

# 1 介绍

作为一个 AI 产品，一个用户能够互动的窗口是必须的。像 Word，Excel 那样，它承载了我们与计算机互动的界面，使用这个窗口程序，能够让不懂计算机技术的普通人也能轻易使用你做出来的技术。

这次我们会讲解 Python 中常用的窗口制作方法，以一个简单的例子入门。

下面是一个最简单的例子，我们使用 `tkinter` 就能在电脑中创建一个窗口程序啦。

```
import tkinter as tk

window = tk.Tk()

# ... all window parts

window.mainloop()
```

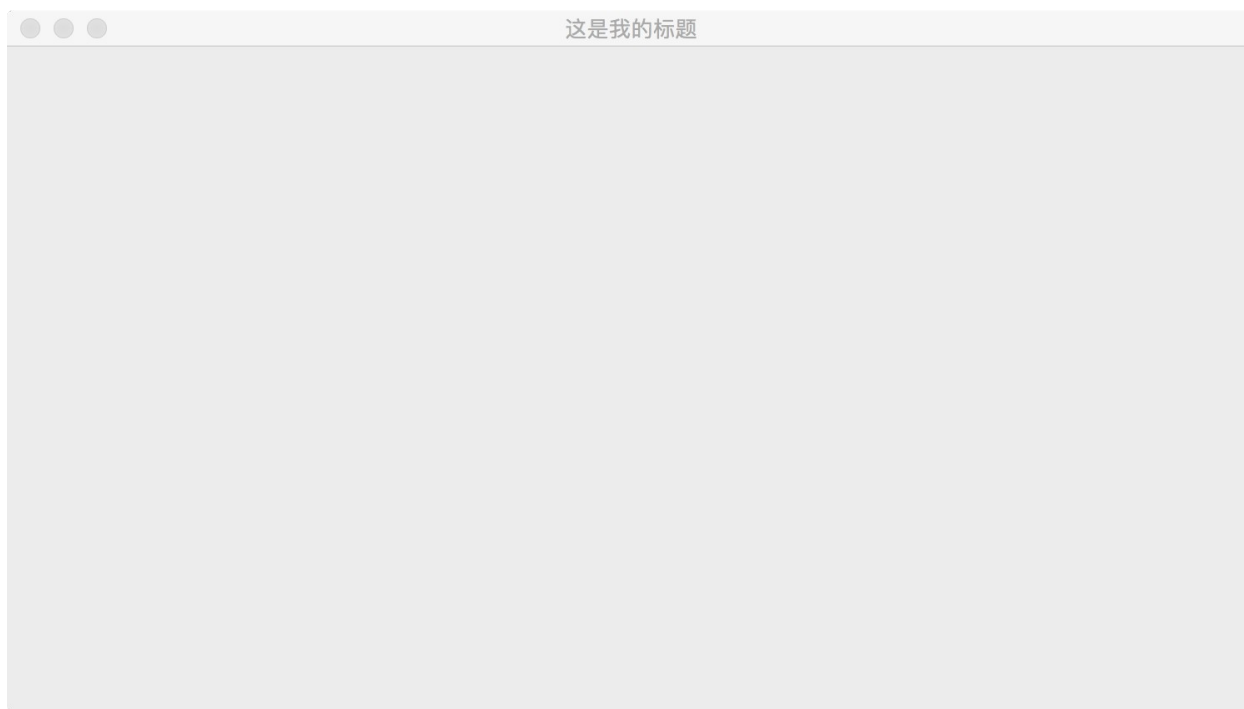
## 2 窗口规格

我们可以为窗口制定自己的标题和窗口大小。只需要调用 `window` 的 `title` 和 `geometry` 功能即可。

```
window.title('这是我的标题')
window.geometry('750x400')
```

运行结果：

```
..
```



basic window

## 3 窗口输入框

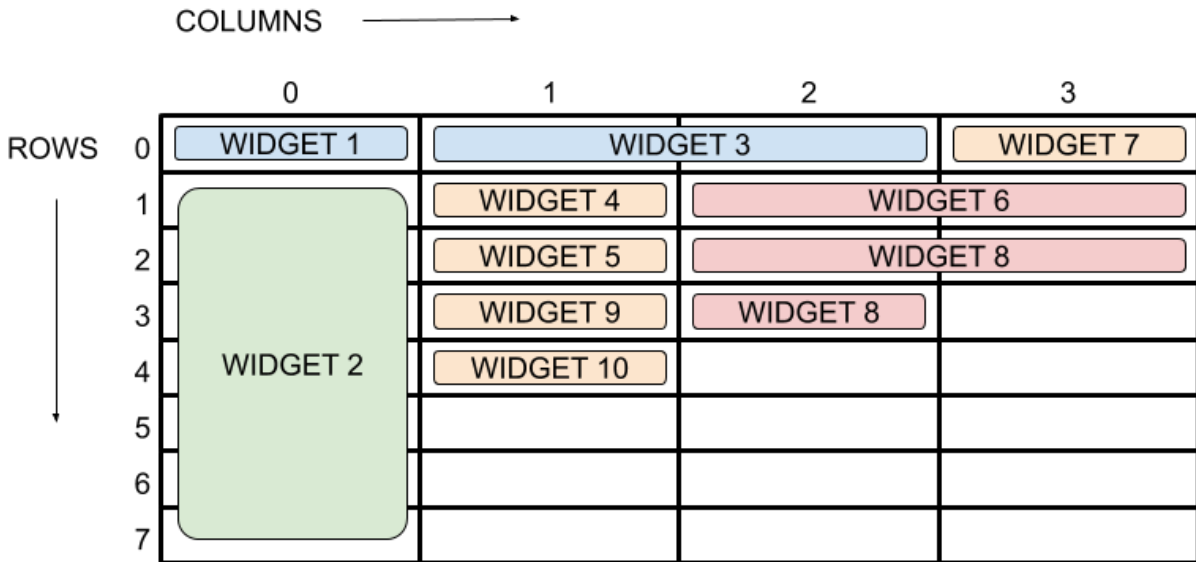
在接下来的项目中，我们会需要一个地方输入用户的信息，程序将会使用到这些用户的信息进行处理。在实际生活中，这就好比我们的百度搜索框。在框中输入内容后，百度会处理你的内容，并返回结果。

