Imports CapaNegocio

Imports Microsoft.VisualBasic

Imports System

Imports System.Windows.Forms

Imports System.Drawing

Imports System.Drawing.Printing

Public Class frm\_trans\_arti\_ventas

Dim objVentas As New cls\_trans\_ventas

Dim objBasicas As New cls\_basicas

Dim dbaction As String = "insert"

Dim action As String = "0"

Dim ls\_mensaje As String = ""

Dim objParameter As New Collection

Public li\_parameter As Integer = 0 ' Parametro enviado desde la ventana de mov\_lista\_compra para saber si es nuevo o modificar

Public Property parametros()

Get

Return objParameter

End Get

Set(ByVal value)

objParameter = value

End Set

End Property

Sub wf\_BloquearControles(ByVal as\_flag As Boolean)

txt\_codsuc.Enabled = as\_flag

btn\_sucursal.Enabled = as\_flag

txt\_codtipodoc.Enabled = as\_flag

Btn\_documento.Enabled = as\_flag

txt\_nser.Enabled = as\_flag

txt\_ndoc.Enabled = as\_flag

txt\_codsuc.Text = "01"

txt\_sucursal.Text = objBasicas.f\_obt\_desc("sucursal", "codsuc", "nombre", txt\_codsuc.Text)

txt\_codtipodoc.Text = "NP"

txt\_sucursal.Text = objBasicas.f\_obt\_desc("documento", "tipodoc", "nomb", txt\_codtipodoc.Text)

txt\_nser.Text = gs\_serie.ToString.PadLeft(4, "0")

txt\_ndoc.Text = objBasicas.f\_obt\_num\_doc(txt\_codtipodoc.Text, txt\_nser.Text, txt\_codsuc.Text)

txt\_ndoc.Text = txt\_ndoc.Text.Trim.PadLeft(7, "0")

End Sub

#Region "Impresión"

Private prtSettings As PrinterSettings

Private prtDoc As PrintDocument

Private prtFont As System.Drawing.Font

Private lineaActual As Integer = 0

Public Título As String = ""

Private Sub frm\_trans\_arti\_ventas\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

prtFont = New System.Drawing.Font("tahoma", 8)

' La configuración actual de la impresora predeterminada

prtSettings = New PrinterSettings

End Sub

Private Sub wf\_imprimir(ByVal esPreview As Boolean, ByVal as\_msg As String)

ls\_mensaje = as\_msg

' imprimir o mostrar el PrintPreview

'

If prtSettings Is Nothing Then

prtSettings = New PrinterSettings

End If

If prtDoc Is Nothing Then

prtDoc = New PrintDocument

AddHandler prtDoc.PrintPage, AddressOf prt\_PrintPage

End If

'

' resetear la línea actual

lineaActual = 0

'

' la configuración a usar en la impresión

prtDoc.PrinterSettings = prtSettings

Dim dtDatosImpresora As New DataTable

dtDatosImpresora = objVentas.f\_DatosImpresora("PR")

If dtDatosImpresora.Rows.Count > 0 Then

If dtDatosImpresora.Rows(0).Item("impresora") = "NO" Then

MsgBox("Impresora no Configurada")

Else

'prtDoc.PrinterSettings.PrinterName = "\\OPTERON\PENTRIX"

prtDoc.PrinterSettings.PrinterName = dtDatosImpresora.Rows(0).Item("impresora").ToString

End If

End If

Call TamanioPapel()

If esPreview Then

Dim prtPrev As New PrintPreviewDialog

prtPrev.Document = prtDoc

prtPrev.Text = "Previsualizar datos de " & Título

prtPrev.ShowDialog()

Else

prtDoc.Print()

End If

End Sub

Sub TamanioPapel()

Dim TamañoPersonal As Printing.PaperSize

Dim Ancho As Short

Dim Alto As Short

Try

Dim li\_ancho As Integer = f\_get\_data\_select("select valor from parametros where codigo='013'")

Ancho = Short.Parse(li\_ancho)

Dim li\_alto As Integer = f\_get\_data\_select("select valor from parametros where codigo='014'")

Alto = Short.Parse(li\_alto)

'TamañoPersonal = New Printing.PaperSize(txtNombre.Text, Ancho, Alto)

TamañoPersonal = New Printing.PaperSize("Personal", Ancho, Alto)

' Asignamos la impresora seleccionada

'prdoDocumento.PrinterSettings = ImpresoraActual

'prtDoc.PrinterSettings = ImpresoraActual

' Asignamos el tamaño personalizado de papel

'prdoDocumento.DefaultPageSettings.PaperSize = TamañoPersonal

prtDoc.DefaultPageSettings.PaperSize = TamañoPersonal

'MessageBox.Show("Nuevo tamaño asignado a documento")

Catch ex As Exception

MessageBox.Show(ex.Message, ex.Source, MessageBoxButtons.OK)

End Try

End Sub

Private Sub prt\_PrintPage(ByVal sender As Object, ByVal e As PrintPageEventArgs)

' Este evento se produce cada vez que se va a imprimir una página

Dim lineHeight As Single

Dim yPos As Single = e.MarginBounds.Top

Dim leftMargin As Single = e.MarginBounds.Left

Dim printFont As System.Drawing.Font

' Asignar el tipo de letra

printFont = prtFont

lineHeight = printFont.GetHeight(e.Graphics)

'Cabecera de Reporte

Dim fontTitulo As New Font("Tahoma", 10)

Dim fontCabecera As New Font("Tahoma", 22)

Dim fontEmpresa As New Font("Tahoma", 12)

Dim objVenta As New cls\_trans\_ventas

Dim dtt As DataTable

dtt = objVentas.f\_imprimir\_venta(txt\_codtipodoc.Text.Trim, txt\_nser.Text.Trim, txt\_ndoc.Text, txt\_codsuc.Text.Trim)

Dim ls\_documento As String = dtt.Rows(0).Item("documento")

Dim ls\_cliente As String = dtt.Rows(0).Item("nombre")

Dim ls\_fecha As String = dtt.Rows(0).Item("fecha")

Dim ls\_direccion As String = dtt.Rows(0).Item("direccion")

Dim ls\_condicion As String = dtt.Rows(0).Item("condicion")

Dim lrd\_cantidad As Decimal = 0

Dim ls\_articulo As String = ""

Dim lrd\_precio As Decimal = 0

Dim lrd\_total As Decimal = 0

Dim lrd\_sumTotal As Decimal = 0

Dim ls\_lineas As String = "----------------------------------------------------------------------------------------------------------------------------------------------"

Hora = DateTime.Now.ToString("h:mm:ss")

yPos = 0

e.Graphics.DrawString(dtt.Rows(0).Item("empresa"), fontCabecera, Brushes.Black, 250, yPos)

yPos = yPos + 35

e.Graphics.DrawString(dtt.Rows(0).Item("direccion\_emp"), fontTitulo, Brushes.Black, 170, yPos)

yPos = yPos + 20

e.Graphics.DrawString(dtt.Rows(0).Item("telefono\_emp"), fontTitulo, Brushes.Black, 250, yPos)

yPos = yPos + 15

e.Graphics.DrawString(ls\_lineas, fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 20

e.Graphics.DrawString("CLIENTE: ", fontTitulo, Brushes.Black, 20, yPos)

e.Graphics.DrawString(ls\_cliente, fontTitulo, Brushes.Black, 120, yPos)

e.Graphics.DrawString("PEDIDO: " & ls\_documento, fontTitulo, Brushes.Black, 500, yPos)

'e.Graphics.DrawString(ls\_documento, fontTitulo, Brushes.Black, 560, yPos)

yPos = yPos + 20

e.Graphics.DrawString("DIRECCIÓN: ", fontTitulo, Brushes.Black, 20, yPos)

e.Graphics.DrawString(ls\_direccion, fontTitulo, Brushes.Black, 120, yPos)

e.Graphics.DrawString("FECHA: " & ls\_fecha & " " & Hora, fontTitulo, Brushes.Black, 500, yPos)

