Tkinter

Created By	© Chris Tang
Last Edited	@Oct 11, 2019 2:23 PM
Notes	

Label & Button

```
#create a new window and set information
window = tkinter.TK()
window.title('_TitleName_')
window.geometry('200x100')

var = tk.StringVar()
#create "lable"
l = tkinter.Lable(window,textvariabler=var,bg='green',font=('Arial',12),width=15,
    height=2)  # unit of width and height: No. character
l.pack() #Place lable-pack

on_hit = False
#establish button
def hit_me:
  global on_hit
  if on_hit == False:
    on_hit = True
    var.set('you hit me')
  else:
    on_hit = False
    vat.set('')
b = tk.Button(window, text='hit me',width=15,height=2,command=hit_me)

#start running the window
window.mainloop()
```

Entry & Text

```
e = tk.Entry(window,show=None) #to enter the passpord, use show='*'
e.pack(window)

def insert_point():
  var = e.get()
  t.insert('insert',var)
def insert_end():
  var = e.get()
  t.insert('end',var)
```

```
b1 = tk.Button(window,text='insert point',width=15,height=2,
    command=insert_point)
b1.pack()
b2 = tk.Button(window,text='insert end',command=insert_end)
b2.pack()

t = tk.Text(window,height=2)
t.pack()

t.insert(1.1,var)
```

entry & text are different : text doesn't have the boundary

Listbox

```
def print_selection():
  value = lb.get(lb.curselection())
  var1.set(value)
#define the variable
var1 = tk.StringVar()

l = tk.Lable(window,bg='yellow',width=4, textvariable=var1)
l.pack()


b = tk.Button(window,text='print selection',command=print_selection)
b.pack()

var2 = tk.StringVar()
var2.set((A,B,C,D))
lb = tk.Listbox(window,listvariable=var2)
lb.pack()

#insert the list
list_items = [1,2,3,4]
for item in list_items:
  lb.insert('end',item)
lb.insert(1,'first')
lb.insert(2,'second')
lb.delete(2) #delete the element at the specific place
lb.pack()
```

Radio-button

```
l = tk.Lable(window,text='None',bg='yellow')

def print_selection():
  l.config(text='You have selected' + var.get())
```

```
r1 = tk.Radiobutton(window,text='Option A'
      variable=var,value='A',command=print_selection)
r1.pack()
r2 = tk.Radiobutton(window,text='Option B'
      variable=var,value='B',command=print_selection)
r2.pack()
r3 = tk.Radiobutton(window,text='Option C'
      variable=var,value='C',command=print_selection)
r3.pack()
```

Scale

```
l = tk.Lable(window,text='you have selected')

def print_selection(v):
  l.config(text='you have selected'+v)

s = tk.Scale(window,lable='try me',from_=5,to=11,
      orient=tk.HORIZONTAL,length=200, showvalue=0
      tickinterval=3,resolution=0.01,command=print_selection)
    # unit (length):Pixel, showvalue = True/False
s.pack()
```

Check-button

```
var1 = tk.IntVar()
var2 = tk.IntVar()

def print_selection():
  if (var1.get() == 1)&(var2.get() == 1):
    l.config(text='I love both!')
...

c1 = tk.Checkbutton(window,text='Python', variable=var1, onvalue=1,offvalue=0
      command=print_selection)
c2 = tk.Checkbutton(window,text='C++',variable=var2, onvalue=1,offvalue=0
      command=print_selection)
c1.pack()
c2.pack()
```

radio-button:only one value can be chosen

check-button: multiple value can be chosen

Canvas

```
canvas = tk.Canvas(window,bg='blue', height=100,width=200) #height,pixel
image_file = tk.PhotoImage(file='ins.gif')
# image_file
image = canvas.create_image(10,10,anchor='nw', image=image_file)
  #anchor：选取坐标轴原点
# create the shape by the user
x0,y0,x1,y1=50,50,80,80
line = canvas.create_line(x0,y0,x1,y1)
oval = canvas.creat_oval(x0,y0,x1,y1,fill='red')
arc = canvas.create_arc(x0+30,y0+30,x1+30,y1+30,start=0,extent=180)
rec = canvas.create_rectangle(100,30,100+20,30+20)

canvas.pack()

def moveit():
  canvas.move(rect,0,2)

b = tk.Button(window,text='move',command=moveit).pack()
```

Menubar

```
count = 0

menubar = tk.Menu(window)
filemune = tk.Menu(menubar,tearoff=0)
  #tearoff: whether they could be split
menubar.add_cascade(lable='File',menu=filemenu)
def do_job():
  global count
  l.config(text='do'+str(count))
  count+=1
filemune.add_command(lable='New',command=do_job)
filemune.add_command(lable='Save',command=do_job)
filemenu.add_seperator()
filemenu.add_command(label='Exit',command=window.quit)

editmenu = tk.Menu(menubar,tearoff=0)
menubar.add_cascade(lable='Edit',menu=editmenu)

# submenu
submenu = tk.Menu(filemenu)
filemenu.add_cascade(lable='Import',menu=submenu)
submenu.add_command(lable='Submenu',command=do_job)


window.config(menu=menubar)
```

window.config: change parameters of the object `window`

Frame

```
frm = tk.Frame(window)
frm.pack()
frm_l = tk.Frame(frm)
frm_r = tk.Frame(frm)
frm_l.pack(side='left')
frm_r.pack(side='right')
```

frame: sub-window