\_Consultar **Habilita la rutina de control de errores**

Dim rs As ADODB.Recordset

Dim sSql As String

Dim CUIL As String

Screen.MousePointer = 11 **MousePointer de reloj de arena**

If IsNumeric(txtapedni.Text) Then **Verifica si la variable txtapedni es numérico. Devuelve un valor Boolean que indica si una expresión de puede evaluar como un número.**

DNI.Text = Val(txtapedni.Text) **La función Val se usa para devolver los números incluidos en una cadena.**

Txt(0).Text = ""  **A Txt le asigna vacio**

Else

DNI.Text = "" **En caso que no sea numérico a DNI le asigna vacío**

Txt(0).Text = Trim(txtapedni.Text) **A la variable Txt le asigna el apellido**

If Txt(0).Text = "" Then **Si el apellido está vacio, determina que no hay datos que ordenar**

MsgBox "No hay datos a Ordenar"

Screen.MousePointer = 0

Exit Sub

End If

End If

sSql = ArmarConsulta() **Arma la consulta con el DNI o el apellido**

Set rs = New ADODB.Recordset

rs.Open sSql, DB, adOpenForwardOnly, adLockReadOnly

**Arma la grilla**

grilla.Rows = 1

grilla.Cols = 7

grilla.Clear

grilla.FormatString = ">DOCUMENTO|<APELLIDO Y NOMBRE |^C.U.I.L. |^LOCALIDAD |^FECHA NAC.|^C.DE AHORROS|^F.S.S."

**Si en el recordset vuelven datos.**

If Not rs.EOF Then

rs.MoveFirst

While Not rs.EOF

CUIL = rs!cuil0 & "-" & rs!DOCUMENTO & "-" & rs!cuil1

If Val(rs!cuil0) = 0 Then

CUIL = "SIN CUIL"

End If

**Agrega los elementos en la grilla**

grilla.AddItem rs!DOCUMENTO & Chr(9) & \_

rs!APELLIDO & Chr(9) & \_

CUIL & Chr(9) & \_

rs!Localidad & Chr(9) & \_

rs![Fecha de Nacimiento] & Chr(9) & \_

rs![caja de ahorro] & Chr(9) & \_

rs![aporte solidario]

rs.MoveNext

Wend

Ajustar\_ColumnasMSHFlexGrid Form1, grilla

End If

Screen.MousePointer = 0

Exit Sub

Error\_Consultar:

Screen.MousePointer = 0

Set rs = Nothing

MsgBox Err.Description, vbCritical, "Error1"

End Sub

**PROCEDIMIENTO**

'ARMA CONSULTA

Private Function ArmarConsulta() As String

Dim sSql As String

sSql = "SELECT \* FROM PERSONAS "

sSql = sSql & "WHERE " **Arma la sentencia de consulta**

If Trim(Txt(0).Text) <> "" Then **Si la variable Txt no esta vacía**

If Not IsNumeric(Txt(0).Text) Then **Si la variable Txt no es numérica. Es decir trae el Apellido a consultar. La agrega a la sentencia SQL para ser consultada a la base de datos.**

sSql = sSql & " APELLIDO LIKE '" & Trim(Txt(0).Text) & "%' AND "

sapellido = Trim(Txt(0).Text)

Else 'codigo nuevo

sSql = sSql & " DOCUMENTO = '" & Txt(0).Text & "' AND " **Si la variable es numérica pasa el DOCUMENTO**

End If

End If

If Trim(DNI.Text) <> "" Then **Si Txt está vacio, va por DNI**

If IsNumeric(DNI.Text) Then **Si DNI es numérico, lo agrega a la sentencia de consulta**

sSql = sSql & " DOCUMENTO = '" & DNI.Text & "' AND "

Else 'codigo nuevo

sSql = sSql & " APELLIDO LIKE '" & DNI.Text & "%' AND " **Si no es numérico, lo toma apellido y asigna a la variable global sapellido Txt**

sapellido = Trim(Txt(0).Text)

