

wxWidgets

Contents:

1. Hello, wxWidgets World!
2. wxFrames
3. wxSizer

Hello, World!

So far, we've been making CLI programs.

- A CLI, you should recall, is a **Command-Line Interface**.

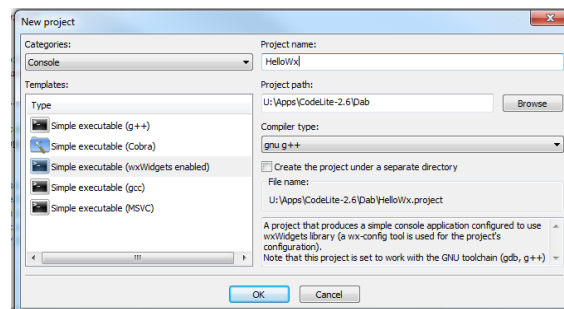
A **GUI** is a Graphical User Interface. Most programs that a typical user interfaces with are GUI-based.

wxWidgets is a popular cross-platform GUI development framework.

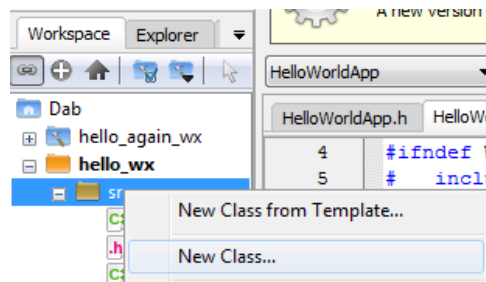
Head to the [Codelite download page](#) and grab the wxWidgets binaries.

Use this code to get started:

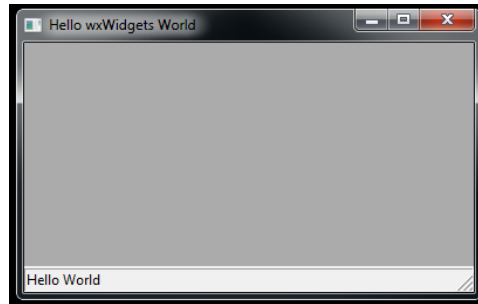
- http://wiki.wxwidgets.org/Hello_World
- Make a new project in Codelite, making sure to select (**wxWidgets enabled**)



- Remove the file **main.cpp** from the project.
- Right click the project's src folder and select **new class...**



- Copy the demo code to **HelloWorldApp.h** and **HelloWorldApp.cpp**
- Compile and run. You should see this:



Now check out the other tutorials:

- [http://wiki.wxwidgets.org/Guides %26 Tutorials](http://wiki.wxwidgets.org/Guides%26Tutorials)

wxFrame

A **wxFrame** is the main container for your GUI. A frame can have one or more of the following:

- menuBar
- toolBar
- statusBar

The user inherits from wxWidgets base classes, and adds functionality.

- See the tutorial: http://wiki.wxwidgets.org/Using_WxFrame
- The full TextFrame class:
 - http://wiki.wxwidgets.org/The_Full_Implementation_Of_The_TextFrame_Class
 - Copy this code into TextFrame.h and TextFrame.cpp files.

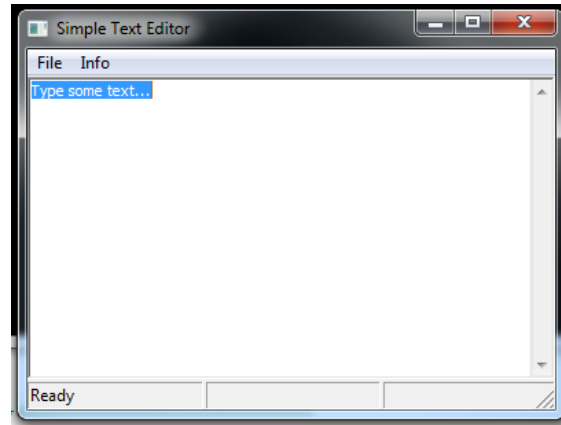
To add the frame to your application, you need to construct an object and set a few attributes.

This is done in the **OnInit** function of your App class (e.g. HelloWorldApp).

- Delete or comment out the previously existing code.
- Insert the code below

```
TextFrame *frame = new TextFrame(  
    wxT("Simple Text Editor"), 100, 100, 400, 300);  
frame->Show(TRUE);  
SetTopWindow(frame);  
return true;
```

- ❑ Compile and run your project. You should see this:



wxSizer

wxSizer

- used to specify layout.
- FYI: Sizer demos
 - <http://neume.sourceforge.net/sizedemo/>

Adding Pointers to Header

To demonstrate sizers, let's make another textCtrl.

- Go to the TextFrame class declaration in TextFrame.h
- Find this line under **private** declarations:

```
wxTextCtrl *m_pTextCtrl;
```

- Change the previous line to add another wxTextCtrl pointer
- Also add a wxSizer pointer:

```
wxTextCtrl *m_pTextCtrl1, *m_pTextCtrl2;  
wxBoxSizer *sizer;
```

Instantiating TextCtrl

- In the TextFrame constructor, find this statement:

```
m_pTextCtrl = new wxTextCtrl(this, -1, _T("Type some text..."),  
                             wxDefaultPosition, wxDefaultSize, wxTE_MULTILINE);
```

- The above statement instantiates a TextCtrl object.
- A control can be **static** or **interactive**.
- Interactive controls include:

- text boxes
 - checkboxes
 - drop-down menus
 - radio buttons
 - etc.
- Below the above statement, instantiate your new TextCtrl:

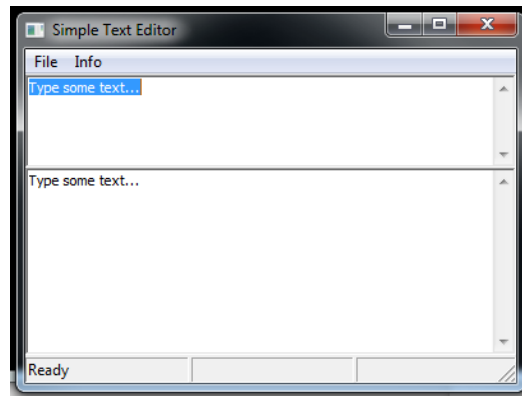
```
m_pTextCtrl2 = new wxTextCtrl(this, -1, _T("Type some text..."),  
                               wxDefaultPosition, wxDefaultSize, wxTE_MULTILINE);
```

Using wxSizer

- Immediately below the previous statements, create a new sizer and use it to layout the two text controls:

```
sizer = new wxBoxSizer(wxVERTICAL);  
  
sizer->Add( m_pTextCtrl1, 1, wxEXPAND, 0);  
sizer->Add( m_pTextCtrl2, 2, wxEXPAND, 0);  
  
SetSizer(sizer);
```

- You should see this:



- ★ Notice that the 2nd textCtrl is twice as tall as the 1st...