

Capítulo 6

Fecha Release: Agosto 26, 2006

Versión: 1.0

Ultima Actualización: Agosto 27, 2006

6. Agregando un formulario con un combobox

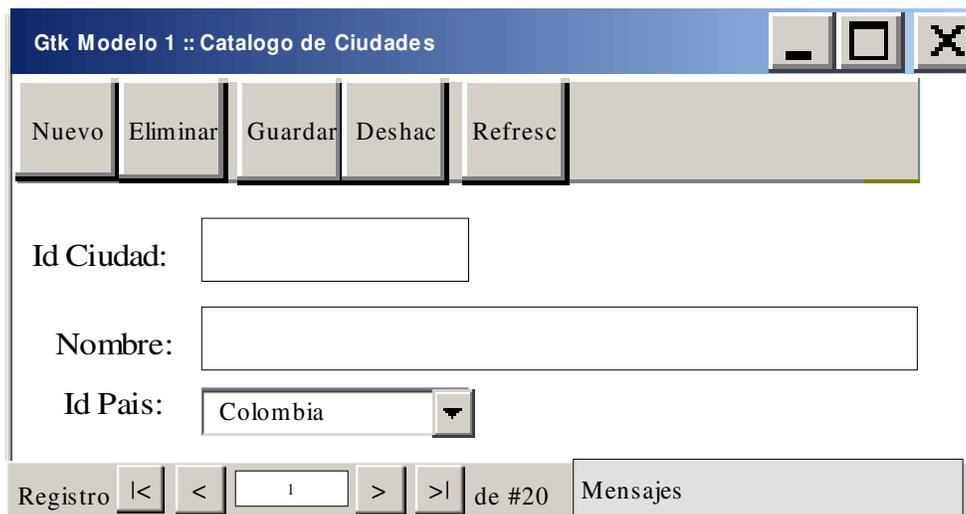
Aprovecharemos en este capítulo para diseñar una interfaz gráfica con un elemento muy útil: un combobox con información de una tabla.

En realidad para la parte de diseño, lo que haremos es aprovechar el formulario que elaboramos para el catálogo de países, copiarlo, agregar el combobox y ajustar las propiedades propias de los países. Hemos considerado incluir este tipo de formularios, ya que contiene al elemento combobox, un widget que puede llegar a causar algunos problemas cuando se intenta que incluya los datos de una tabla almacenada en un servidor de datos.

Para este capítulo necesitamos el IDE de Visual Studio, el proyecto de ejemplo que venimos construyendo a la largo de estos capítulos, el Glade y lo más importante: una cerveza (Destápela y empecemos a diseñar el formulario).

6.1. Diseñando el formulario para el catálogo de ciudades

Abra el IDE de Visual Studio, cargue el proyecto *GtkModelo1* y en el explorador de soluciones haga doble clic sobre el objeto Glade (*gladesharp.glade*). Ahora ya debe estar el Glade cargado en la pantalla. Demos un vistazo al formulario que queremos lograr para las ciudades:



The screenshot shows a window titled "Gtk Modelo 1 :: Catalogo de Ciudades". The window has a menu bar with buttons: "Nuevo", "Eliminar", "Guardar", "Deshac", "Refresc". Below the menu bar, there are three input fields: "Id Ciudad:" with a text box, "Nombre:" with a text box, and "Id Pais:" with a combobox showing "Colombia". At the bottom, there is a status bar with "Registro" and navigation buttons, a page indicator "1 de #20", and a "Mensajes" button.