End If

End If

sSql = Left(sSql, Len(sSql) - 6)

ArmarConsulta = sSql & orden

**Arma la consulta con la sentencia compuesta**

End Function

**PROCEDIMIENTO**

Private Sub DNI1()

Dim rs As ADODB.Recordset

Dim sSql As String

Txt(0).Text = ""

orden = " ORDER BY documento "

ordenSQL = " documento "

**Si la opción es 8, ordena por documento y envía una consulta**

If opcion = 8 Then 'Aporte solidario

orden = " ORDER BY documento "

ordenSQL = " documento "

Consultar

Exit Sub

End If

sSql = ArmarConsulta()

Set rs = New ADODB.Recordset

rs.Open sSql, DB, adOpenForwardOnly, adLockReadOnly

If rs.EOF Then **Si la consulta vuelve vacía, envía mensaje de error**

MsgBox "Documento inexistente"

rs.Close

Set rs = Nothing

Exit Sub

End If

If DNI.Text <> "" Then **Si DNI no ésta vacío**

If IsNumeric(DNI.Text) Then **Si DNI es numerico**

sdocum = DNI.Text **Asigna a la variable global sdocum, el contenido de DNI**

Else

sdocum = rs!DOCUMENTO **Si DNI no es numérico, es que se hizo la consulta con el APELLIDO, entonces guarda en la variable global sdocum, el DNI que viene en la consulta**

End If

If opcion = 1 Then **Si la opción es 1, abre el formulario 2.**

Form2.Show

Form1.Visible = False

End If

**Si la opción es 6 (CERTIFICACIONES), comprueba si puede acceder y le asigna DEPENDENCIA “HA” y muestra el formulario de certificaciones FormCertif**

If opcion = 6 Then

volver = ""

'If Tarea = "1" And seg\_proc >= 2 Then ' Valor\_Dep = "HA"

If TareaLlamada = "1" Then ' Valor\_Dep = "HA"

FormCertif.Show

CS = ""

Form1.Visible = False

Else

If TareaLlamada = "2" Then **Si la tarea es 2, se abre el formulario de Certificaciones de servicios FormCertifRH**

FormCertifRH.Show

Form1.Visible = False

Else

If tarea = "3" Then **Si la tarea es 3, se abre el formulario de Certificaciones de anteriores al 2005 FormABMCertifAntiguas**

FormABMCertifAntiguas.Show

End If

End If

End If

End If

If IsNumeric(txtapedni.Text) Then **Si la variable txtapedni es numérica genera una consulta ordenada por documento**

orden = " ORDER BY documento "

ordenSQL = " documento "

Consultar

End If

End If

End Sub

**PROCEDIMIENTO**

Private Sub grilla\_Click()

'Toma el documento y ejecuta el segundo formulario

If grilla.Text <> "" Then **Si el valor de la grilla no está vacío**

sdocum = grilla.Text **Asigna a la variable global sdocum el valor de la grilla**

If opcion = 1 Then 'Tercera Copia

Form2.Show **Si opción es igual a 1 abre el Form2**

Form1.Visible = False

End If

If opcion = 6 Then 'Certificaciones **Si la opción es igual a 6 (CERTIFICACIONES) verifica si la tarea es igual a 1( FormCertif) y la seguridad, además de asignar al valor de dependencia “HA”**

volver = ""

'If Tarea = "1" And seg\_proc >= 2 Then ' Valor\_Dep = "HA"

**En caso que TareaLlamada es igual a 1, Abre Formulario de Certificaciones, si es igual a 2, Certificación de Servicios, si es igual a 3, Certificaciones anteriores al 2005**

Select Case TareaLlamada

Case "1"

FormCertif.Show

CS = ""

Form1.Visible = False

Case "2"

FormCertifRH.Show

Form1.Visible = False

Case "3"