'e.Graphics.DrawString(ls\_fecha, fontTitulo, Brushes.Black, 580, yPos)

yPos = yPos + 20

e.Graphics.DrawString("CONDICIÓN: ", fontTitulo, Brushes.Black, 20, yPos)

e.Graphics.DrawString(ls\_condicion, fontTitulo, Brushes.Black, 120, yPos)

yPos = yPos + 20

e.Graphics.DrawString(ls\_lineas, fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 10

e.Graphics.DrawString("CANTIDAD", fontTitulo, Brushes.Black, 20, yPos)

e.Graphics.DrawString("DESCRPCIÓN", fontTitulo, Brushes.Black, 110, yPos)

e.Graphics.DrawString("PRECIO", fontTitulo, Brushes.Black, 450, yPos)

e.Graphics.DrawString("TOTAL", fontTitulo, Brushes.Black, 600, yPos)

yPos = yPos + 10

e.Graphics.DrawString(ls\_lineas, fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 10

Dim sf As New StringFormat With {.Alignment = StringAlignment.Far}

For i As Integer = 0 To dtt.Rows.Count - 1

e.Graphics.DrawString(FormatNumber(dtt.Rows(i).Item("cantidad"), 0), fontTitulo, Brushes.Black, 55, yPos + i \* 18, sf)

e.Graphics.DrawString(dtt.Rows(i).Item("articulo"), fontTitulo, Brushes.Black, 110, yPos + i \* 18)

e.Graphics.DrawString(FormatNumber(dtt.Rows(i).Item("precio"), 2), fontTitulo, Brushes.Black, 500, yPos + i \* 18, sf)

e.Graphics.DrawString(FormatNumber(dtt.Rows(i).Item("total"), 2), fontTitulo, Brushes.Black, 650, yPos + i \* 18, sf)

lrd\_sumTotal = lrd\_sumTotal + dtt.Rows(i).Item("total")

If i = dtt.Rows.Count - 1 Then

' 16 Valor de incremento

yPos = (yPos + (i) \* 18) + 16

End If

Next

e.Graphics.DrawString(ls\_lineas, fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 20

e.Graphics.DrawString(FormatNumber(lrd\_sumTotal, 2), fontTitulo, Brushes.Black, 650, yPos, sf)

yPos = yPos + 50

e.Graphics.DrawString(ls\_lineas, fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 10

e.Graphics.DrawString(" V° B° ALMACÉN RECIBÍ CONFORME", fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 25

e.Graphics.DrawString("CTA BANCARIA:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_", fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 25

e.Graphics.DrawString("A CUENTA SOLES:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ A CUENTA BANCO:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ SALDO:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_", fontTitulo, Brushes.Black, 0, yPos)

yPos = yPos + 25

e.Graphics.DrawString(ls\_mensaje, fontTitulo, Brushes.Black, 20, yPos)

End Sub

#End Region

Private Sub frm\_trans\_arti\_ventas\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles Me.KeyDown

Select Case e.KeyCode

Case Keys.Insert

e.Handled = True

Call wf\_agregar\_articulos()

Case Keys.Delete

e.Handled = True

Call wf\_eliminar\_articulo()

Case Keys.Escape

e.Handled = True

If Me.action = "0" Then

Me.Close()

Else

Call wf\_agregar()

End If

End Select

End Sub

#Region "Validar Solo Numeros en Grid"

Private WithEvents cellTextBox As DataGridViewTextBoxEditingControl

Private Sub cellTextBox\_KeyDown( \_

ByVal sender As Object, \_

ByVal e As System.Windows.Forms.KeyEventArgs) Handles cellTextBox.KeyDown

End Sub

'Private Sub cellTextBox\_KeyPress( \_

' ByVal sender As Object, \_

' ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles cellTextBox.KeyPress

'End Sub

Private Sub cellTextBox\_KeyPress( \_

ByVal sender As Object, \_

ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles cellTextBox.KeyPress

' Referenciamos el control TextBox subyacente.

'

Dim tb As TextBox = TryCast(sender, TextBox)

' Si la conversión ha fallado, abandonamos el procedimiento.

'

If (tb Is Nothing) Then

e.Handled = True

Return

End If

Dim isDecimal, isSign, isValidChar As Boolean

Dim decimalSeparator As String = Nothing

Select Case e.KeyChar

Case "."c, ","c

' Obtenemos el carácter separador decimal existente

' actualmente en la configuración regional de Windows.

'

decimalSeparator = \_

Threading.Thread.CurrentThread. \_

CurrentCulture.NumberFormat.NumberDecimalSeparator

' Hacemos que el carácter tecleado coincida con el

' carácter separador existentente en la configuración

' regional.

'

e.KeyChar = decimalSeparator.Chars(0)

' Si el primer carácter que se teclea es el separador decimal,

' o si bien, existe un signo en el primer carácter, envío la

' combinación '0,'.

'

If (((tb.TextLength = 0) OrElse (tb.SelectionLength = tb.TextLength)) OrElse \_

((tb.TextLength = 1) AndAlso ((tb.Text.Contains("-")) OrElse \_

(Text.Contains("+"))))) Then

' NOTA: Envío la combinación "0," mediante el método Send,

' para que en el código cliente se desencadenen los

' eventos de teclado.

'

SendKeys.Send("{0}")

SendKeys.Send("{" & decimalSeparator & "}")

e.Handled = True

Return

End If

' Es un carácter válido.

'

isDecimal = True

isValidChar = True

Case "-"c, "+"c ' Signos negativo y positivo

' Es un carácter válido.

'

isMinus = True

isValidChar = True

Case Else

' Sólo se admitirán números y la tecla de retroceso.

'

Dim isDigit As Boolean = Char.IsDigit(e.KeyChar)

Dim isControl As Boolean = Char.IsControl(e.KeyChar)

If ((isDigit) OrElse (isControl)) Then

isValidChar = True

Else

e.Handled = True

Return

End If

End Select

' Si es un carácter válido, y el texto del control

' se encuentra totalmente seleccionado, elimino

' el valor actual del control.

'

If ((isValidChar) And (tb.SelectionLength = tb.TextLength)) Then

tb.Text = String.Empty

End If

If (isSign) Then

' Admitimos los caracteres positivo y negativo, siempre y cuando

' sea el primer carácter del texto, y no exista ya ningún otro

' signo escrito en el control.

'

If ((tb.SelectionStart <> 0) OrElse \_

(tb.Text.IndexOf("-") >= 0) OrElse \_

(tb.Text.IndexOf("+") >= 0)) Then

e.Handled = True

Return

End If

End If

If (isDecimal) Then

' Si en el control hay ya escrito un separador decimal,

' deshechamos insertar otro separador más.

'

If (tb.Text.IndexOf(decimalSeparator) >= 0) Then

e.Handled = True

Return

End If

End If

End Sub

Private Sub cellTextBox\_KeyUp( \_

ByVal sender As Object, \_

ByVal e As System.Windows.Forms.KeyEventArgs) Handles cellTextBox.KeyUp

End Sub

Private Sub dgv\_ventas\_EditingControlShowing( \_

ByVal sender As Object, \_

ByVal e As DataGridViewEditingControlShowingEventArgs) \_

Handles dgv\_ventas.EditingControlShowing

' Este evento se producirá cuando la celda pasa a modo de edición.

' Referenciamos el control DataGridViewTextBoxEditingControl actual.

'

cellTextBox = TryCast(e.Control, DataGridViewTextBoxEditingControl)

' Obtenemos el estilo de la celda actual

'

Dim style As DataGridViewCellStyle = e.CellStyle

' Mientras se edita la celda, aumentaremos la fuente

' y rellenaremos el color de fondo de la celda actual.

