

MANTENIMIENTO DE LA TABLA PRODUCTOS UTILIZANDO LINQ TO ENTITIES Y DATAGRIDVIEW

Caso N° 03: Mantenimiento de Productos

Aplicación que permite realizar el mantenimiento de los registros pertenecientes a la tabla Productos de la base de datos Ventas, usando el modelo ADO Entity.

Tener en cuenta:

- Al inicial el formulario deberá mostrar los datos de los productos en un control DataGridView.
- Al presionar el botón Registrar deberá registrar los datos del productos ingresado y mostrarlos en el control DataGridView.
- Para la actualización deberá presionar doble clic sobre un producto desde el control DataGridView y enviar los datos a las controles TextBox para poder modificarlos.

Pasos:

1. Diseñar la siguiente GUI:

	COD_PRO	DESCRIP_PRO	PRECIO_PRO	STOCK_ACT_PRO	STOCK_MIN_PRO
	P013	TORTA DE FRESA	41.0000	100	60
	P014	ALFAJORES	0.3000	400	300
	P015	CHOCOTEJAS	2.0000	100	80
	P016	SUSPIRO A LA LIMEÑA	3.5000	100	80
	P017	Mouse de Chocolate	15.75	100	20

2. Agregar el siguiente script al load del formulario:

```
Imports System.Data.SqlClient
```

```
Public Class Form1
```

```
Dim cn As New SqlConnection("Server=.;DataBase=Ventas2014;uid=sa;pwd=sql")  
Dim db As New VENTAS2014Entities
```

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load  
    Call llenarProductos()  
End Sub
```

```
Sub llenarProductos()  
    dgProductos.DataSource = db.PRODUCTO.ToList  
End Sub
```

```
Private Sub dgProductos_CurrentCellChanged(sender As Object, e As EventArgs)  
    Try  
        txtCodigo.Text = dgProductos.CurrentRow.Cells(0).Value  
        txtDescripcion.Text = dgProductos.CurrentRow.Cells(1).Value  
        txtPrecio.Text = dgProductos.CurrentRow.Cells(2).Value  
        txtStockA.Text = dgProductos.CurrentRow.Cells(3).Value  
        txtStockM.Text = dgProductos.CurrentRow.Cells(4).Value  
    Catch ex As Exception  
    End Try  
End Sub
```

```
Private Sub btnRegistrar_Click(sender As Object, e As EventArgs) Handles  
Dim objPro = db.PRODUCTO.Find(txtCodigo.Text)  
If objPro Is Nothing Then  
    objPro = New PRODUCTO  
    Call Autogenera_Codigo()  
    objPro.COD_PRO = txtCodigo.Text  
    objPro.DESCRIP_PRO = txtDescripcion.Text  
    objPro.PRECIO_PRO = txtPrecio.Text  
    objPro.STOCK_ACT_PRO = txtStockA.Text  
    objPro.STOCK_MIN_PRO = txtStockM.Text  
    db.PRODUCTO.Add(objPro)  
    db.SaveChanges()  
    Call llenarProductos()  
Else  
    MessageBox.Show("Codigo Registrado!!!")  
End If  
End Sub
```

```
Private Sub btnActualizar_Click(sender As Object, e As EventArgs) Handles  
Dim objPro As PRODUCTO=db.PRODUCTO.ToList.ElementAt(dgProductos.CurrentRow.Index)  
objPro.COD_PRO = txtCodigo.Text  
objPro.DESCRIP_PRO = txtDescripcion.Text  
objPro.PRECIO_PRO = txtPrecio.Text  
objPro.STOCK_ACT_PRO = txtStockA.Text  
objPro.STOCK_MIN_PRO = txtStockM.Text  
db.SaveChanges()  
Call llenarProductos()  
End Sub
```

```
Sub Autogenera_Codigo()  
    cn.Open()  
    Using cmd As New SqlCommand  
        cmd.CommandType = CommandType.StoredProcedure  
        cmd.Connection = cn  
        cmd.CommandText = "sp_generaCodigo"  
        With cmd.Parameters  
            .Add("@Cod", SqlDbType.Char, 4).Direction = ParameterDirection.Output  
        End With  
        cmd.ExecuteNonQuery()  
        Dim NewCodigo As String  
        NewCodigo = cmd.Parameters(0).Value.ToString  
        cn.Close()  
    End Using  
End Sub
```

```
        txtCodigo.Text = NewCodigo
    End Using
End Sub
```

```
Sub LimpiarControles()
    Dim misObjetos As Control
    For Each misObjetos In Me.Controls
        If TypeOf misObjetos Is System.Windows.Forms.TextBox Then
            misObjetos.Text = ""
        End If
    Next
End Sub
```

```
Private Sub btnNuevo_Click(sender As Object, e As EventArgs) Handles
    CALL LimpiarControles()
End Sub
```

3.- Crear el siguiente procedimiento almacenado:

```
-- autogenera el codigo del Producto
If OBJECT_ID('sp_generaCodigo') is not null
    drop proc sp_generaCodigo
go
Create procedure sp_generaCodigo
@cod varchar (20) output
as
declare @aux varchar(20)
select @aux = max(right(COD_PRO,3)) from PRODUCTO
select @cod = 'P'+right(str(convert(int,@aux)+10001),3)
--noten que en este procedimiento el parámetro @cod dice
output que quiere decir que es de salida.
go

--para probar nuestro sp seria de la siguiente manera
--(esto siempre y cuando el sp vote un parámetro de
--salida):
declare @aux varchar(20)
execute sp_generaCODIGO @aux output
select @aux
```