

«به نام خداوند بخشنده و مهربان»



آموزش Tkinter به زبان ساده

نویسنده : سید محمدرضا حسینیان راوندی

وبسایت: [www.pachenar.ir](http://www.pachenar.ir)

وبلاگ: [www.barnamenevisi.blogspot.com](http://www.barnamenevisi.blogspot.com)

رایانامه: [ravand@chmail.ir](mailto:ravand@chmail.ir)

تاریخ: ۱۳۹۲/۱/۱

## مقدمه

ابتدا یاد می‌کنم از شهدای انقلاب اسلامی ایران و شهدای هشت سال دفاع مقدس و همچنین یاد می‌کنم از رهبر کبیر انقلاب اسلامی امام خمینی (ره). این کتاب رو و این تلاش رو تقدیم میکنم به روح پاک آن بزرگواران.

در این کتاب به آموزش tkinter می‌پردازیم برای یادگیری tkinter شما باید قبلاً python کار کرده باشید. tkinter کتابخانه‌ی واسط کاربر گرافیکی برای زبان برنامه نویسی پایتون است.

<http://fa.wikipedia.org/wiki/تکینتر>

<http://wiki.python.org/moin/TkInter>

## واسط گرافیکی:

واسط گرافیکی چیست؟

[http://fa.wikipedia.org/wiki/واسط\\_گرافیکی\\_کاربر](http://fa.wikipedia.org/wiki/واسط_گرافیکی_کاربر)

برای پایتون واسط گرافیکی کاربر بسیاری نوشته شده است پرکاربردترین آنها به شرح زیر می‌باشد:

tkinter (به صورت پیش‌فرض همراه با نسخه‌های استاندارد پایتون ارائه می‌شود و یک رابط شی‌گرا برای ابزار Tcl/Tk در محیط پایتون فراهم می‌کند).

PyGTK

wxPython

pyFLTK

FXpy

PyOpenGL

**پایتون:** یک زبان برنامه نویسی است که هم‌میشه برنامه‌ی تحت وب باهاش نوشت و هم تحت دسکتاپ و حتی همیشه با اون برنامه‌ی ای نوشت که روی گوشی‌های هوشمند هم اجرا بشه.

[http://fa.wikipedia.org/wiki/پایتون\\_\(زبان\\_برنامه‌نویسی\)](http://fa.wikipedia.org/wiki/پایتون_(زبان_برنامه‌نویسی))

شما می‌توانید با استفاده از tkinter یک برنامه‌ی تحت دسکتاپ بنویسید .  
تحت دسکتاپ چیست؟

برنامه‌هایی که به صورت تحت دسکتاپ نوشته میشوند بر عکس برنامه‌هایی که به صورت تحت وب نوشته می‌شوند روی خود سیستم کاربر اجرا و نصب می‌شوند ولی برنامه‌های تحت وب روی سرور سایت نصب و اجرا می‌شوند. مثلاً برنامه‌هایی که با نسخه‌ی تحت وب php نوشته می‌شوند روی سرور سایت نصب و اجرا میشوند ولی برنامه‌ی ای مثل Adobe Acrobat Reader که باهاش فایل‌های pdf را توی سیستم مشاهده می‌کنید تحت دسکتاپ هستند. پایتون یک زبان platform است. پلت فرم یعنی اینکه برنامه‌هایی که شما می‌نویسید روی همه‌ی سیستم‌عامل‌ها نصب و اجرا می‌شوند.

پایتون یک زبان منبع باز است (open source) منبع باز یعنی اینکه همیشه راحت به سورس(منبع) یا همون کدهای برنامه دسترسی پیدا کرد برنامه‌های سورس باز رو همیشه با برنامه‌ی

notepad ویندوز هم نوشت. البته با برخی برنامه ها میان سورس رو به صورت رمزگذاری شده در میان.

خصوصیت دیگر پایتون سادگی آن است اگر مبتدی باشید و بخواهید با جاوا یا C# کار کنید مطمئناً برایتان مشکل خواهد بود، زبانی مثل جاوا کاملاً شی گزایی هست ولی پایتون هم روبه گراست (به صورت ساده نوشته میشه) و هم شی گراست. و همیشه آن را گسترش هم داد. سرعت: برنامه های پایتون با سرعت بیشتری نسبت به خیلی از برنامه ها اجرا می شوند.