'

With style

.Font = New Font(style.Font.FontFamily, 11, FontStyle.Bold)

.BackColor = Color.Beige

End With

End Sub

#End Region

Private Sub frm\_trans\_arti\_ventas\_Shown(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Shown

If li\_parameter = 1 Then

dbpanel.dbaction = "insert"

Call wf\_nuevo()

If gs\_user\_grupo = "01" Or gs\_user\_grupo = "99" Then

txt\_codsuc.Text = gs\_codsuc

txt\_sucursal.Text = objBasicas.f\_obt\_desc("sucursal", "codsuc", "nombre", txt\_codsuc.Text)

txt\_codsuc.Enabled = True

btn\_sucursal.Enabled = True

txt\_codsuc.Focus()

Else

txt\_codsuc.Text = gs\_codsuc

txt\_sucursal.Text = objBasicas.f\_obt\_desc("sucursal", "codsuc", "nombre", txt\_codsuc.Text)

txt\_codsuc.Enabled = False

btn\_sucursal.Enabled = False

txt\_codvend.Text = gs\_Codvend

txt\_vendedor.Text = objBasicas.f\_obt\_desc("vendedor", "codvend", "nombre", txt\_codvend.Text)

txt\_codvend.Enabled = False

Btn\_vendedor.Enabled = False

Call wf\_BloquearControles(False)

mtb\_fecha.Focus()

End If

Else

dbpanel.dbaction = "update"

Call wf\_modificar(objParameter.Item(1), objParameter.Item(2), objParameter.Item(3), objParameter.Item(4))

End If

AddHandler dgv\_ventas.CellValueChanged, AddressOf f\_valida\_ingreso\_datos\_grilla

AddHandler dgv\_ventas.CellValueChanged, AddressOf f\_Costo\_minimo

AddHandler dgv\_ventas.CellValueChanged, AddressOf f\_obt\_stock\_minimo

AddHandler dgv\_ventas.CellValueChanged, AddressOf wf\_calcular\_importe

End Sub

Private Sub dgv\_ventas\_AlternatingRowsDefaultCellStyleChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles dgv\_ventas.AlternatingRowsDefaultCellStyleChanged

Me.action = "1"

End Sub

Sub wf\_nuevo()

wf\_limpiar\_controles()

chk\_vta\_adelantada.Checked = False

txt\_codsuc.Focus()

End Sub

Sub wf\_limpiar\_controles()

For Each Control In Me.Panel3.Controls

If (TypeOf Control Is TextBox) Then Control.text = ""

Next

txt\_codmoneda.Text = "S"

txt\_moneda.Text = "SOLES"

txt\_idmodalidad.Text = "N"

dgv\_ventas.Rows.Clear()

mtb\_fecha.Text = f\_fecha\_servidor()

txt\_subtotal.Text = FormatNumber(0.0, 2)

txt\_igv.Text = FormatNumber(0.0, 2)

txt\_total.Text = FormatNumber(0.0, 2)

txt\_dias.Text = FormatNumber(0.0, 2)

txt\_dolar.Text = FormatNumber(0.0, 2)

txt\_impuesto.Text = objBasicas.f\_impuesto()

txt\_dolar.Text = objBasicas.f\_obt\_dolar()

txt\_usuario.Text = gs\_usureg

End Sub

Sub wf\_obt\_desc()

' Obtener los campos del indice indicado y llenar los textos

Dim objBasicas As New cls\_basicas

txt\_documento.Text = objBasicas.f\_obt\_desc("documento", "tipodoc", "nomb", txt\_codtipodoc.Text)

txt\_cliente.Text = objBasicas.f\_obt\_desc("cliente", "codclie", "nombre", txt\_codclie.Text)

txt\_sucursal.Text = objBasicas.f\_obt\_desc("sucursal", "codsuc", "nombre", txt\_codsuc.Text)

txt\_moneda.Text = objBasicas.f\_obt\_desc("moneda", "codmoneda", "nombre", txt\_codmoneda.Text)

txt\_condicion.Text = objBasicas.f\_obt\_desc("condicion", "opci", "nombre", txt\_codcondicion.Text)

txt\_cuenta.Text = objBasicas.f\_obt\_desc("cuenta", "codcta", "n\_cta", txt\_codcta.Text)

txt\_vendedor.Text = objBasicas.f\_obt\_desc("vendedor", "codvend", "nombre", txt\_codvend.Text)

End Sub

Private Sub txt\_codtipodoc\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codtipodoc.KeyDown

If e.KeyCode = Keys.Enter Then

wf\_sel\_tipodoc()

End If

End Sub

Sub wf\_sel\_tipodoc()

Dim objTipodoc As New cls\_mant\_docu

li\_dato = objTipodoc.f\_ver\_existe(txt\_codtipodoc.Text.Trim, "2")

If li\_dato > 0 Then

txt\_documento.Text = objTipodoc.f\_obt\_nombre(txt\_codtipodoc.Text)

Else

Call wf\_obt\_tipodoc()

End If

End Sub

Sub wf\_obt\_tipodoc()

txt\_codtipodoc.Text = Search.f\_buscar\_tipodocVenta("0", " tipo like '%V%' ")

Call wf\_obt\_desc()

End Sub

Private Sub txt\_nser\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_nser.KeyDown

If e.KeyCode = Keys.Enter Then

If txt\_nser.Text = "" Then

If f\_checkedvalue(txt\_nser) = False Then

Return

End If

Else

txt\_nser.Text = txt\_nser.Text.Trim.PadLeft(4, "0")

txt\_ndoc.Text = objBasicas.f\_obt\_num\_doc(txt\_codtipodoc.Text, txt\_nser.Text, txt\_codsuc.Text)

txt\_ndoc.Text = txt\_ndoc.Text.Trim.PadLeft(7, "0")

End If

End If

End Sub

Private Sub txt\_ndoc\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_ndoc.KeyDown

If e.KeyCode = Keys.Enter Then

If txt\_ndoc.Text = "" Then

If f\_checkedvalue(txt\_ndoc) = False Then

Return

End If

Else

txt\_ndoc.Text = txt\_ndoc.Text.Trim.PadLeft(7, "0")

ls\_sql = "select count(\*) from venta where codsuc='" & txt\_codsuc.Text & "' and tipodoc='" & txt\_codtipodoc.Text.Trim & "' and nser='" & txt\_nser.Text.Trim & "' and ndoc='" & txt\_ndoc.Text.Trim & "' and codconc in ('VTAS','VTAA')"

If f\_get\_data\_select(ls\_sql) > 0 Then

Call wf\_modificar(txt\_codsuc.Text, txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text)

End If

End If

End If

End Sub

Private Sub txt\_codclie\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codclie.KeyDown

If e.KeyCode = Keys.Enter Then

Call wf\_sel\_cliente()

End If

End Sub

Sub wf\_sel\_cliente()

Dim objCliente As New cls\_mant\_clientes

li\_dato = objCliente.f\_ver\_existe(txt\_codclie.Text.Trim, "2")

If li\_dato > 0 Then

txt\_cliente.Text = objCliente.obt\_nombre(txt\_codclie.Text.Trim)

Else

Call wf\_obt\_cliente()

End If

End Sub

Sub wf\_obt\_cliente()

txt\_codclie.Text = f\_buscar\_cliente("1", "1")

wf\_obt\_desc()

End Sub

Private Sub mtb\_fecha\_GotFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles mtb\_fecha.GotFocus

mtb\_fecha.BackColor = Color.AntiqueWhite

Me.mtb\_fecha.SelectAll()

End Sub

Private Sub mtb\_fecha\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles mtb\_fecha.KeyDown

If e.KeyData = Keys.Enter Then

txt\_codclie.Focus()

End If

End Sub

Private Sub mtb\_fecha\_LostFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles mtb\_fecha.LostFocus

mtb\_fecha.BackColor = Color.White

End Sub

Private Sub txt\_codmoneda\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codmoneda.KeyDown

