**Controles de Windows Forms**

**El control ComboBox**

**Autor:**

**Samuel Cruz Parra**

# El control ComboBox

## ¿En qué consiste?

El control ComboBox es utilizado para mostrar un conjunto de datos dentro de un cuadro desplegable. En este sentido, podemos tanto escribir en el espacio que nos deja la selección o elegir un elemento ya dentro de la “lista”.

## Constructores para ComboBox

Perteneciente al nombre de espacios System.Windows.Forms, Combobox no tiene constructores.

ComboBox micombo = new ComboBox();

## Selección de Propiedades de ComboBox

1. **SelectedIndex**: Devuelve el valor entero de la posición que correspondería al elemento seleccionado de la lista. Si no se llegara a seleccionar ningún elemento el valor seria -1.
2. **SelectedItem**: A diferencia de SelectedIndex, esta propiedad devuelve el elemento en sí, que puede corresponder a una cadena, objeto, etc.
3. **DropDownStyle**: Permite elegir el estilo del ComboBox. Tiene tres valores posibles, que son **Simple** (sin menú desplegable, se verán todos los elementos dentro del mismo), **DropDown** (muestra un desplegable y nos da la opción de introducir texto) y **DropDownList** (muestra un desplegable para elegir nuestra opción).
4. **RightToLeft**: Booleano que indica si alinearemos nuestro texto a la izquierda o derecha.
5. **Sorted**: Booleano que indica si los elementos de la lista de nuestro ComboBox están ordenados alfabéticamente.
6. **AllowsDrop**: Indica si el control puede aceptar datos que el usuario arrastre. (*práctica Drag&Drop con sus eventos*).
7. **AutoCompleteMode**: Forma en la que queremos tener el autocompletado. Los valores son Suggest, Append y SuggestAppend.
8. **AutoCompleteSource**: Fuente desde donde cargaremos las sugerencias. Lo más aconsejable sería desde la propia lista del ComboBox. (*ListItems*).

## Selección de Métodos de ComboBox

Son archivos de texto.

1. **Añadir elementos** El usuario puede añadir elementos a la lista de las siguientes formas:

**1.- Items.Add():** Permite añadir un elemento al final de la lista. Acepta string, int, objetos, etc.

cbHorario.Items.Add("Nueva Zelanda");

**2.**- **Items.Insert():** A diferencia de Items.Add(), además de pedir el parámetro a introducir, pide antes su índice.

cbHorario.Items.Insert(0, "Sevilla");

1. **Quitar elementos**

El usuario puede eliminar contenido de su ComboBox de las siguientes maneras:

**1.- Items.Remove():** Eliminar un elemento concreto:

cbHorario.Items.Remove(cbHorario.Text.Trim());

**2.**- **Items.RemoveAt():** A diferencia de Items.Remove(), nos permite borrar un elemento que corresponda a un índice concreto.

cbHorario.Items.Insert(0, "Málaga");

cbHorario.Items.Insert(1, "DIW");

cbHorario.Items.Insert(2, "DWC");

cbHorario.Items.Insert(3, "DAW");

cbHorario.Items.RemoveAt(0);

**3.**- **Items.Clear():** Elimina todos los elementos de la lista.

1. **FindString()**. Este método tiene dos constructores:
   1. **FindString(String):** Devuelve el índice del primer elemento que comience por la cadena especificada.
   2. **FindString(String,Int32):** Devuelve el índice del primer elemento que comience por la cadena especificada a partir del índice indicado en el segundo parámetro.
2. **FindStringExact(String):** Al igual que el anterior, tiene dos constructores:
   1. **FindStringExact(String):** Busca el primer elemento de nuestro ComboBox que coincida exactamente con la cadena especificada.
   2. **FindStringExact(String,Int32):** Igual que el anterior, sólo que a partir del índice especificado en el segundo parámetro.

## Selección de Eventos de ComboBox

1. **SelectedIndexChange()**. Se produce cuando el usuario despliega el ComboBox y selecciona uno de los elementos (*ver práctica Dibujo*).
2. **DragEnter()**. Se da cuando se arrastra un elemento al área del control.
3. **DragDrop()**.Se produce cuando se completa la operación de arrastrar y soltar en el control.

## Observaciones a tener en cuenta sobre ComboBox

Al usar este control debemos tener en cuenta los siguientes puntos:

* Es un control muy utilizado en cualquier tipo de aplicación, ya que ofrece un sinfín de posibilidades.
* Según el estilo que le demos al menú desplegable, podremos introducir texto de una manera más fácil,

## Documentación de referencia

* <https://msdn.microsoft.com/es-es/library/system.windows.forms.combobox(v=vs.110).aspx>
* <http://csharp.net-informations.com/gui/cs-combobox.htm>