ابزار های مورد نیاز برای کار با tkinter :

ما چون با پایتون ۲,۳,۰ کار می کنیم باید اول پایتون رو از این آدرس دانلود کنیم:

<http://www.python.org/download/>

حالا شما به یک نرم افزار ویرایشگر برای نوشتن برنامه های خود نیاز دارید که این برنامه همراه خود نرم افزار پایتون هست . وقتی پایتون رو نصب کردید به All program رفته و سپس Python 3.3 و بعد از آن IDLE(Python GUI) را انتخاب کنید.

برای کار با پایتون در این صفحه کلیک کنید مثلاً برای چاپ رشته دستور زیر را وارد کنید:

```
print("ravandi")
```

```
out: ravandi
```

برای اینکه راحت تر با این برنامه کار کنید به منوی file و سپس new رفته و گزینه ی New window را کلیک کنید تا پنجره باز شود . و سپس کدهای خود را وارد کنید برای اجرای برنامه ی خود می توانید فایل های خود را هم با پسوند .py و هم .pyw. در هر جای ویندوز که خواستید ذخیره کنید. آگه با pyw ذخیره کنید دیگه برنامه ی command prompt نمایش داده نمیشه. بهتره شما از همین ویرایشگر پایتون استفاده کنید چون قابلیت این را دارد که خطا ها رو هم به شما اطلاع دهد.

یک برنامه ی ساده با tkinter :

برنامه ای برای نمایش متن روی برجسب:

```
from tkinter import*
# ایجاد پنجره
barname = Tk()
# اندازه ی پنجره
barname.geometry("150x100")
# عنوان برنامه
barname.title('عنوان پنجره')
# ایجاد برجسب
w = Label(barname, text="راوندي")
# نمایش برجسب
w.pack()
mainloop()
```

در خط اول کتابخانه ی tkinter را وارد کردیم  
برای توضیحات یک خطی ما از # استفاده کردیم.  
برای توضیحات چند خطی هم از روش زیر استفاده کنید:

```
"""
```

```
متن چند خطي
```

```
خط دوم
```

```
خط سوم
```

```
خط چهارم
```

```
"""
```

اگر برنامه mainloop() نداشته باشد با دابل کلیک روی فایل برنامه ، برنامه اجرا نخواهد شد.

نکته اینکه در نسخه های دیگر پایتون tkinter با حرف T بزرگ نوشته می شود ولی در نسخه ی 3.3.0 با حرف t کوچک نوشته می شود. برنامه ی بالا را با مثلا نام test.pyw در هر کجای ویندوز ذخیره کنید .و سپس برای اجرای برنامه دکمه ی F5 را فشار دهید در این صورت برنامه از شما میخواهد که آیا میخواهید ذخیره کنید با زدن ok پنجره ی برنامه ی شما باز می شود:



برای اجرای برنامه هم می توانید راحت روی فایل test.pyw دابل کلیک کنید. تا اجرا شود. اگر حروف بهم ریخته مشاهده کردید در همین پنجره ی test.pyw به منوی options رفته و زیر منوی Configure IDE... را انتخاب کنید. تب General را انتخاب کنید و در قسمت Default Source Encoding گزینه ی UTF-8 را انتخاب کنید و ok کنید.  
**دادن فونت و رنگ و اندازه به متن:**

```
from tkinter import *
root = Tk()
Label(root,text="سید محمدرضا حسینیان راوندي",fg = "light green",bg = "darkgreen",
font = "tahoma 16 bold italic").pack()
root.mainloop()
```

fg : مخفف foreground رنگ نوشته  
bg : مخفف background رنگ زمینه ی متن.

tahoma 16 bold italic

نوع فونت تاهوما اندازه ۱۶ ضخیم و کج

برای رفتن قسمتی از متن به خط بعد به این روش عمل کنید:

```
text="خط بعد\nمتن"
```

**تعیین محدودیت برای متن:**

```
from tkinter import*
root = Tk()
T = Text(root, height=2, width=30)
T.pack()
T.insert(END, "متن خط اول\nمتن خط دوم")
mainloop()
```

یک صفحه برای متن ایجاد کردیم با text با ارتفاع دو(دو خط) با عرض ۳۰ (۳۰ کاراکتر) با insert مقادیر را که شامل متن متن خط اول\n متن خط دوم\n هست را وارد کردیم.