If e.KeyCode = Keys.Enter Then

wf\_sel\_moneda()

End If

End Sub

Sub wf\_sel\_moneda()

Dim ObjMonedas As New cls\_mant\_monedas

li\_dato = ObjMonedas.f\_ver\_existe(Trim(txt\_codmoneda.Text), "2")

If li\_dato > 0 Then

txt\_moneda.Text = ObjMonedas.f\_obt\_nombre(txt\_codmoneda.Text)

Else

Call wf\_obt\_moneda()

End If

End Sub

Sub wf\_obt\_moneda()

txt\_codmoneda.Text = Search.f\_buscar\_moneda("0", "1")

wf\_obt\_desc()

End Sub

Private Sub txt\_codsuc\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codsuc.KeyDown

If e.KeyData = Keys.Enter Then

Call wf\_sel\_sucursal()

End If

End Sub

Sub wf\_sel\_sucursal()

Dim objSucursal As New cls\_mant\_sucursal

li\_dato = objSucursal.f\_ver\_existe(Trim(txt\_codsuc.Text.Trim), "2")

If li\_dato > 0 Then

txt\_sucursal.Text = objSucursal.f\_obt\_nombre(txt\_codsuc.Text.Trim)

Else

Call wf\_obt\_sucursal()

End If

End Sub

Sub wf\_obt\_sucursal()

txt\_codsuc.Text = Search.f\_buscar\_sucursal("0", "1")

Call wf\_obt\_desc()

End Sub

Private Sub txt\_codcondicion\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codcondicion.KeyDown

If e.KeyCode = Keys.Enter Then

Call wf\_sel\_condicion()

If txt\_codcondicion.Text.Trim = "2" Then 'MsgBox("contado")

txt\_codcta.Enabled = True

Btn\_cuenta.Enabled = True

Else 'MsgBox("credito")

txt\_codcta.Enabled = False

txt\_codcta.Text = ""

Btn\_cuenta.Enabled = False

End If

End If

End Sub

Sub wf\_sel\_condicion()

Dim ObjCondicion As New cls\_mant\_condicion

li\_dato = ObjCondicion.f\_ver\_existe(Trim(txt\_codcondicion.Text), "2")

If li\_dato > 0 Then

txt\_condicion.Text = ObjCondicion.f\_obt\_nombre(txt\_codcondicion.Text)

Else

Call wf\_obt\_condicion()

End If

End Sub

Sub wf\_obt\_condicion()

txt\_codcondicion.Text = Search.f\_buscar\_condicionVenta("0", "filtro like '%2%'")

Call wf\_obt\_desc()

If txt\_codcondicion.Text.Trim = "2" Then 'MsgBox("contado")

txt\_codcta.Enabled = True

Btn\_cuenta.Enabled = True

Else 'MsgBox("credito")

txt\_codcta.Enabled = False

txt\_codcta.Text = ""

Btn\_cuenta.Enabled = False

End If

End Sub

Private Sub txt\_codcta\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codcta.KeyDown

If e.KeyCode = Keys.Enter Then

Call wf\_sel\_cuenta()

End If

End Sub

Sub wf\_sel\_cuenta()

Dim ObjCuentas As New cls\_mant\_cuentas

li\_dato = ObjCuentas.f\_ver\_existe(Trim(txt\_codcta.Text), "2")

If li\_dato > 0 Then

txt\_cuenta.Text = ObjCuentas.obt\_nombre(txt\_codcta.Text)

Else

Call wf\_obt\_cuentas()

End If

If txt\_codcta.Text.Trim = "2" Then

Btn\_cuenta.Enabled = True

Else

Btn\_cuenta.Enabled = False

End If

End Sub

Sub wf\_obt\_cuentas()

'txt\_codcta.Text = Search.f\_buscar\_cuenta("0", "1")

txt\_codcta.Text = Search.f\_buscar\_cuenta("1", "codsuc='" & txt\_codsuc.Text & "'")

wf\_obt\_desc()

If txt\_codcta.Text.Trim = "2" Then

Btn\_cuenta.Enabled = True

Else

Btn\_cuenta.Enabled = False

End If

End Sub

Private Sub txt\_codvend\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_codvend.KeyDown

If e.KeyCode = Keys.Enter Then

wf\_sel\_vendedor()

End If

End Sub

Sub wf\_sel\_vendedor()

Dim ObjVendedor As New cls\_mant\_vendedores

li\_dato = ObjVendedor.f\_ver\_existe(Trim(txt\_codvend.Text), "2")

If li\_dato > 0 Then

txt\_vendedor.Text = ObjVendedor.obt\_nombre(txt\_codvend.Text)

Else

Call wf\_obt\_vendedor()

End If

End Sub

Sub wf\_obt\_vendedor()

Dim ls\_sql As String = "select dbo.usp\_trim(valor) from parametros where codigo='001'"

If f\_get\_data\_select(ls\_sql.Trim) = "0" Then

txt\_codvend.Text = Search.f\_buscar\_vendedor("0", "1")

Else

txt\_codvend.Text = Search.f\_buscar\_vendedor("0", "codsuc like '" & txt\_codsuc.Text & "'")

End If

wf\_obt\_desc()

End Sub

Private Sub txt\_detalle\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_detalle.KeyDown

If e.KeyCode = Keys.Enter Then

If dbaction = "insert" Then

Me.action = "1"

Call wf\_agregar\_articulos()

End If

Me.action = "1"

If dgv\_ventas.Rows.Count > 0 Then

Dim idx As Integer = dgv\_ventas.Rows.Count - 1

dgv\_ventas.Rows(idx).Cells(3).Selected = True

Call wf\_format\_detalle(idx)

dgv\_ventas.Focus()

End If

End If

End Sub

Sub wf\_agregar\_articulos()

If objVentas.f\_obt\_stock\_almacen(txt\_codsuc.Text, "1") > 0 Then ' verificar que almacen tenga stock

ls\_codigo = Search.f\_buscar\_articulo("0", "2", txt\_codsuc.Text)

'Call wf\_validar\_duplicado\_detalle(ls\_codigo)

'' Verificar que no se encuentre el articulo en la lista

If wf\_validar\_duplicado\_detalle(ls\_codigo) = False Then

Return

End If

If ls\_codigo <> "" Then

Me.action = "1"

Dim objBasicas As New cls\_basicas

ls\_nombre = objBasicas.f\_obt\_desc("articulo", "codarti", "nombre", ls\_codigo)

ls\_unidadm = objBasicas.f\_obt\_desc("articulo", "codarti", "unidadm", ls\_codigo)

lrd\_costo = objBasicas.f\_obt\_desc("articulo", "codarti", "precio", ls\_codigo)

lrd\_peso = objBasicas.f\_obt\_desc("articulo", "codarti", "peso", ls\_codigo)

dgv\_ventas.Rows.Add(ls\_codigo, ls\_nombre, ls\_unidadm, 0, lrd\_costo, 0, 0, lrd\_peso)

Call wf\_calcular\_importe()

Dim idx As Integer

idx = dgv\_ventas.Rows.Count - 1

dgv\_ventas.Rows(idx).Cells(3).Selected = True

Call wf\_format\_detalle(idx)

Call wf\_WidthColumn()

End If

Else

MessageBox.Show("Sucursal " & Trim(txt\_sucursal.Text) & " no cuenta con Stock", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Stop, MessageBoxDefaultButton.Button1)

End If

End Sub

Sub wf\_WidthColumn()

If dgv\_ventas.RowCount > 10 Then

dgv\_ventas.Columns(1).Width = 460

Else

dgv\_ventas.Columns(1).Width = 480

End If

End Sub

Function wf\_validar\_duplicado\_detalle(ByVal as\_codigo As String)

ls\_nombre = f\_get\_data\_select("select nombre from articulo where codarti='" & as\_codigo & "'")