那么这中输入框可以按照下面的方式定义。

```
e = tk.Entry(window, width=80)
e.grid(row=0, column=0, columnspan=2)
```

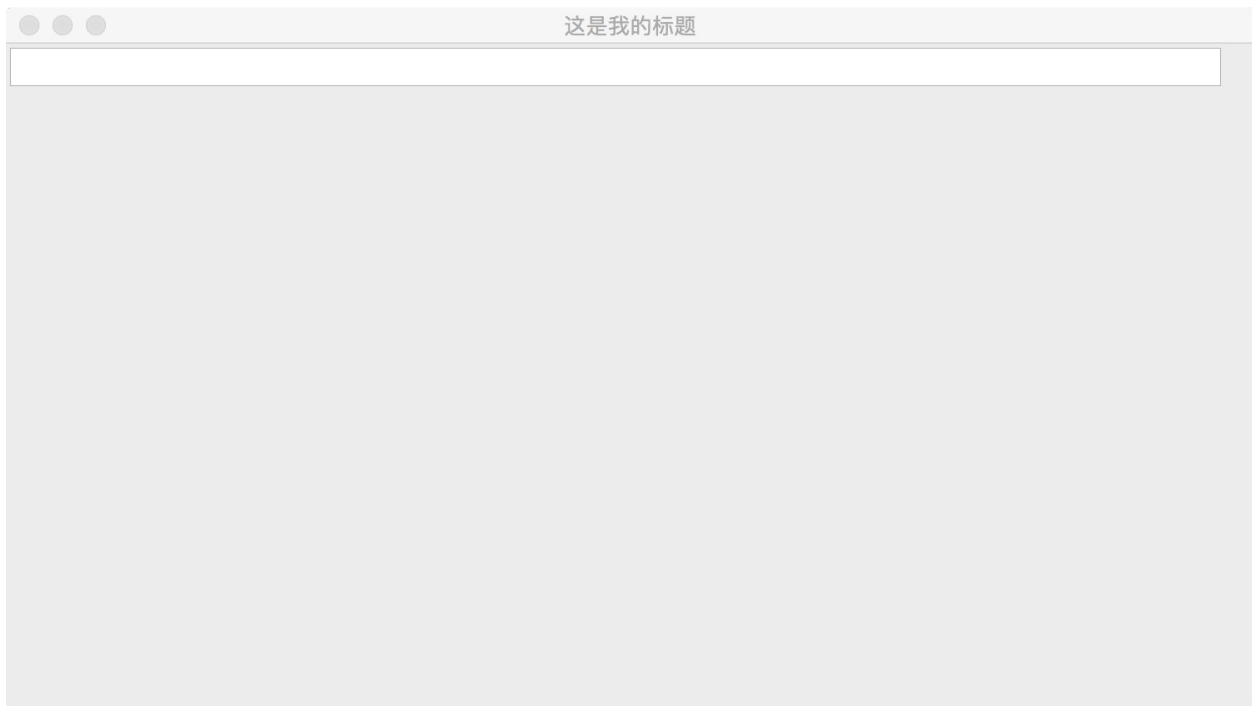
### 3.1 网格概念

这里我们涉及到了部件的位置，也就是说我们要在窗口的什么地方摆放这些部件。在定义好这个输入框 `Entry` 后，我们使用网格方案（`grid`）来放置输入框。



## grid

从图中可以看出，`grid` 中的参数定义了我放在窗口的哪一行哪一列，行列的跨度又是多少。但是 `tkinter` 还提供了几种其他的定义方式，不过这种方式定位在我们的项目中比较方便。

