



Programación Orientada A Objetos II Guía Práctica Nº 09 LinQ

COMPETENCIAS:

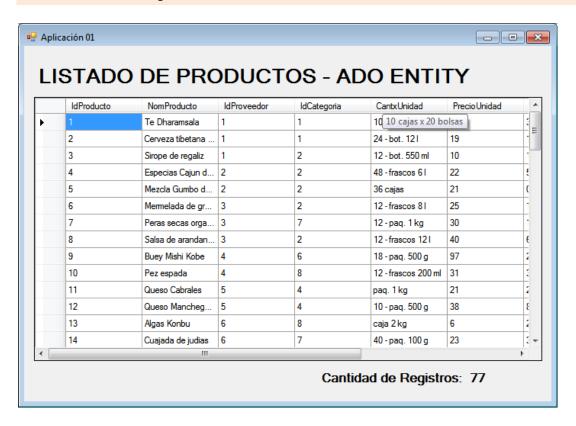
- Implementa aplicaciones usando el modelo relacional de Visual Studio con Ado Entity.

Caso 1: Listado de Productos

Aplicación que permite mostrar un listado de los productos de la base de datos Negocios 2014 usando el modelo ADO Entity.

Tener en cuenta:

- Mostrar los registros en un control DataGridView.
- Mostrar el total de registros encontrados.



Public Class Form1

Dim Modelo As New Negocios2014Entities

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
    Dim lista = From c In Modelo.productos Select c
    DataGridView1.DataSource = lista.ToList
    lblCantidad.Text = lista.Count
End Sub
```

Caso 1: Listado de Productos

Aplicación que permite mostrar un listado de los productos de la base de datos Negocios 2014 usando el modelo ADO Entity.

Tener en cuenta:

- Mostrar los registros en un control ListView.
- Mostrar el total de registros encontrados.



Imports System.Data.EntityClient
Public Class Form2

```
Sub FormatoLista()
    With ListView1
    .View = View.Details
    .GridLines = True
    .Columns.Add("IdProducto", 100)
    .Columns.Add("NomProducto", 100)
    .Columns.Add("IdProveedor", 100)
    .Columns.Add("IdCategoria", 100)
    .Columns.Add("CantxUnidad", 100)
    End With
End Sub
```

```
Sub LlenaLista()
Using cn As New EntityConnection("Name=Negocios2014Entities")
Dim cmd As New EntityCommand("Select VALUE Prod from
Negocios2014Entities.productos as prod", cn)
cn.Open()
Dim dr As EntityDataReader = cmd.ExecuteReader(CommandBehavior.SequentialAccess)
  While dr.Read
     Dim it As New ListViewItem(dr("IdProducto").ToString)
      it.SubItems.Add(dr("NomProducto"))
      it.SubItems.Add(dr("IdProveedor"))
      it.SubItems.Add(dr("IdCategoria"))
      it.SubItems.Add(dr("CantxUnidad"))
     ListView1.Items.Add(it)
  End While
  lblCantidad.Text = ListView1.Items.Count
   dr.Close()
End Using
End Sub
```

End Class

Caso Nº 02: Listado de Clientes por País

Aplicación que permite mostrar un listado de los clientes según el país seleccionado desde un cuadro combinado, usando el modelo ADO Entity.

Tener en cuenta:

- Al inicial el formulario deberá mostrar los datos de los paises dentro de un cuadro combinado y los datos de los clientes en un control DataGridView.
- Al seleccionar un país y presionar el botón Buscar deberá filtrar los clientes que cumplan con la condición.



Imports System.Data.EntityClient

```
Imports System.Data.SqlClient
Public Class Form3
    Dim db As New Negocios2014Entities

Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As System.E
    DataGridView1.DataSource = db.clientes.ToList
    ComboBox1.DataSource = db.paises.ToList
    ComboBox1.DisplayMember = "NombrePais"
    ComboBox1.ValueMember = "IdPais"
End Sub
```

```
Private Sub btnConsultar_Click(ByVal sender As System.Object, ByVal e As
Dim datos As New ArrayList
Dim cn As New EntityConnection("Name=Negocios2014Entities")
Dim cmd As New EntityCommand("Select VALUE cli from Negocios2014Entities.clientes
as cli where cli.idPais=@cod", cn)
With cmd.Parameters
   .Add("cod", DbType.String).Value = ComboBox1.SelectedValue
End With
cn.Open()
Dim dr As EntityDataReader = cmd.ExecuteReader(CommandBehavior.SequentialAccess)
DataGridView1.DataSource = Nothing
Do While dr.Read
   Dim ObjCliente As New clientes
   ObjCliente.IdCliente = dr.GetString(0)
   ObjCliente.NomCliente = dr.GetString(1)
   ObjCliente.DirCliente = dr.GetString(2)
   ObjCliente.idpais = dr.GetString(3)
  ObjCliente.fonoCliente = dr.GetString(4)
   datos.Add(ObjCliente)
DataGridView1.DataSource = datos
dr.Close()
cn.Close()
End Sub
```

End Class