For i As Integer = 0 To dgv\_ventas.RowCount - 1

If dgv\_ventas.Rows(i).Cells(0).Value = as\_codigo Then

MessageBox.Show(ls\_nombre + Chr(13) + Chr(13) + "Articulo ya existe en la Lista ?", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

Return False

'Return True

End If

Next

Return True

End Function

Sub f\_obt\_stock\_minimo()

'Dim ls\_codsuc As String = txt\_codsuc.Text.Trim

'Dim ls\_tipodoc As String = txt\_codtipodoc.Text.Trim

'Dim ls\_nser As String = txt\_nser.Text.Trim

'Dim ls\_ndoc As String = txt\_ndoc.Text.Trim

'Dim idx As Byte = dgv\_ventas.CurrentRow.Index()

'Dim ls\_codigo As String = dgv\_ventas.Rows(idx).Cells(0).Value

'ls\_cant\_ant = f\_get\_data\_select("select cant from venta\_detalle where codsuc='" & ls\_codsuc & "' and tipodoc='" & ls\_tipodoc & "' and nser='" & ls\_nser & "' and ndoc='" & ls\_ndoc & "' and codarti='" & ls\_codigo & "'")

'ls\_stock = f\_get\_data\_select("select stock from stock\_sucursal where codarti='" & ls\_codigo & "' and codsuc='" & txt\_codsuc.Text.Trim & "'")

'ls\_stock = ls\_cant\_ant + ls\_stock

'If String.IsNullOrEmpty(dgv\_ventas.CurrentRow.Cells("cant").FormattedValue.ToString) Then

' dgv\_ventas.Rows(idx).Cells("cant").Value = 0

'Else

' If dgv\_ventas.Rows(idx).Cells("cant").Value > ls\_stock Then

' MessageBox.Show("No cuenta con Stock suficiente !!" + Chr(13) + "Stock Disponible: " + ls\_stock.ToString, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

' dgv\_ventas.Rows(idx).Cells("cant").Value = ls\_stock

' Return

' End If

'End If

Try

Dim ls\_tipodoc As String = txt\_codtipodoc.Text.Trim

Dim ls\_nser As String = txt\_nser.Text.Trim

Dim ls\_ndoc As String = txt\_ndoc.Text.Trim

Dim ls\_codsuc As String = txt\_codsuc.Text.Trim

Dim ls\_stock As Decimal = 0

Dim ls\_frac\_ant As Decimal = 0

Dim idx As Byte = dgv\_ventas.CurrentRow.Index()

Dim ls\_codigo As String = dgv\_ventas.Rows(idx).Cells("codigo").Value

If dgv\_ventas.Rows(idx).Cells("cant").Value <> 0 Then

ls\_stock = objVentas.f\_retornaStock(ls\_codsuc, ls\_tipodoc, ls\_nser, ls\_ndoc, ls\_codigo, 0, 1)

If dgv\_ventas.Rows(idx).Cells("cant").Value > ls\_stock Then

MessageBox.Show("No cuenta con Stock suficiente !!" + Chr(13) + "Stock Disponible: " + ls\_stock.ToString, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

dgv\_ventas.Rows(idx).Cells("cant").Value = ls\_stock

Return

End If

End If

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

Sub f\_Costo\_minimo()

'Dim lrd\_costo As Decimal = 0

'Dim lrd\_Cero As Decimal = 0

'Dim idx As Byte = dgv\_ventas.CurrentRow.Index()

'Dim ls\_codigo As String = dgv\_ventas.Rows(idx).Cells("codigo").Value

''ls\_costo = f\_get\_data\_select("select costo from articulo where codarti='" & ls\_codigo & "'")

'lrd\_costo = FormatNumber(f\_get\_data\_select("select dbo.usp\_obtener\_costo\_promedio('" & mtb\_fecha.Text & "', '" & ls\_codigo & "', '" & txt\_codsuc.Text & "')as cp from articulo where codarti='" & ls\_codigo & "'"), 2)

'If String.IsNullOrEmpty(dgv\_ventas.Rows(idx).Cells("precio").FormattedValue.ToString) Then

' dgv\_ventas.Rows(idx).Cells("precio").Value = 0

'End If

'If CDec(dgv\_ventas.Rows(idx).Cells("precio").Value) <> CDec(lrd\_Cero) Then

' 'If CDec(dgv\_ventas.Rows(idx).Cells("precio").Value) < CDec(lrd\_costo) Then

' 'dgv\_ventas.Rows(idx).Cells("precio").Value = FormatNumber(lrd\_costo, 2)

' lbl\_CostoPromedio.Text = "Artículo: [ " & dgv\_ventas.Rows(idx).Cells("articulo").Value & "] Costo Promedio: " & FormatNumber(lrd\_costo, 2).ToString

' 'MessageBox.Show("Costo por debajo del Precio de Compra" + Chr(13) + "Precio de Costo: " + FormatNumber(lrd\_costo, 2).ToString, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

' 'Return

' 'End If

'End If

Call MostrarCostoPromedio()

End Sub

Sub MostrarCostoPromedio()

Dim lrd\_Cero As Decimal = 0

Dim i As Byte = dgv\_ventas.CurrentRow.Index()

Dim ls\_codigo As String = dgv\_ventas.Rows(i).Cells("codigo").Value

If String.IsNullOrEmpty(dgv\_ventas.CurrentRow.Cells("precio").FormattedValue.ToString) Then

dgv\_ventas.Rows(i).Cells("precio").Value = 0

'Else

' lrd\_dcto = FormatNumber(CDec(dgv\_ventas.Rows(i).Cells("dcto").Value), 3)

End If

If CDec(dgv\_ventas.Rows(i).Cells("precio").Value) <> CDec(lrd\_Cero) Then

lrd\_costo = FormatNumber(f\_get\_data\_select("select dbo.usp\_obtener\_costo\_promedio('" & mtb\_fecha.Text & "', '" & ls\_codigo & "', '" & txt\_codsuc.Text & "')as cp from articulo where codarti='" & ls\_codigo & "'"), 2)

If CDec(dgv\_ventas.Rows(i).Cells("precio").Value) < CDec(lrd\_costo) Then

lbl\_CostoPromedio.Text = "Artículo: [ " & dgv\_ventas.Rows(i).Cells("articulo").Value & "] Costo Promedio: " & FormatNumber(lrd\_costo, 2).ToString

lbl\_CostoPromedio.Visible = True

Else

lbl\_CostoPromedio.Visible = False

End If

End If

End Sub

Sub wf\_format\_detalle(ByVal as\_idx As Integer)

dgv\_ventas.ReadOnly = False

dgv\_ventas.Rows(as\_idx).Cells(0).ReadOnly = True

dgv\_ventas.Rows(as\_idx).Cells(1).ReadOnly = True

dgv\_ventas.Rows(as\_idx).Cells(2).ReadOnly = True

dgv\_ventas.Rows(as\_idx).Cells(3).ReadOnly = False

dgv\_ventas.Rows(as\_idx).Cells(4).ReadOnly = False

dgv\_ventas.Rows(as\_idx).Cells(5).ReadOnly = False ' No se pueda modificar el Descuento

dgv\_ventas.Rows(as\_idx).Cells(6).ReadOnly = True

End Sub

Private Sub btn\_agregar\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_agregar.Click

Call wf\_agregar\_articulos()

End Sub

Private Sub btn\_eliminar\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_eliminar.Click

Me.action = "1"

Call wf\_eliminar\_articulo()

End Sub

Sub wf\_eliminar\_articulo()

If Me.action = "1" Then

Dim idx As Byte = dgv\_ventas.CurrentRow.Index()

ls\_articulo = dgv\_ventas.Rows(idx).Cells(1).Value

Rpta = MessageBox.Show(gs\_app\_db\_question\_delete + " : " + ls\_articulo, gs\_app\_name, MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button2)

