# 导入easyGui

* 为了使用easygui,你必须将其导入。最简单的导入声明是:
* import easygui

如果你使用这种形式的导入，那么你使用easygui的函数的时候，必须在函数的前面加上前缀easygui，像这样：

* easygui.msgbox(...)
* 另外一种选择是这样导入easygui的包：
* from easygui import \*

这使得我们更容易调用easygui的函数；你不需要再在函数的前面加上easygui的前缀，你可以这样编写代码：

* + msgbox(...)
* 第三种选择是按照以下的导入声明去使用：

import easygui as g 这可以让你保持easygui命名空间使用最短的输入。 你可以这样访问easgui函数,例如:

g.msgbox(...)

# MsgBox

* Msgbox用来显示消息,并给出了OK按钮。你可以发送任何你想要的消息,带着任何你要的标题。你甚至可以覆盖默认文本在“ok”按钮如果你愿意的话。以下是msgbox的示例函数：

def msgbox(msg="(Your message goes here)", title="", ok\_button="OK"):

…

* 最简洁的方法去覆盖它的按钮文本是使用一个关键参数,像这样：

输入：

msgbox("Backup complete!", ok\_button="Good job!")

显示：



点击 Good job!显示：



# CCBox

* ccbox提供了一个选择:继续和取消,并返回要么1(继续)或0(取消)。注意:这个ccbox返回整数值(1和0),不是真正的布尔值(true或false).

msg = "Do you want to continue?"

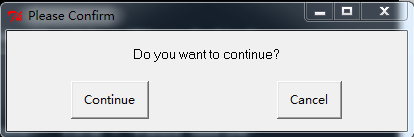
title = "Please Confirm"

if ccbox(msg, title): # show a Continue/Cancel dialog

pass # user chose Continue

else: # user chose Cancel

sys.exit(0)



# YNBox

* ynbox提供了一个选择是或者不是，并返回要么1(是的)或0(没有)。 注意:这个ynbox返回整数值(1和0),不是真正的布尔值(true或false)

# 自定义Box

* 为了指定自己的一套按钮在buttonbox里可以使用buttonbox()函数,这buttonbox可被用来显示一组你能选择的按钮。当用户点击一个按钮时,buttonbox(返回选择的文本）。如果用户取消或关闭buttonbox,默认选择(第一选择)将被返回。

## buttonbox

Buttonbox显示一个消息，一个标题和一组按钮。并且返回用户所选择的按钮的文本。

## indexbox

Indexbox显示一个消息,一个标题,和一组按钮。返回用户所选择的相应元素的索引。例如,如果你调用indexbox函数并且有三种选择(A,B,C),则A返回0，B返回1，C返回2

## boolbox

Boolbox (boolean box) 显示一个消息,一个标题,和一组按钮。它所返回的如果是1则第一个按钮被选中。否则返回0。

这里有个关于boolbox的简单例子：

message = "What does she say?"

title = ""

if boolbox(message, title, ["She loves me", "She loves me not"]): sendher("Flowers")

else:

pass

## 自定义图片

当你调用buttonbox的函数时，或者其他显示一个按钮的函数时，像msgbox, indexbox, ynbox等等，你可以指定关键字参数image=xxx，xxx是文件名，而且是gif格式的图像。注意目前只支持gif格式的图片。 如果图片的参数被指定了，那么图片将会显示在信息上面。 这里是一些简单的代码在easygui的演示程序上：