FormABMCertifAntiguas.Show

Form1.Visible = False

End Select

End If

Else

MsgBox "Pulse alguno de los botones para seleccionar datos"

End If

End Sub

**PROCEDIMIENTO**

Private Sub Orden\_Apellido\_Click(index As Integer)

orden = " ORDER BY apellido "

ordenSQL = " apellido "

Consultar

**Arma una consulta para ordenar por apellido**

End Sub

**PROCEDIMIENTO**

Private Sub Orden\_Documento\_Click()

orden = " ORDER BY documento "

ordenSQL = " documento "

Consultar

**Arma una consulta para ordenar por documento**

End Sub

**PROCEDIMIENTO**

Private Sub Cerrar\_Click()

Unload Me

**Cierra el formulario**

End Sub

**PROCEDIMIENTO**

Private Sub Txt\_KeyPress(index As Integer, KeyAscii As Integer)

If KeyAscii = 13 Then **Verifica si se aprieta la tecla ENTER**

orden = " ORDER BY apellido "

ordenSQL = " apellido "

volver = ""

Consultar

**Genera una consulta ordenada por apellido**

End If

End Sub

**PROCEDIMIENTO**

'Esta nota es para varias personas con al misma fecha de giro

Private Sub Cmd\_Giro\_Click()

Dim cadena As String

Dim ssSql As String

Dim cont As Integer

Dim u As String

Dim rs As ADODB.Recordset

Dim empresas As String

Dim sSql As String

Dim sdocum1 As String

Dim a As String

DBLocal.Execute ("IF OBJECT\_ID ('certificaciones') is not null ") & \_

"drop table certificaciones"

DBLocal.Execute ("IF OBJECT\_ID ('temporal') is not null ") & \_

"drop table temporal"

DB.Execute ("IF OBJECT\_ID ('temporal1') is not null ") & \_

"drop table temporal1"

**Ejecuta tres procedimientos donde verifica si existen esas tablas y si existen las elimina**

If sect = 3 Then

**Si el sector es 3 (DOCENTES), verifica en seguridad donde la dependencia sea RH**

ssSql = "select usuario from seguridad WHERE dependencia = 'RH' and sector in(2,3) order by sector "

Set rs = New ADODB.Recordset

rs.Open ssSql, DB, adOpenForwardOnly, adLockReadOnly

cadena = ""

cont = 0

If Not rs.EOF Then **Mientras haya elementos en el RecordSet**

rs.MoveFirst

While Not rs.EOF

cont = cont + 1

cmbusuarios.AddItem (cont & " - " & rs!usuario) **Va agregando Items**

cadena = cadena & cont & " : " & rs!usuario & vbNewLine

rs.MoveNext

Wend

End If

rs.Close

u = InputBox(cadena, "Seleccione el Nro del Usuario") **Muesta un inputbox para elegir el nro de usuario**

If IsNumeric(u) Then **Si “u” es numérico asigna a “usu” el valor del inputbox**

usu = u

Else

If u = "" Then ' es cancel **Si está vacío, cancela**

Exit Sub

Else

MsgBox "Seleccione un Numero de Usuario"

Exit Sub

End If

End If

If (usu < 1) Or (usu > cont) Then **Si el valor del inputbox es menor a 1 o mayor al contador usado para agregar elementos, determina que no existe el usuario**

MsgBox "No Existe el usuario"

Exit Sub

Else

Carga\_usuario **Si existe el usuario a buscar, llama al procedimiento Carga\_usuario**

End If

End If

a = InputBox("Fecha de Realizacion ", "Fecha de Certificacion", (Date)) **Abre un inputbox para tomar dos fechas**

If a = "" Then **Si el contenido del inputbox está vacío, cancela.**

Exit Sub

Else

If IsDate(a) Then **Si el contenido del inputbox es una fecha, asigna a fecha\_giro, el valor del inputbox, sino da un error.**

fecha\_giro = a

Else

MsgBox "Ingreso una Fecha Invalida"