If Rpta = 6 Then

dgv\_ventas.Rows.RemoveAt(idx)

Call wf\_WidthColumn()

Call wf\_calcular\_importe()

End If

End If

End Sub

Sub f\_valida\_ingreso\_datos\_grilla()

'For i As Integer = 0 To dgv\_ventas.Rows.Count - 1

' If dgv\_ventas.Rows(i).Cells(3).Value.ToString.Trim.Length = 0 Then

' dgv\_ventas.Rows(i).Cells(3).Value = 0

' Return

' End If

' If dgv\_ventas.Rows(i).Cells(5).Value.ToString.Trim.Length = 0 Then

' dgv\_ventas.Rows(i).Cells(5).Value = 0

' Return

' End If

' If dgv\_ventas.Rows(i).Cells(4).Value.ToString.Trim.Length = 0 Then

' dgv\_ventas.Rows(i).Cells(4).Value = 0

' Return

' End If

'Next

End Sub

Public Sub wf\_calcular\_importe()

Dim lrd\_Resul As Decimal = 0

Dim lrd\_total As Decimal = 0

Dim lrd\_cant As Decimal = 0

Dim lrd\_prec As Decimal = 0

Dim li\_cant As Decimal = 0

Dim lrd\_dcto As Decimal = 0

Dim lrd\_totalPeso As Decimal = 0

Dim lrd\_peso As Decimal = 0

For i As Integer = 0 To dgv\_ventas.Rows.Count - 1

If String.IsNullOrEmpty(dgv\_ventas.CurrentRow.Cells("dcto").FormattedValue.ToString) Then

dgv\_ventas.Rows(i).Cells("dcto").Value = 0

Else

lrd\_dcto = FormatNumber(CDec(dgv\_ventas.Rows(i).Cells("dcto").Value), 3)

End If

If String.IsNullOrEmpty(dgv\_ventas.CurrentRow.Cells("cant").FormattedValue.ToString) Then

dgv\_ventas.Rows(i).Cells("cant").Value = 0

Else

li\_cant = FormatNumber(CDec(dgv\_ventas.Rows(i).Cells("cant").Value), 3)

End If

If String.IsNullOrEmpty(dgv\_ventas.CurrentRow.Cells("precio").FormattedValue.ToString) Then

dgv\_ventas.Rows(i).Cells("precio").Value = 0

Else

lrd\_prec = FormatNumber(CDec(dgv\_ventas.Rows(i).Cells("precio").Value), 3)

End If

If String.IsNullOrEmpty(dgv\_ventas.CurrentRow.Cells("peso").FormattedValue.ToString) Then

dgv\_ventas.Rows(i).Cells("peso").Value = 0

Else

lrd\_peso = FormatNumber(CDec(dgv\_ventas.Rows(i).Cells("peso").Value), 3)

End If

lrd\_Resul = FormatNumber((li\_cant \* lrd\_prec), 3)

lrd\_Resul = lrd\_Resul - (lrd\_Resul \* (lrd\_dcto / 100))

dgv\_ventas.Rows(i).Cells("importe").Value = FormatNumber(lrd\_Resul.ToString, 2)

lrd\_total = lrd\_total + (dgv\_ventas.Rows(i).Cells("importe").Value)

lrd\_cant = lrd\_cant + li\_cant

lrd\_totalPeso = lrd\_totalPeso + (lrd\_peso \* lrd\_cant)

dgv\_ventas.Rows(i).Cells("cant").Value = FormatNumber(dgv\_ventas.Rows(i).Cells("cant").Value, 0)

dgv\_ventas.Rows(i).Cells("precio").Value = FormatNumber(dgv\_ventas.Rows(i).Cells("precio").Value, 2)

dgv\_ventas.Rows(i).Cells("dcto").Value = FormatNumber(dgv\_ventas.Rows(i).Cells("dcto").Value, 2)

dgv\_ventas.Rows(i).Cells("importe").Value = FormatNumber(dgv\_ventas.Rows(i).Cells("importe").Value, 2)

Next

li\_impu = txt\_impuesto.Text

li\_igv = (lrd\_total \* li\_impu / 100)

txt\_subtotal.Text = FormatNumber(lrd\_total - li\_igv, 2)

txt\_igv.Text = FormatNumber(li\_igv, 2)

txt\_total.Text = FormatNumber(lrd\_total, 2)

lbl\_Cantidad.Text = "Nº de Items: " + dgv\_ventas.RowCount.ToString

lbl\_TotalPeso.Text = "Peso Total: " + FormatNumber(lrd\_totalPeso, 3).ToString

End Sub

Function wf\_validate()

If txt\_codcondicion.Text = "04" Then

txt\_codcta.Text = ""

txt\_cuenta.Text = ""

End If

txt\_fecha.Text = mtb\_fecha.Text

If mtb\_fecha.Text.Length <> 10 Then

MessageBox.Show("Fecha Incorrecta, Ingrese Fecha con el siguiente Formato: [dd/mm/aaaa]", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error)

mtb\_fecha.Focus()

Return False

End If

Call wf\_calcular\_importe()

txt\_usuario.Text = gs\_usureg

txt\_subtotal\_c.Text = f\_EliminarFormato(txt\_subtotal.Text)

txt\_mtoigv\_c.Text = f\_EliminarFormato(txt\_igv.Text)

txt\_total\_c.Text = f\_EliminarFormato(txt\_total.Text)

'select count(\*) as dato from cuenta where codsuc='01' and codcta='004'

ls\_codsuc = txt\_codsuc.Text.Trim

ls\_codcta = txt\_codcta.Text.Trim

ls\_codcondicion = txt\_codcondicion.Text.Trim

If ls\_codcondicion = "2" Then

If f\_get\_data\_select("select count(\*) as dato from cuenta where codsuc='" & ls\_codsuc & "' and codcta='" & ls\_codcta & "'") = 0 Then

MessageBox.Show("No coincide la Cuenta: " + txt\_cuenta.Text.Trim + Chr(13) + " en la Sucursal: " + txt\_sucursal.Text.Trim, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error)

Call wf\_obt\_cuentas()

Return False

End If

End If

'txt\_codconc.Text = "VTAS"

If chk\_vta\_adelantada.Checked = True Then

txt\_codconc.Text = "VTAA"

Else

txt\_codconc.Text = "VTAS"

End If

'Call wf\_calcular\_importe()

If f\_checkedvalue(txt\_codtipodoc) = False Then

Return False

End If

If f\_checkedvalue(txt\_nser) = False Then

Return False

End If

If f\_checkedvalue(txt\_ndoc) = False Then

Return False

End If

If f\_checkedvalue(txt\_codclie) = False Then

Return False

End If

If f\_checkedvalue(txt\_codmoneda) = False Then

Return False

End If

If f\_checkedvalue(txt\_codsuc) = False Then

Return False

End If

If f\_checkedvalue(txt\_codcondicion) = False Then

Return False

End If

If txt\_codcondicion.Text = "2" Then

If f\_checkedvalue(txt\_codcta) = False Then

Return False

End If

End If

If f\_checkedvalue(txt\_codvend) = False Then

Return False

End If

If f\_checkedvalue(txt\_idmodalidad) = False Then

Return False

End If

If txt\_codmoneda.Text <> "S" Then

MessageBox.Show("Está intentando generar una Venta con un tipo de Moneda que no es SOLES a S/.", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error)

txt\_codmoneda.Focus()

Return False

End If

If mtb\_fecha.Text = "" Then

MessageBox.Show("Fecha Incorrecta, Ingrese Fecha con el siguiente Formato: [dd/mm/aaaa]", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error)

mtb\_fecha.Focus()

Return False

End If

If dgv\_ventas.Rows.Count = 0 Then

MessageBox.Show("Ingrese Detalle de la Venta", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error)

dgv\_ventas.Focus()