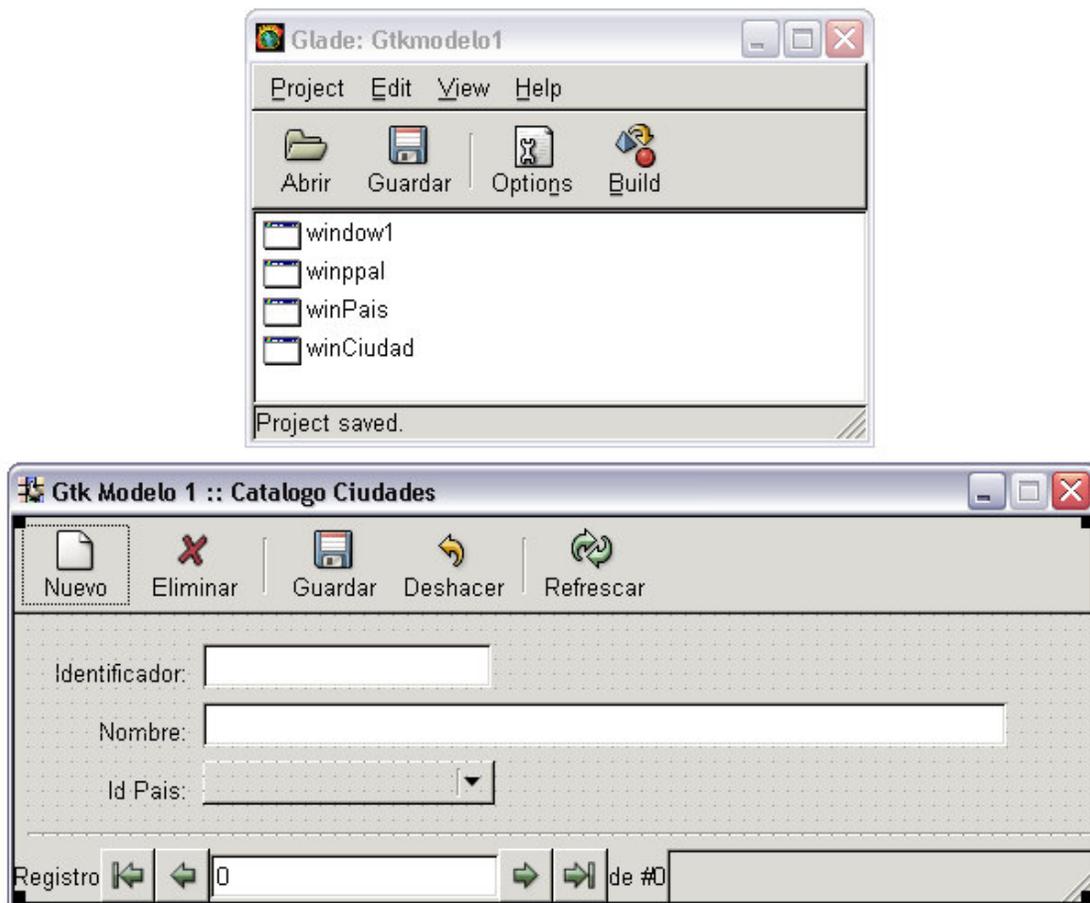
Respecto al formulario diseñado para el catálogo de países, solo difiere en algunos textos, título de ventana y la lista desplegable (combobox en Gtk) para los países. Como da la casualidad que es un formulario igual en muchas cosas, aprovecharemos la capacidad que tiene Glade de duplicar objetos.

En la ventana del proyecto Glade haga clic sobre el nombre del formulario winPais. Ahora haga clic en la opción *Edit* (Editar) y seleccione *Copy* (Copiar). Ahora ya tenemos el formulario duplicado en la memoria con todos los objetos que diseñamos para el catálogo de países.

Ahora haga clic en *Edit* (Editar) y seleccione *Paste* (Pegar). Glade automáticamente abre el nuevo formulario y asigna como nombre de objeto, el nombre de *windown*, donde *n* será el máximo número disponible en el orden de objetos existentes similares.

Lo que haremos ahora es ajustar el nuevo formulario para que permita manipular el catálogo de ciudades. Una mala noticia es que debemos volver a nombrar cada uno de los objetos que necesitamos alcanzar desde el código. Revise el capítulo 5 y nombre nuevamente a los objetos del formulario. Para el nombre de este formulario, especifique el nombre de *winCiudad* y como título de ventana (*Title*) especifique *Gtk Modelo 1 :: Catalogo Ciudades*.