Exit Sub

End If

End If

If sect = 3 Then 'docent **Si el sector es 3 (DOCENTES) genera una consulta para traer de la tabla personas los documentos que tengan esas empresas**

empresas = "('DOCENT','DOCSEC','DOCPRI')"

sSql = "SELECT ce.\*,pe.apellido Into certificaciones From [" & DB\_SERVER & "].personas.DBO.certificaciones ce join [" & DB\_SERVER & "].personas.DBO.personas pe " & \_

" on ce.documento=pe.documento WHERE ce.[fecha pedido]= '" & Format(fecha\_giro, "yyyy/mm/dd") & "' and ce.empresa IN " & empresas & " and ce.control =' ' and usuario='" & susuario & "' and [fecha pase rh] is null"

DBLocal.Execute (sSql)

End If

If sect = 5 Then ' solo administacion y por ususario **Si el sector es 5 (ADMINISTRACIÓN) genera una consulta para traer de la tabla personas los documentos que NO ESTEN en esas empresas** 'DOCENT','DOCSEC','DOCPRI'

empresas = "('DOCENT','DOCSEC','DOCPRI')"

sSql = "SELECT ce.\*,pe.apellido Into certificaciones From [" & DB\_SERVER & "].personas.DBO.certificaciones ce join [" & DB\_SERVER & "].personas.DBO.personas pe " & \_

" on ce.documento=pe.documento WHERE ce.[fecha pedido] = '" & Format(fecha\_giro, "yyyy/mm/dd") & "' and not ce.empresa IN " & empresas & " and ce.control =' ' and usuario='" & susuario & "' and [fecha pase rh] is null"

DBLocal.Execute (sSql)

End If

If sect = 4 Or sect = 0 Then 'administracion

**Si el sector es 4 o 0 genera una consulta para traer de la tabla personas los documentos que NO ESTEN en esas empresas** 'DOCENT','DOCSEC','DOCPRI','POLICI','PENIT','POLCIV'

DBLocal.Execute ("IF OBJECT\_ID ('temporal') is not null drop table temporal")

sSql = "SELECT ce.\*,pe.apellido into temporal From [" & DB\_SERVER & "].personas.DBO.certificaciones ce join [" & DB\_SERVER & "].personas.DBO.personas pe " & \_

" on ce.documento=pe.documento WHERE not ce.empresa IN ('DOCENT','DOCSEC','DOCPRI','POLICI','PENIT','POLCIV') and ce.control =' ' and [fecha pase rh] is null"

DBLocal.Execute (sSql)

sSql = "SELECT \* Into certificaciones from temporal WHERE [fecha pedido]= '" & fecha\_giro & "' "

DBLocal.Execute (sSql)

End If

If sect = "1" Then 'todas las empresas

**Si el sector es 1 genera una consulta para traer de la tabla personas los documentos que estén en todas las empresas**

sSql = "SELECT ce.\*,pe.apellido into temporal1 From [" & DB\_SERVER & "].personas.DBO.certificaciones ce join [" & DB\_SERVER & "].personas.DBO.personas pe " & \_

" on ce.documento=pe.documento WHERE ce.control =' ' and [fecha pedido]='" & fecha\_giro & "' and [fecha pase rh] is null "

DB.Execute (sSql)

DBLocal.Execute (" select \* into temporal from [" & DB\_SERVER & "].personas.dbo.temporal1")

sSql = "SELECT \* Into certificaciones from temporal WHERE [fecha pedido]= '" & fecha\_giro & "' "

DBLocal.Execute (sSql) **Ejecuta la sentencia**

End If

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

'DBLocal.Execute ("VB\_Carga\_Nota\_certificaciones '" & sdocum & "','" & fecha\_giro & "','" & empre & "','" & causaglobal & "','" & sUsuario & "' ")