Return False

End If

For i As Integer = 0 To dgv\_ventas.RowCount - 1

If dgv\_ventas.Rows(i).Cells("cant").Value = 0 Then

Dim ls\_codigo As String = dgv\_ventas.Rows(i).Cells("codigo").Value

Dim ls\_nombre1 As String = f\_get\_data\_select("select nombre from articulo where codarti='" & ls\_codigo & "'")

MessageBox.Show("Ingrese una Cantidad en Enteros ó Fracción para el Artículo: " + Chr(13) + ls\_nombre1, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

dgv\_ventas.Focus()

Return False

End If

Next

For j As Integer = 0 To dgv\_ventas.RowCount - 1

If dgv\_ventas.Rows(j).Cells("precio").Value = 0 Then

Dim ls\_codigo As String = dgv\_ventas.Rows(j).Cells("codigo").Value

Dim ls\_nombre2 As String = f\_get\_data\_select("select nombre from articulo where codarti='" & ls\_codigo & "'")

MessageBox.Show("Ingrese Precio de Venta para el Artículo: " + Chr(13) + ls\_nombre2, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

dgv\_ventas.Focus()

Return False

End If

Next

Return True

End Function

Private Sub Btn\_salir\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_salir.Click

Me.Close()

End Sub

Private Sub Btn\_guardar\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_guardar.Click

If wf\_agregar() = True Then

Me.action = "0"

Call wf\_limpiar\_controles()

Me.Close()

End If

End Sub

Function wf\_agregar()

Try

If wf\_validate() = False Then

Return False

End If

If MessageBox.Show(gs\_app\_db\_question\_save, gs\_app\_name, MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button1) = 6 Then

Dim mensaje As String = ""

Dim tipo As String = IIf(Me.dbaction = "insert", "I", "M")

Dim nitem As Integer = 0

If Me.dbaction = "insert" Then

txt\_ndoc.Text = objBasicas.f\_obt\_num\_doc(txt\_codtipodoc.Text, txt\_nser.Text, txt\_codsuc.Text).ToString.PadLeft(7, "0")

End If

For i = 0 To dgv\_ventas.Rows.Count - 1

nitem = nitem + 1

Dim codigo As String = dgv\_ventas.Rows(i).Cells("codigo").Value

Dim cantidad As Decimal = dgv\_ventas.Rows(i).Cells("cant").Value

Dim precio As Decimal = dgv\_ventas.Rows(i).Cells("precio").Value

Dim dcto As Decimal = dgv\_ventas.Rows(i).Cells("dcto").Value

'mensaje = objVentas.usp\_RegistarVentas(txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, txt\_codclie.Text, mtb\_fecha.Text, txt\_subtotal.Text, \_

' txt\_igv.Text, txt\_total.Text, txt\_codmoneda.Text, txt\_codvend.Text, txt\_detalle.Text, txt\_codcondicion.Text, \_

' txt\_dolar.Text, txt\_dias.Text, txt\_codcta.Text, txt\_impuesto.Text, txt\_codsuc.Text, txt\_usuario.Text, \_

' txt\_codconc.Text, txt\_idmodalidad.Text, "", codigo, nitem, cantidad, precio, dcto, \_

' 0, txt\_codclie\_ant.Text, txt\_opci.Text, tipo)

objVentas.usp\_RegistarVentas(txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, txt\_codclie.Text, mtb\_fecha.Text, txt\_subtotal.Text, \_

txt\_igv.Text, txt\_total.Text, txt\_codmoneda.Text, txt\_codvend.Text, txt\_detalle.Text, txt\_codcondicion.Text, \_

txt\_dolar.Text, txt\_dias.Text, txt\_codcta.Text, txt\_impuesto.Text, txt\_codsuc.Text, txt\_usuario.Text, \_

txt\_codconc.Text, txt\_idmodalidad.Text, "", codigo, nitem, cantidad, precio, dcto, \_

0, txt\_codclie\_ant.Text, txt\_opci.Text, tipo)

Next

'If mensaje = "COMMIT" Then

'Else

' MessageBox.Show("¡ERROR! al Realizar la Transacción, No se Guardarón los DATOS", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1)

'End If

If MessageBox.Show("Desea Imprimir", gs\_app\_name, MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button1) = 6 Then

''Dim li\_count As Integer = f\_get\_data\_select("select valor from parametros where codigo='011'")

'Dim li\_msg As String = ""

'For j As Integer = 1 To 2 'li\_count

' Select Case j

' Case 1

' li\_msg = "\*\*\* CONTROL ADMINISTRATIVO \*\*\*"

' Case 2

' li\_msg = "\*\*\* CLIENTE \*\*\*"

' 'Case 3

' ' li\_msg = "\*\*\* TIENDA \*\*\*"

' 'Case Else

' ' li\_msg = "\*\*\* OTROS \*\*\*"

' End Select

' Call wf\_imprimir(False, li\_msg)

'Next

Call wf\_imprimir(False, "\*\*\* CONTROL ADMINISTRATIVO \*\*\*")

Call wf\_imprimir(False, "\*\*\* CLIENTE \*\*\*")

End If

Me.Close()

End If

Return True

Catch ex As Exception

MsgBox(ex.Message)

Me.Close()

Return False

End Try

End Function

'Private Sub Btn\_guardar\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_guardar.Click

' Call wf\_agregar()

'End Sub

'Sub wf\_agregar()

' If wf\_validate() = False Then

' Return

' End If

' If MessageBox.Show(gs\_app\_db\_question\_save, gs\_app\_name, MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button1) = 6 Then

' Call wf\_actualizar\_stock\_venta(txt\_codsuc.Text, txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, "1")

' Call wf\_eliminar()

' If dbpanel.dbUpdate = True Then

' If Me.dbaction = "insert" Then

' If wf\_guardar() = False Then

' Return

' End If

' ' actualizar correlativo

' objVentas.f\_act\_correlativo(txt\_codsuc.Text.Trim, txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text)

' If wf\_guardar\_movp("1") = False Then

' Return

' End If

' Else ' modificar

' If wf\_eliminar() = False Then

' Return

' End If

' If wf\_guardar() = False Then

' Return

' End If

' If wf\_guardar\_movp("2") = False Then

' Return

' End If

' End If

' End If

' Me.action = "0"

' If MessageBox.Show("Desea Imprimir", gs\_app\_name, MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button1) = 6 Then

' Dim li\_count As Integer = f\_get\_data\_select("select valor from parametros where codigo='011'")

' Dim li\_msg As String = ""

' For i As Integer = 1 To li\_count

' 'Select Case i

' 'Case 1

' ' li\_msg = "\*\*\* MOLINO \*\*\*"

' 'Case 2

' li\_msg = "\*\*\* CLIENTE \*\*\*"

' 'Case 3

' ' li\_msg = "\*\*\* TIENDA \*\*\*"

' 'Case Else

' ' li\_msg = "\*\*\* OTROS \*\*\*"

' 'End Select

' 'Call wf\_ImpresionVenta()

' Call wf\_imprimir(False, li\_msg)

' Next

' End If

' Call wf\_limpiar\_controles()

' Me.Close()

' 'Else

' ' MessageBox.Show("Etapa de prueba ha caducado" + Chr(13) + "Comuniquese con el programador", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1)

' ' Me.action = "0"

' ' End If

' 'Else

' 'MsgBox("Periodo de Prueba Caduco")

' End If

'End Sub

'Function wf\_guardar()

' Dim objArticulos As New cls\_mant\_articulos

' Dim objStockSucursal As New cls\_stock\_sucursal

' Try

' objVentas.f\_guardar(UpDatePanel(dbpanel))

' nitem = 0

' For i = 0 To dgv\_ventas.Rows.Count - 1

' nitem = nitem + 1

' objVentas.f\_guardar\_detalle(txt\_codsuc.Text, txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, dgv\_ventas.Rows(i).Cells(0).Value, nitem, dgv\_ventas.Rows(i).Cells(3).Value, dgv\_ventas.Rows(i).Cells(4).Value, dgv\_ventas.Rows(i).Cells(5).Value, txt\_impuesto.Text)