Organice el área de datos del formulario de forma tal que quede espacio para un nuevo texto (Label) y una lista desplegable (*ComboBox*). Inserte un objeto Label debajo de la etiqueta nombre. Al objeto Label especifique etiqueta (Label) el texto “*Id Pais:*”. Ahora inserte un objeto ComboBox al lado de la etiqueta “*Id Pais:*”. Especifique como nombre del combobox el de *listaPais*. Su formulario debe lucir como la siguiente gráfica:



Olvidó asignar el mismo icono que tiene el botón ciudad en el formulario principal de la aplicación?

Ahora solo falta indicarle a Glade que guarde y que cree el código XML que necesitamos para poder utilizar este formulario desde nuestro código. Guarde y genere el código (ya olvidó como????). Cierre el Glade.

6.2. Agregando el catálogo de ciudades a la GUI principal

Solo nos falta definir la clase que cargará este nuevo formulario. Como podrá imaginar en este punto, nos dedicaremos a crear una clase a partir de la que ya tenemos definida para el catálogo de los Países (*clsCatPais.cs*). En el explorador de soluciones del IDE Visual Studio, haga clic derecho sobre la clase *clsCatPais.cs* y seleccione *Copiar*. Para crear la nueva clase a partir de esta copia en memoria, haga clic derecho sobre el nombre de la solución *gtkmodelo1* y seleccione *Pegar*.

Ahora solo hace falta que renombramos la clase Copia de *clsCatPais.cs* al nombre *clsCatCiudad.cs*. Para hacerlo haga clic derecho sobre *Copia de clsCatPais.cs* y seleccione del menú emergente *Cambiar Nombre* y escriba como nombre *clsCatCiudad.cs*.

Haga doble clic sobre la clase *clsCatCiudad.cs*. Lo que haremos es ajustar el código para que cargue el formulario del catálogo de las ciudades. El código de la clase debe quedar como aparece a continuación (en **negrita** lo que debe ser ajustado):

```
using System;
using Gtk;
using Glade;

namespace gtkmodelo1
{
    /// <summary>
    /// Descripción breve de clsCatPais.
    /// </summary>
    public class clsCatCiudad
    {
        #region Glade objects
        //hacer referencias a los objetos Glade
        public Gtk.Window win;
        Glade.XML xml;
        #endregion Glade objects

        public clsCatCiudad()
        {
            //cargar formulario winPais
            xml = new Glade.XML( null, "gladesharp1.glade", "winCiudad", null);
            xml.Autoconnect(this);
            win = (xml.GetWidget("winCiudad") as Gtk.Window);
        }
    }
}
```

Observe que los cambios son mínimos en el código en el código. Compile para que verifique si no tiene errores en el código.