**NO SE ENCUENTRA EL PROCEDIMIENTO EN LA BASE DE DATOS**

'Verificamos que hayan datos en la tabla

sSql = "SELECT \* from certificaciones "

'Set rs = New ADODB.Recordset

'rs.Open sSQL, DBLocal, adOpenForwardOnly, adLockReadOnly

**HACE UNA CONSULTA TRAYENDO TODAS LAS CERTIFICACIONES**

'Set rsl = DBLocal.OpenResultset(sSQL, rdOpenKeyset, rdConcurReadOnly)

Set rsl = New ADODB.Recordset

rsl.Open sSql, DBLocal, adOpenKeyset, adLockReadOnly

Dim leg1 As String

Dim pla1 As String

Dim sql As String

If rsl.EOF Then **SI EL RECORDSET ESTA VACÍO DA UN MENSAJE DE ERROR**

MsgBox "No hay planillas en condiciones para girar"

Exit Sub

Else

'Recorremos la tabla local y seteamos el campo fecha pase rh en la fecha en la tabla certificaciones en el servidos'

**RECORRE EL RECORDSET Y ACTUALIZA LA FECHA DE PASE RH**

rsl.MoveFirst

While Not rsl.EOF

sdocum1 = rsl!DOCUMENTO

leg1 = rsl!legajo

pla1 = rsl!Planilla

If sect = 3 Then **SI EL SECTOR ES 3 (DOCENTE)**

DB.Execute ("UPDATE Certificaciones SET [Fecha Pase rh] = '" & Date & "' " & \_

"WHERE Documento = '" & sdocum1 & "' AND Legajo = '" & leg1 & "' AND Planilla = '" & pla1 & "' ")

Else  **SI NO ES SECTOR DOCENTE**

DB.Execute ("UPDATE Certificaciones SET [Fecha Pase rh] = '" & Date & "' " & \_

"WHERE Documento = '" & sdocum1 & "' AND Legajo = '" & leg1 & "' AND Planilla = '" & pla1 & "' ")

End If

rsl.MoveNext

Wend

End If

If sect = 3 Then ' DOCENTES **SI ES SECTOR 3, TRAE DE CERTIFICACIONES LOS DOCENTES**

sql = "select count(distinct (documento))as cantidad FROM certificaciones WHERE [fecha pedido]= '" & fecha\_giro & "' and empresa IN ('DOCENT','DOCSEC','DOCPRI')"

End If

If sect = 5 Then ' DOCENTES **SI ES SECTOR 5, TRAE DE CERTIFICACIONES DE LOS QUE NO SON DOCENTES**

sql = "select count(distinct (documento))as cantidad FROM certificaciones WHERE [fecha pedido]= '" & fecha\_giro & "' and not empresa IN ('DOCENT','DOCSEC','DOCPRI')"

End If

If sect = 0 Or sect = 1 Then 'ADMIN **SI ES SECTOR 0 O 1, TRAE DE CERTIFICACIONES DE LA ADMINISTRACIÓN O TODOS EN GENERAL**

sql = "select count(distinct (documento))as cantidad FROM certificaciones WHERE [fecha pedido]= '" & fecha\_giro & "'"

End If

Set rs = New ADODB.Recordset

rs.Open sql, DB, adOpenForwardOnly, adLockReadOnly

If Not rs.EOF Then **Si no está vacío y verifica la cantidad y depende si es mayor a 1, genera un reporte mostrando el resultado**

If rs!cantidad > 1 Then

agregado = "de las personas que se detallan a continuación: "

Else

agregado = "de la persona que se detalla a continuación:"

End If

End If

numero\_reporte = 42

Form\_Report.Show (1)

End Sub

**PROCEDIMIENTO**

Public Sub ArmaFecha(fecha As Date) **Procedimiento que arma la fecha y verifica si es válida**

dia = Day(fecha)

Mes = Month(fecha)

Año = Year(fecha)