' objStockSucursal.f\_actualizar(dgv\_ventas.Rows(i).Cells(0).Value, txt\_codsuc.Text.Trim, dgv\_ventas.Rows(i).Cells(3).Value, "2")

' Next

' Return True

' Catch ex As Exception

' Return False

' End Try

'End Function

'Function wf\_guardar\_movp(ByVal as\_tipo As String)

' Dim ObjMant As New cls\_mant\_mov\_clie

' ls\_codcond = txt\_opci.Text

' ls\_codclie = txt\_codclie.Text

' ls\_codclie\_ant = txt\_codclie\_ant.Text

' If as\_tipo = "1" Then

' ObjMant.f\_guardar\_venta(txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, txt\_codsuc.Text)

' Else

' 'Si no existe la Venta Insertarla

' ObjMant.f\_modificar\_venta(txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, ls\_codclie, ls\_codcond, ls\_codclie\_ant, txt\_codsuc.Text)

' End If

' Return True

'End Function

'Sub wf\_actualizar\_stock\_venta(ByVal as\_codsuc As String, ByVal as\_tipodoc As String, ByVal as\_nser As String, ByVal as\_ndoc As String, ByVal as\_tipo As String)

' Dim objStockSucursal As New cls\_stock\_sucursal

' Dim dtt As DataTable

' dtt = objVentas.f\_obt\_stock(as\_codsuc, as\_tipodoc, as\_nser, as\_ndoc)

' For i As Integer = 0 To dtt.Rows.Count - 1

' ls\_codarti = dtt.Rows(i).Item(0)

' lrd\_cant = dtt.Rows(i).Item(1)

' ls\_codsuc = dtt.Rows(i).Item(2)

' objStockSucursal.f\_actualizar(ls\_codarti, ls\_codsuc, lrd\_cant, as\_tipo)

' Next

'End Sub

'Function wf\_eliminar()

' Try

' objVentas.f\_eliminar\_venta\_detalle(txt\_codsuc.Text, txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text)

' Return True

' Catch ex As Exception

' Return False

' End Try

'End Function

Sub wf\_modificar(ByVal as\_codsuc As String, ByVal as\_tipodoc As String, ByVal as\_nser As String, ByVal as\_ndoc As String)

li\_dato = objVentas.f\_ver\_existe(as\_codsuc, as\_tipodoc, as\_nser, as\_ndoc)

If li\_dato > 0 Then

txt\_codsuc.Text = as\_codsuc

txt\_codtipodoc.Text = as\_tipodoc

txt\_nser.Text = as\_nser

txt\_ndoc.Text = as\_ndoc

MessageBox.Show(gs\_app\_db\_documento\_existe, gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Information)

Me.dbaction = "update"

Me.dbpanel.dbaction = "update"

txt\_codsuc.Enabled = False

txt\_codtipodoc.Enabled = False

txt\_nser.Enabled = False

txt\_ndoc.Enabled = False

' Recuperar datos de tabla Compra

objVentas.f\_modificar(RetrievePanel(dbpanel))

Call wf\_obt\_desc()

mtb\_fecha.Text = txt\_fecha.Text

txt\_subtotal.Text = txt\_subtotal\_c.Text

txt\_igv.Text = txt\_mtoigv\_c.Text

txt\_total.Text = txt\_total\_c.Text

If txt\_codcondicion.Text = "1" Then

txt\_codcta.Enabled = False

Btn\_cuenta.Enabled = False

Else

txt\_codcta.Enabled = True

End If

txt\_opci.Text = txt\_codcondicion.Text

If txt\_codconc.Text = "VTAA" Then

chk\_vta\_adelantada.Checked = True

Else

chk\_vta\_adelantada.Checked = False

End If

btn\_sucursal.Enabled = False

Btn\_documento.Enabled = False

If gs\_user\_grupo = "01" Or gs\_user\_grupo = "99" Then

txt\_codvend.Enabled = True

Btn\_vendedor.Enabled = True

Else

txt\_codvend.Enabled = False

Btn\_vendedor.Enabled = False

End If

' Recuperar datos de tabla Venta\_detalle

Call wf\_recuperar\_venta\_detalle()

Call wf\_calcular\_importe()

' Habilitar la escritura en el grid

If dgv\_ventas.Rows.Count > 0 Then

Dim idx As Integer = dgv\_ventas.Rows.Count - 1

dgv\_ventas.Rows(idx).Cells(3).Selected = True

Call wf\_format\_detalle(idx)

End If

Call wf\_WidthColumn()

mtb\_fecha.Focus()

End If

End Sub

Sub wf\_recuperar\_venta\_detalle()

Dim dtt As DataTable

dtt = objVentas.f\_consultar(txt\_codsuc.Text, txt\_codtipodoc.Text, txt\_nser.Text, txt\_ndoc.Text, "2")

For i As Integer = 0 To dtt.Rows.Count - 1

dgv\_ventas.Rows.Add(dtt.Rows(i).Item("codarti"), dtt.Rows(i).Item("nombre"), dtt.Rows(i).Item("unidadm"), dtt.Rows(i).Item("cant"), \_

dtt.Rows(i).Item("precio"), dtt.Rows(i).Item("dcto"), dtt.Rows(i).Item("item"), dtt.Rows(i).Item("peso"))

Next

End Sub

Private Sub Btn\_documento\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_documento.Click

Call wf\_obt\_tipodoc()

End Sub

Private Sub Btn\_cliente\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_cliente.Click

Call wf\_obt\_cliente()

End Sub

Private Sub Btn\_moneda\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_moneda.Click

Call wf\_obt\_moneda()

End Sub

Private Sub btn\_sucursal\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_sucursal.Click

Call wf\_obt\_sucursal()

End Sub

Private Sub btn\_condicion\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_condicion.Click

Call wf\_obt\_condicion()

End Sub

Private Sub Btn\_cuenta\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_cuenta.Click

Call wf\_obt\_cuentas()

End Sub

Private Sub Btn\_vendedor\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_vendedor.Click

Call wf\_obt\_vendedor()

End Sub

Private Sub txt\_idmodalidad\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txt\_idmodalidad.KeyDown

If e.KeyCode = Keys.Enter Then

Call wf\_sel\_modalidad()

End If

End Sub

Sub wf\_sel\_modalidad()

Dim ObjModalidad As New cls\_mant\_modalidad

li\_dato = ObjModalidad.f\_ver\_existe(Trim(txt\_idmodalidad.Text), "2")

If li\_dato > 0 Then

' txt\_cuenta.Text = ObjCuentas.obt\_nombre(txt\_codcta.Text)

Else

Call wf\_obt\_modalidad()

End If

End Sub

Sub wf\_obt\_modalidad()

txt\_idmodalidad.Text = Search.f\_buscar\_modalidad("0", "1")

End Sub

Private Sub Btn\_helpme1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn\_helpme1.Click

Call wf\_obt\_modalidad()

End Sub

Private Sub mtb\_fecha\_TypeValidationCompleted(ByVal sender As Object, ByVal e As System.Windows.Forms.TypeValidationEventArgs) Handles mtb\_fecha.TypeValidationCompleted

If Not e.IsValidInput Then

MessageBox.Show("Fecha Incorrecta, Ingrese Fecha con el siguiente Formato: [dd/mm/aaaa]", gs\_app\_name, MessageBoxButtons.OK, MessageBoxIcon.Error)

mtb\_fecha.Focus()

Return

e.Cancel = True

End If

End Sub

Private Sub dgv\_ventas\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles dgv\_ventas.Click

MostrarCostoPromedio()

End Sub

Private Sub dgv\_ventas\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles dgv\_ventas.KeyDown

MostrarCostoPromedio()

End Sub

End Class