

Python 图形用户界面

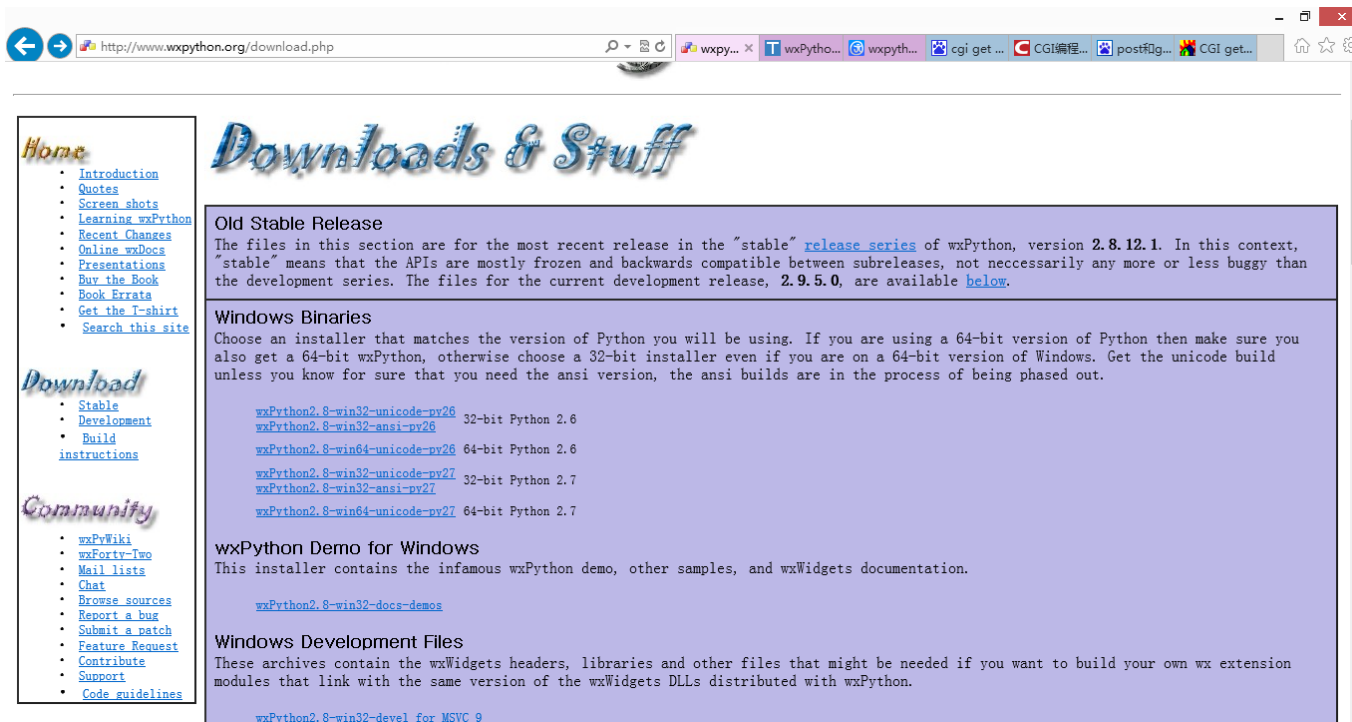
流行的图形用户界面

- Tkinter 使用 TK 平台，版标准化
- Wxpython 基于 wxwindows 跨平台
- Pythonwin 使用用于 windows
- Java swing 只用用于 jython
- pyGTK 使用 GTK 平台 流行于 linux
- Pyqt 使用 QT 平台 跨平台

Wxpython

- 下载网址

— <http://www.wxpython.org/download.php>



安装后测试

```
import wx  
app=wx.App()  
win=wx.Frame(None)  
win.Show()  
app.MainLoop()
```

wx.App 使用 wx.App 类创建一个应用程序对象

app.MainLoop() 进入主循环。主循环是一个不停的捕捉你应用程序发生的事件的无穷循环。它对任何 GUI 应用程序来说都是一个完整的部分

wx.Frame

- 创建一个框架构件。仅当调用一个构件的 `Show()` 方法时，它才会显现
 - `wx.Frame(parent, id=-1, title="", pos=wx.DefaultPosition, size=wx.DefaultSize, style=wx.DEFAULT_FRAME_STYLE, name="frame")`
 - `pos=(5,5)`
 - 距离左边 5 像素，距离上边界 5 像素

wx.Button

- 定义窗体按钮
 - `wx.Button(parent,id,label,pos, size=wxDefaultSize, style=0, validator, name="button")`
 - `wx.DefaultPosition`
 - `(-1,-1)`

wx.TextCtrl

- 文本输入框
 - `wx.TextCtrl(win,pos=(5,5),size=(210,25))`
 - `wx.TextCtrl(win,pos=(5,35),size=(390,260),style=wx.TE_MULTILINE|wx.HSCROLL)`
 - `style=wx.TE_MULTILINE`
 - 多行文本框，自动带有下拉条
 - `style=wx.HSCROLL`
 - 水平滚动条

exmaple

```
rt wx
app=wx.App()
win=wx.Frame(None,title="simple Editor",size=(410,335))
win.Show()
btn=wx.Button(win,label="open",pos=(225,5),size=(80,25))
btn=wx.Button(win,label="save",pos=(315,5),size=(80,25))

filename=wx.TextCtrl(win,pos=(5,5),size=(210,25))
contents=wx.TextCtrl(win,pos=(5,35),size=(390,260),style=wx.TE_MULTILINE|wx.HSCROLL)
app.MainLoop()
```


BoxSizer

- `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
- `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`
 - 定义宽度
- We do it with the following flags :
 - `wx.ALIGN_LEFT`
 - `wx.ALIGN_RIGHT`
 - `wx.ALIGN_TOP`
 - `wx.ALIGN_BOTTOM`
 - `wx.ALIGN_CENTER_VERTICAL`
 - `wx.ALIGN_CENTER_HORIZONTAL`
 - `wx.ALIGN_CENTER`

尺寸器布局

- Panel 是窗口的容器，通常其大小与 Frame 一样，在其上放置各种控件，这样可将窗口内容与工具栏及状态栏区分开，能过 TAB 键可遍历 Panel 中的元素
- `bkg=wx.Panel(win)`
- `hbox=wx.BoxSizer()`
- `vbox=wx.BoxSizer(wx.VERTICAL)`
- `bkg.SetSizer(vbox)`

事件

```
def load(event):
```

```
    file=input(filename.GetValue())
```

```
    contents.SetValue(file.read())
```

```
    file.close()
```

- Event 接受事件

- Filename 为文本框名字， filename.GetValue() 获取文本框传入的值

- Contents 为文本框的名字， contents.SetValue 将值放入文本框

事件绑定

- `loadbutton=wx.Button(bkg,label="open")`
- `loadbutton.Bind(wx.EVT_BUTTON,load)`
- `savebutton=wx.Button(bkg,label="save")`
- `savebutton.Bind(wx.EVT_BUTTON,save)`