Creating a Simple GUI

You will want to go to wxPython's website (www.wxpython.org) and download a copy that matches your Python version. If you have a 32-bit Python, make sure you download a 32-bit wxPython. You cannot use easy_install or pip to install wxPython unless you get the bleeding edge Phoenix version of wxPython, so you'll have to grab a copy that is pre-built for your system either from the wxPython website or from your system's package manager. I recommend using at least wxPython 2.9 or higher.

Let's write some code!

```
import wx

class DemoPanel(wx.Panel):
    """"""

    def __init__(self, parent):
        """Constructor"""
        wx.Panel.__init__(self, parent)

        labels = ["Name", "Address", "City", "State", "Zip",
                  "Phone", "Email", "Notes"]

        mainSizer = wx.BoxSizer(wx.VERTICAL)
        lbl = wx.StaticText(self, label="Please enter your information here:")
        lbl.SetFont(wx.Font(12, wx.SWISS, wx.NORMAL, wx.BOLD))
        mainSizer.Add(lbl, 0, wx.ALL, 5)
        for lbl in labels:
            sizer = self.buildControls(lbl)
            mainSizer.Add(sizer, 1, wx.EXPAND)
        self.SetSizer(mainSizer)
        mainSizer.Layout()

    def buildControls(self, label):
        """
        Put the widgets together
        """
        sizer = wx.BoxSizer(wx.HORIZONTAL)
        size = (80,40)
        font = wx.Font(12, wx.SWISS, wx.NORMAL, wx.BOLD)

        lbl = wx.StaticText(self, label=label, size=size)
        lbl.SetFont(font)
        sizer.Add(lbl, 0, wx.ALL|wx.CENTER, 5)
```

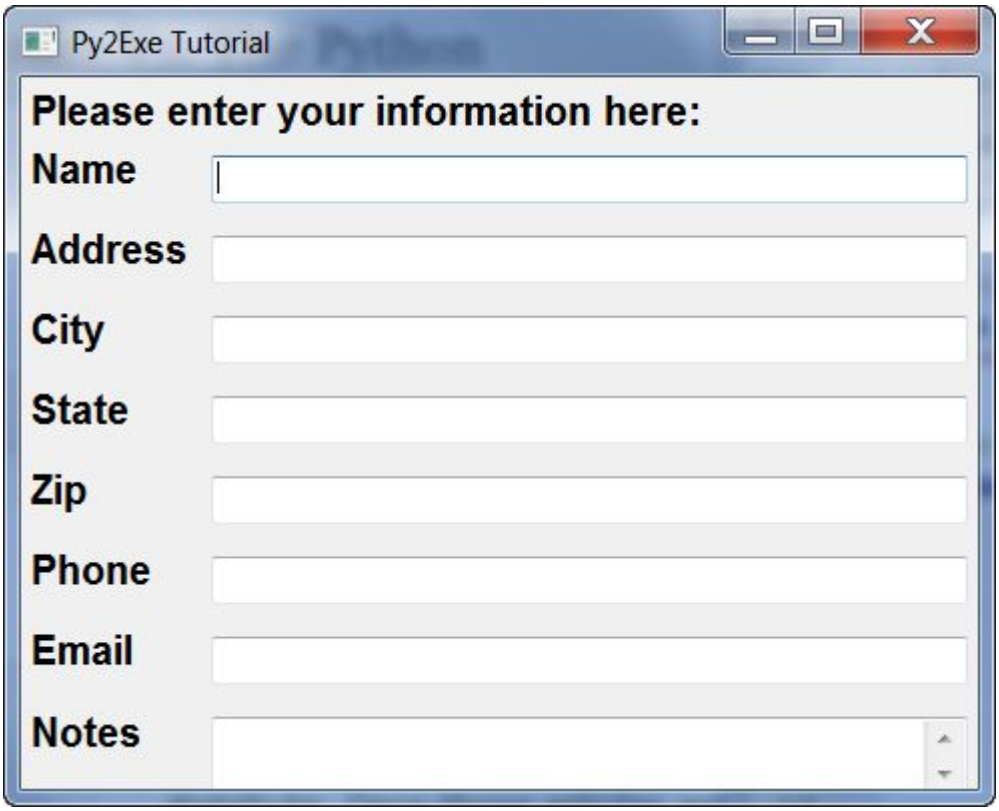
```
        if label != "Notes":
            txt = wx.TextCtrl(self, name=label)
        else:
            txt = wx.TextCtrl(self, style=wx.TE_MULTILINE, name=label)
        sizer.Add(txt, 1, wx.ALL, 5)
        return sizer

class DemoFrame(wx.Frame):
    """
    Frame that holds all other widgets
    """

    def __init__(self):
        """Constructor"""
        wx.Frame.__init__(self, None, wx.ID_ANY,
                          "Py2Exe Tutorial",
                          size=(600,400)
                          )
        panel = DemoPanel(self)
        self.Show()

if __name__ == "__main__":
    app = wx.App(False)
    frame = DemoFrame()
    app.MainLoop()
```

If you run the code above, you should see something like the following:



Let's break this down a bit. We create two classes, **DemoPanel** and **DemoFrame**. In wxPython, the **wx.Frame** object is what you use to create the actual "window" that you see in most cases. You add a **wx.Panel** to give your application the proper look and feel and to add tabbing between fields. The panel object's parent is the frame. The frame, being the top level widget, has

no parent. The panel contains all the other widgets in this example. We use sizers to help layout the widgets. Sizers allow the developer to create widgets that will resize appropriately when the window itself is resized. You can also place the widgets on the panel using absolute positioning, which is not recommended. We call the **MainLoop** method of the **wx.App** object at the end to start the event loop, which allows wxPython to respond to mouse and keyboard events (like clicking, typing, etc).

Now we're ready to learn how to package this application up into an executable!

Note: I tested on Windows 7 using Python 2.7.3, wxPython 2.9.4.0 (classic) and py2exe 0.6.9.