If dia = 0 Then

MsgBox (" No es una Fecha correcta"), vbCritical, "Error de Fecha"

Exit Sub

End If

End Sub

**PROCEDIMIENTO**

**Procedimiento para cargar la interface de usuarios**

Private Function Carga\_usuario()

If usu <> 0 Then

cmbusuarios.ListIndex = 0

Do While cmbusuarios.ListIndex < cmbusuarios.ListCount

If usu = Val(Left(cmbusuarios.Text, 1)) Then

usuarioc = Trim(Mid(cmbusuarios.Text, 5, 25))

Exit Do

End If

cmbusuarios.ListIndex = cmbusuarios.ListIndex + 1

Loop

End If

End Function

**PROCEDIMIENTO**

Private Sub txtapedni\_KeyPress(KeyAscii As Integer)

If IsNumeric(txtapedni) Then **Si la variable txtapedni es numerico**

If KeyAscii = 13 Then **Si se ingresó la tecla ENTER**

DNI.Text = Trim(txtapedni) **Se le asigna a DNI el valor numérico de txtapedni**

Txt(0).Text = "" **Se le asigna a Txt un valor vacío**

If DNI.Text = "" Then **Si DNI está vacío, muestra un mensaje para ingresar dato a consultar**

MsgBox " Ingrese Dato a Consultar "

Exit Sub

End If

orden = " ORDER BY documento "

ordenSQL = " documento "

Consultar **Genera una consulta con el documento**

DNI1

End If

Else ' es una cadena

If KeyAscii = 13 Then **Si se ingreso ENTER, y si DNI esta vacío y la variable es texto, le asigna a Txt el contenido de txtapedni y luego si está vacía, pide un dato a consultar**

DNI.Text = ""

Txt(0).Text = Trim(txtapedni)

If Txt(0).Text = "" Then

MsgBox " Ingrese Dato a Consultar "

Exit Sub

End If

orden = " ORDER BY apellido "

ordenSQL = " apellido "

volver = ""

Consultar **Genera consulta con el apellido**

End If

End If

End Sub

**PROCEDIMIENTO**

Private Sub buscadni()

Dim sSql As String

' txt(0).Text = ""

orden = " ORDER BY documento " **Si la opción es 8 (aporte solidario), genera una consulta por documento**

ordenSQL = " documento "

If opcion = 8 Then 'Aporte solidario

orden = " ORDER BY documento "

ordenSQL = " documento "

Consultar

Exit Sub

End If

Dim rs As ADODB.Recordset

sSql = ArmarConsulta()

Set rs = New ADODB.Recordset

rs.Open sSql, DB, adOpenForwardOnly, adLockReadOnly

If rs.EOF Then **Si la consulta vuelve vacía, el documento es inexistente**

MsgBox "Documento inexistente"

rs.Close

Set rs = Nothing

Exit Sub

End If

If txtapedni <> "" Then **Si la variable txtapedni no está vacía**

If IsNumeric(txtapedni) Then **Si es numérica, le asigna sdocum el contenido de txtapedni**

sdocum = txtapedni

Else

sdocum = rs!DOCUMENTO **Si no es numérica, toma el valor de la consulta a la base de datos y se lo asigna a sdocum**

End If

If opcion = 1 Then **Si es opción 1, (tercera copia) abre el FORM2**

Form2.Show

Form1.Visible = False

End If

If opcion = 6 Then **Si es opción 6 (certificaciones) abre el formulario de Certificaciones**

volver = ""

If TareaLlamada = "1" Then ' Valor\_Dep = "HA"

FormCertif.Show

CS = ""

Form1.Visible = False

Else

If TareaLlamada = "2" Then **Si es opción 2 (certificaciones) abre el formulario de Certificacion de servicios**

FormCertifRH.Show

Form1.Visible = False

End If

End If

End If

Else **Sino pide ingresar un DNI**

MsgBox "Debe ingresar el DNI"

End If

End Sub