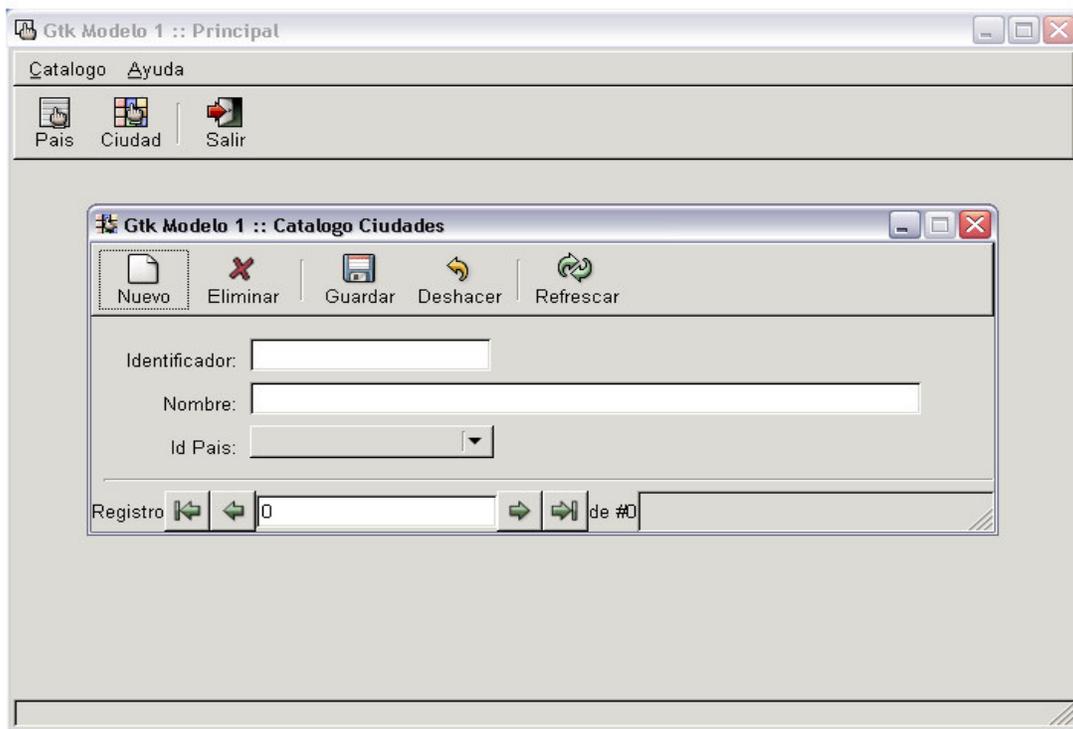
A continuación agregaremos el código que vinculará el formulario del catálogo de ciudades a la GUI principal. Este formulario debe ser cargado por medio de una opción en el submenú catálogo y por un botón disponible en la barra de herramientas. En el explorador de soluciones haga doble clic sobre la clase Driver.cs (nuestro programa principal). Cuando se cargue la ventana de código busque la rutina `on_tbCiudad_clicked` y escriba el siguiente código (en **negrita**):

```
protected void on_tbCiudad_clicked(object o, EventArgs args)
{
    clsCatCiudad ventana1 = new clsCatCiudad();
    ventana1.win.ShowAll();
    return;
}
```

A continuación busque la rutina `on_opcCiudad_activate` y escriba el siguiente código (en **negrita**):

```
protected void on_opcCiudad_activate(object o, EventArgs args)
{
    clsCatCiudad ventana1 = new clsCatCiudad();
    ventana1.win.ShowAll();
    return;
}
```

Ya tenemos la clase catálogo ciudad vinculada a la GUI principal. Compile y ejecute la aplicación. Cuando haga clic sobre el botón Ciudad de la barra de herramientas o seleccione la opción Ciudad del menú Catálogo debe cargar el formulario para las ciudades. A continuación una gráfica que nos muestra como luce el nuevo formulario cuando se le carga a través de nuestra GUI principal:



6.3. Resumen

En este capítulo aprendimos a elaborar diseños Glade a partir de otros existentes en el proyecto. Adicionamos un elemento extra típico en los formularios de las aplicaciones: el combobox o lista desplegable.

Hasta este capítulo llega la primera parte del curso de programación de GUIs para Windows y Linux. Esta primera parte trató acerca de desarrollar la habilidad necesaria para diseñar interfaces gráficas con la herramienta Glade. Además aprendimos como integrar los diseños elaborados con Glade a un proyecto desarrollado con Visual Studio 2003.

La segunda parte trata específicamente de integrar el motor de bases de datos mySQL a un proyecto que utiliza una GUI en el motor gráfico Gtk. Aprenderemos como escribir el código que vincule las tablas de una base de datos mySQL a un formulario que emplea widgets de Gtk.

Cordialmente,

Mauricio Cano Ossa

Un discípulo más del dios de los monos...