thinter(Tk interface) 是python的标准GUI库，支持跨平台GUI程序开发，适合小型GUI程序编写。

wxPython是比较流行的GUI库，适合大型应用程序开发，整体设计框架类似于MFC(Microsoft Foundation Classes)微软基础类库。

PyQT

Qt是一种开源的GUI库，适合大型应用程序开发。PyQT是Qt工具包标准的Python实现。可以使用Qt Desginer界面设计器快速开发GUI应用程序。

<https://docs.python.org/3.7/library/tk.html>

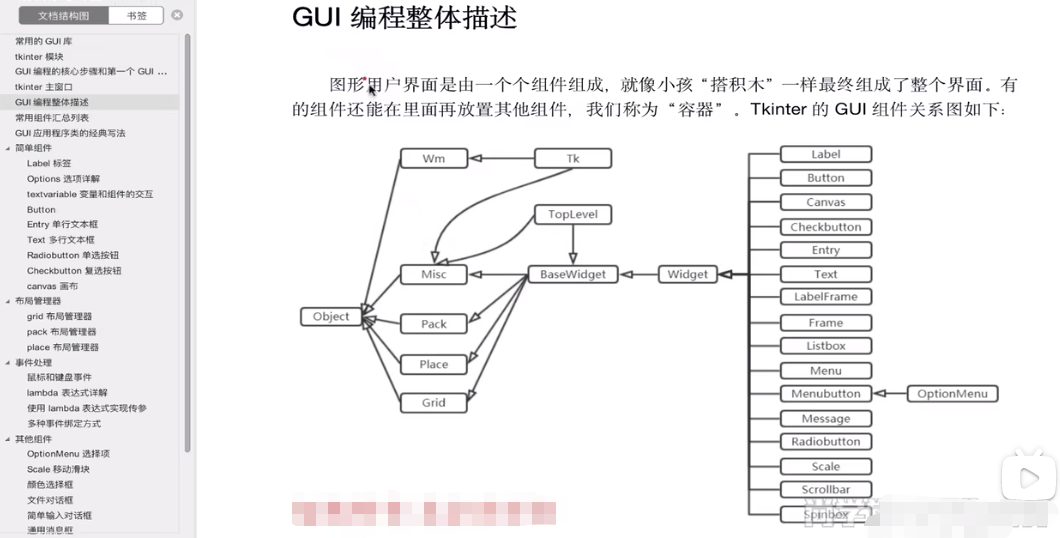
<http://effbot.org/tkinterbook>

PEP8编码规范

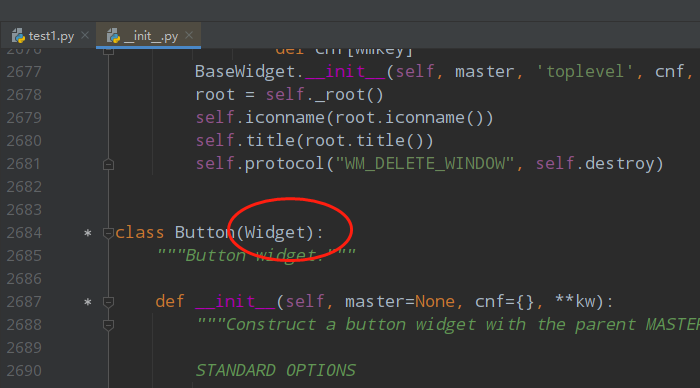
<https://www.python.org/dev/peps/pep-0008>

■thinter库

●thinter组件类的继承关系图



在pycharm中，可以按住Ctrl加组件名，来查看继承关系。例如下图，Button继承了Widget



●

常用组件汇总表



●第一个GUI程序

from tkinter import \*  
from tkinter import messagebox  
  
  
root = Tk()  
  
root.title("我的第一个GUI程序")  
# geometry(窗口宽度x窗口高度+左侧距离+上方距离) -右侧距离-下方距离  
root.geometry("500x300+200+100")  
  
btn01 = Button(root)  
btn01["text"] = "点我就送花"  
  
btn01.pack()  
  
  
def songhua(e): # e是事件对象  
 messagebox.showinfo("Message", "送你一朵小红花")  
 print("后台打印，送了一朵花")  
  
  
btn01.bind("<Button-1>", songhua) # 给btn01绑定一个鼠标点击事件  
  
root.mainloop() # 调用组件的mainloop方法，进入事件循环

●官方给的经典写法，使用面向对象的方式



from tkinter import \*  
from tkinter import messagebox  
  
  
class Application(Frame):  
 *"""一个经典的GUI程序的类的写法"""* def \_\_init\_\_(self, master=None):  
 super().\_\_init\_\_(master) # super调用父的初始化构造方法  
 self.master = master  
 self.pack()  
  
 self.createWidget()  
  
 def createWidget(self):  
 *"""在这里创建所有组件"""* # 创建一个按钮  
 self.btn01 = Button(self)  
 self.btn01["text"] = "点击送花"  
 self.btn01.pack()  
 self.btn01["command"] = self.songhua  
  
 # 再创建一个按钮  
 self.btn02 = Button(self, text="点击打招呼", command=self.sayhi)  
 self.btn02.pack()  
  
 # 创建一个退出按钮  
 self.btnQuit = Button(self, text="推出", command=root.destroy)  
 self.btnQuit.pack()  
  
 def songhua(self):  
 messagebox.showinfo("送花", "送你一朵小红花")  
 print("后台打印，送一朵花")  
  
 def sayhi(self):  
 messagebox.showinfo("打招呼", "你好哈！")  
 print("后台打印，你好")  
  
  
root = Tk()  
root.geometry("400x200+200+100")  
root.title("一个经典的GUI程序类的测试")  
app = Application(master=root)  
  
root.mainloop()