# 基本组件

## 按钮Button

## 标签Label

## 静态文本wx.StaticTex

### 效果：

****

### 源码：

***class StaticTextFrame(wx.Frame):***

***def \_\_init\_\_(self):***

***wx.Frame.\_\_init\_\_(self, None, -1, ’Static Text Example’,***

***size=(400, 300))***

***panel = wx.Panel(self, -1)***

***#*** 这是一个基本的静态文本

***wx.StaticText(panel, -1, ”This is an example of static text”,***

***(100, 10))***

***#*** 指定了前景色和背景色的静态文本

***rev = wx.StaticText(panel, -1, ”Static Text With Reversed Colors”,***

***(100, 30))***

***rev.SetForegroundColour(‘white’)***

***rev.SetBackgroundColour(‘black’)***

***#*** 指定居中对齐的的静态文本

***center = wx.StaticText(panel, -1, ”align center”, (100, 50),***

***(160, -1), wx.ALIGN\_CENTER)***

***center.SetForegroundColour(‘white’)***

***center.SetBackgroundColour(‘black’)***

***#*** 指定右对齐的静态文本

***right = wx.StaticText(panel, -1, ”align right”, (100, 70),***

***(160, -1), wx.ALIGN\_RIGHT)***

***right.SetForegroundColour(‘white’)***

***right.SetBackgroundColour(‘black’)***

***#*** 指定新字体的静态文本

***str = ”You can also change the font.”***

***text = wx.StaticText(panel, -1, str, (20, 100))***

***font = wx.Font(18, wx.DECORATIVE, wx.ITALIC, wx.NORMAL)***

176 / 565

***text.SetFont(font)***

***#*** 显示多行文本

***wx.StaticText(panel, -1, ”Your text\ncan be split\n”***

***”over multiple lines\n\neven blank ones”, (20,150))***

***#***显示对齐的多行文本

***wx.StaticText(panel, -1, ”Multi-line text\ncan also\n”***

***”be right aligned\n\neven with a blank”, (220,150),***

***style=wx.ALIGN\_RIGHT)***

***if \_\_name\_\_ == ’\_\_main\_\_’:***

***app = wx.PySimpleApp()***

***frame = StaticTextFrame()***

***frame.Show()***

***app.MainLoop()***

## 多行文本：

使用wx.TE\_MULTILINE样式标记创建一个多行文本控件。

### 源码：

***richText = wx.TextCtrl(panel, -1,***

***”If supported by the native control, this is reversed, and this is a different***

***font.”,***

***size=(200, 100), style=wx.TE\_MULTILINE|wx.TE\_RICH2) #***创建丰富

文本控件

***richText.SetInsertionPoint(0)***

***richText.SetStyle(44, 52, wx.TextAttr(“white”, ”black”)) #***设置文本样式

### 方法

* AppendText() 在控件内容后面添加文本

# 高级修饰布局类

## wx.BoxSizer

### 方法及参数说明

* **box = wx.BoxSizer(integer orient)**

参数**orient**是 wx.VERTICAL 或者 wx.HORIZONTAL

* **box.Add(wx.Window window, integer proportion=0, integer flag = 0, integer border = 0)**
* 参数**proportion**：参数定义了构件在既定方向上所占空间的比例，是相对的，相对于其他组件举个例子：我们有三个按钮，proportion的值分别是0，1，2。三个按钮被放置到水平方向的sizer当中，Button with proportion 0 will not change at all。但是， Button with proportion 2将会比the one with proportion 1大两倍在水平位置上。
* 参数flag：flag参数可以使用 '|'来产生组合的多个flags。flag参数定义了两个主要的行为：第一个参数是窗口的边框：这个参数决定了边框的宽度，在此决定窗口某一侧添加边框的事件。另一个参数决定了sizer事件的行为，当sizer改变时，空间的分配。并且分配的多少依赖于特定种类的sizer被使用。
* border：边框宽度。
  + **flag和borde**r参数两者结合起来使用flag=wx.EXPAND | wx.ALL,border=20---------这代表的是midPan这个组件将充分的利用空间，但是会从四周预留出来20px作为border。
* 可以在sizer分配给当前构件的空间之内的调整组件：

wx.ALIGN\_LEFT   
wx.ALIGN\_RIGHT   
wx.ALIGN\_TOP   
wx.ALIGN\_BOTTOM   
wx.ALIGN\_CENTER\_VERTICAL   
wx.ALIGN\_CENTER\_HORIZONTAL   
wx.ALIGN\_CENTER

**如：**

1. hbox5 = wx.BoxSizer(wx.HORIZONTAL)
2. btn1 = wx.Button(panel, -1, 'Ok', size=(70, 30))
3. hbox5.Add(btn1, 0)
4. btn2 = wx.Button(panel, -1, 'Close', size=(70, 30))
5. hbox5.Add(btn2, 0, wx.LEFT | wx.BOTTOM , 5)
6. vbox.Add(hbox5, 0, wx.ALIGN\_RIGHT | wx.RIGHT, 10)

 这个例子将已有的hbox5放置到vbox当总，而wx.ALIGN\_RIGHT所起到的作用就是在hbox5这个sizer当中的横向往右放置的调整

## static box sizer

类似于window编程的GroupBox. 一个static box sizer合并了box sizer和静态框（static box），静态框在sizer的周围提供了一个漂亮的边框和文本标签。效果如下图。

**效果：**

****

**源码：**

***import wx***

***from blockwindow import BlockWindow***

***labels = ”one two three four five six seven eight nine”.split()***

***class TestFrame(wx.Frame):***

***def \_\_init\_\_(self):***

***wx.Frame.\_\_init\_\_(self, None, -1, ”StaticBoxSizer Test”)***

***self.panel = wx.Panel(self)***

***# make three static boxes with windows positioned inside them***

***box1 = self.MakeStaticBoxSizer(“Box 1”, labels[0:3])***

***box2 = self.MakeStaticBoxSizer(“Box 2”, labels[3:6])***

***box3 = self.MakeStaticBoxSizer(“Box 3”, labels[6:9])***

***# We can also use a sizer to manage the placement of other***

***# sizers (and therefore the windows and sub-sizers that they***

***# manage as well.)***

***sizer = wx.BoxSizer(wx.HORIZONTAL)***

***sizer.Add(box1, 0, wx.ALL, 10)***

***sizer.Add(box2, 0, wx.ALL, 10)***

***sizer.Add(box3, 0, wx.ALL, 10)***

***self.panel.SetSizer(sizer)***

***sizer.Fit(self)***

***def MakeStaticBoxSizer(self, boxlabel, itemlabels):***

***# first the static box***

***box = wx.StaticBox(self.panel, -1, boxlabel)***

***# then the sizer***

***sizer = wx.StaticBoxSizer(box, wx.VERTICAL)***

***# then add items to it like normal***

***for label in itemlabels:***

***bw = BlockWindow(self.panel, label=label)***

***sizer.Add(bw, 0, wx.ALL, 2)***

***return sizer***

***app = wx.PySimpleApp()***

***TestFrame().Show()***

***app.MainLoop()***

# 类

## 拖拽