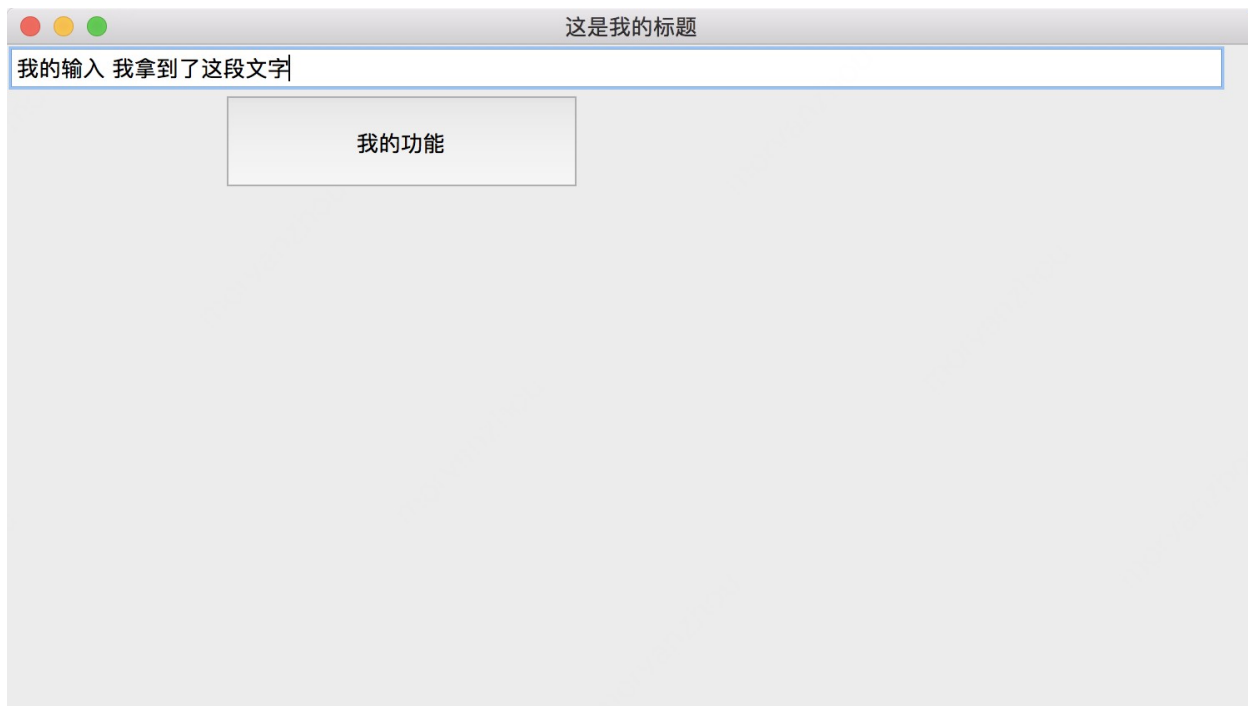


## basic entry

## 4 按钮

仔细观察百度的搜索界面，光是拥有一个输入框，我们也只是能输入文字而已，我们还需要一个按钮来触发这段文字的处理过程。换成百度搜索框，就是搜索框旁边的那个按钮啦。

在 `tkinter` 中自然也会有按钮这种组件。



basic button

```
def my_func():  
    e.insert("end", " 我拿到了这段文字")  
  
b = tk.Button(window, text='我的功能', width=20, height=3,  
               command=my_func)  
b.grid(row=1, column=0)
```

"Button" 的几个参数分别定义了按钮上的文字信息（`text`），按钮的宽高（`width`，`height`），和点击按钮时需要触发什么样的功能（`my_func`）。同时，我们也要在外面先定义好这样的触发功能将要完成什么任务。在"`my_func`"中，我们的任务就是给输入框中的文字后面加上一段话"我拿到了这段文字"。当然你也可以制作你自己的一些个性化功能。

## 5 列表

下面还有一个常用的功能，是为了展示出来一个个的内容列表的。类似于百度搜索后，给你展示的一条条搜索结果。



basic list

```
lb = tk.Listbox(window, width=40)
lb.grid(row=2, column=0)
```

为了添加一个实用的功能，我再创建一个按钮，来触发列表内容的显示。将输入框中的内容转移到列表中。

```
def move():
    var = e.get()
    lb.insert("end", var)
b = tk.Button(window, text='转移内容', width=20, height=3,
command=move)
b.grid(row=1, column=1)
```

## 6 图片

在窗口中，还可以插入你的个性化图片哦。



## basic image

```
canvas = tk.Canvas(window, height=200, width=500)
image_file = tk.PhotoImage(file="data/welcome.gif")
image = canvas.create_image(0, 0, anchor='nw', image=image_file)
canvas.grid(row=3, column=0, columnspan=2)
```

在创建图片的时候，我们还要给图片一张画布。如果你学过画画，在画画时你需要一个平面来放画纸，这个平面可以是桌子，可以是画架等。这些平面就是我们这里的画布 `canvas` 了。我们将图片读取到 `image_file` 中，然后将 `image_file` 放置到画布里 `canvas.create_image`。最后再将画布放到窗口中。

而且还能替换图片。

```
image_file = tk.PhotoImage(file="data/face_happy1.png")
canvas.itemconfig(image, image=image_file)
```



change pic