

3.2. wxStaticText y wxTextCtrl

wxStaticText

Este componente muestra una o más líneas de texto en modo sólo lectura.

Para crear este componente, debemos pasar al constructor, la ventana padre, el identificador, la etiqueta, la posición, el tamaño y el estilo.

```
wxStaticText(wxWindow* parent, wxWindowID id, const wxString& label,
             const wxPoint& pos = wxDefaultPosition,
             const wxSize& size = wxDefaultSize, long style = 0)
```

Los estilos que podemos usar en este componente son los siguientes: **wxALIGN_LEFT**, **wxALIGN_RIGHT**, **wxALIGN_CENTRE** y **wxALIGN_CENTER**, para centrar el texto en la etiqueta, y **wxST_NO_AUTORESIZE**, para que la etiqueta no se ajuste al texto y siempre mantenga su tamaño original (normalmente se ajusta al texto).

Al ser un componente tan simple, sólo tiene dos métodos destacables: *GetLabel()* y *SetLabel()*, que sirven para ver y modificar el contenido de la etiqueta.

wxTextCtrl

Este componente es parecido al anterior, pero en este caso podemos editar y escribir texto en él.

Para crearlo, debemos pasar al constructor, la ventana padre, el identificador, la etiqueta, la posición, el tamaño y el estilo, además del texto inicial que queremos que muestre.

```
wxTextCtrl(wxWindow* parent, wxWindowID id, const wxString& value = "",
           const wxPoint& pos = wxDefaultPosition,
           const wxSize& size = wxDefaultSize, long style = 0)
```

Existen multitud de estilos para este componente, a continuación veremos algunos: **wxTE_PROCESS_ENTER** y **wxTE_PROCESS_TAB** para controlar el cambio de un **wxTextCtrl** a otro con las teclas de enter y tabulación respectivamente, **wxTE_MULTILINE**, para que permita varias líneas de texto, **wxTE_PASSWORD**, para que muestre asteriscos en lugar de lo que escribimos, **wxTE_READONLY**, para que sea de sólo lectura, **wxHSCROLL**, para que muestre una barra horizontal, **wxTE_LEFT**, **wxTE_CENTRE**, **wxTE_RIGHT**, para la alineación del texto, y **wxTE_NO_VSCROLL** para eliminar la barra vertical que por defecto muestra.

Son tres eventos los que este componente genera, **EVT_TEXT**, cuando el texto cambia, **EVT_TEXT_ENTER**, cuando se pulsa la tecla enter, y **EVT_TEXT_MAXLEN**, cuando el texto introducido supera el límite indicado.

De los muchos métodos que tiene este componente destacaremos los siguientes: *AppendText()*, para introducir texto al final del existente (si lo hay), *WriteText()*, con el que se puede escribir en cualquier punto del mismo, *SetValue()* y *GetValue()*, que trabajan con el texto completo, *GetLineText()*, que utiliza líneas para mostrar el texto, *Copy()*, *Cut()* y *Paste()*, que trabajan con el portapapeles, *GetNumberOfLines()*, que devuelve el número de líneas, *Clear()*, para limpiar el texto, y *LoadFile()* y *SaveFile()*, para trabajar con ficheros.