**رنگ زمینه ی پنجره:**

```
import tkinter
from tkinter import *
root = Tk()
root.geometry("300x200")
root.title("عنوان")
root.configure(background='black')
root.mainloop()
```

## روش دیگر:

```
import tkinter
from tkinter import *
root = Tk()
root.geometry("300x200")
root.title("عنوان")
root['background'] = 'green'
root.mainloop()
```

## نمایش تصویر:

```
from tkinter import *
root = Tk()
logo = PhotoImage(file="ravandi.gif")
explanation = "متن روی عکس"
w = Label(root,compound = CENTER,text=explanation,image=logo).pack(side="right")
root.mainloop()
```

Side="left" را به سمت چپ می برد.  
compound = CENTER : موقعیت متن روی عکس  
فقط فرمت gif و PPM/PGM رو پشتیبانی میکند.

اگر عکس در کنار فایل برنامه نباشد و در درایو دیگری باشد به این شکل آدرس دهی کنید:  
C:/ravandi/ravand.gif

## ایجاد عکس و متن:

```
from tkinter import *
root = Tk()
logo = PhotoImage(file="ravandi.gif")
w1 = Label(root, image=logo).pack(side="right")
explanation = "این متن تصویر است"
w2 = Label(root,justify=LEFT,padx = 10,text=explanation).pack(side="left")
root.mainloop()
```



اگر فایل ما که کدهای بالا در آن است در پوشه ی python باشد. و عکس ما در پوشه ای دیگر در همان درایو با نام مثلاً image باشد روش آدرس دهی ما میتواند اینطوری باشد:

../image/ravandi.gif

یا اینطوری:

F:/image/ravandi.gif

نکته اگر هنگام کار با ویرایشگر تان دکمه های صفحه کلید مثلا ctrl و v کار نکرد از گزینه هایی که در منوهای ویرایشگر هست استفاده کنید.

## ایجاد برجسب و تکست:

```
from tkinter import *
top = Tk()
Label(top, text="نام شما:").grid(row=0, column=1)
e = Entry(top)
e.grid(row=0, column=0)
top.mainloop()
```

column برای ایجاد ستون و row هم برای ایجاد ردیف بکار میره. یعنی برجسب ما در ردیف اول و ستون دوم هست. و ورودی یا تکست هم در ردیف اول و ستون اول است. اینجا مقدار از صفر شروع می شود.

## ایجاد دکمه:

```
from tkinter import*
root = Tk()
root.title("عنوان")
button =Button(root, text='خروج', width=25, command=root.destroy)
button.pack()
root.mainloop()
```

با کلیک روی دکمه با کمک دستور زیر پنجره بسته می شود:

```
command=root.destroy
```

destroy به معنای از بین بردن است.

## ایجاد رادیو باتن:

```
from tkinter import*
root = Tk()
#متغییر از نوع عدد صحیح
v = IntVar()
Label(root, text="کدام را انتخاب می کنید؟", justify = LEFT, padx = 20).pack()
Radiobutton(root, text="Python", padx = 20, variable=v, value=1).pack(anchor=W)
Radiobutton(root, text="php", padx = 20, variable=v, value=2).pack(anchor=W)
mainloop()
```



## ایجاد چک باکس:

```
from tkinter import *
master = Tk()
var1 = IntVar()
Checkbox(master, text="male", variable=var1).grid(row=0, sticky=W)
var2 = IntVar()
Checkbox(master, text="female", variable=var2).grid(row=1, sticky=W)
mainloop()
```

## نمایش متن ورودی تکست با کلیک روی دکمه:

```
from tkinter import *
master = Tk()
# ایجاد تابع
def show_entry_fields():
    # ایجاد برچسب در ردیف دو و ستون دو
    Label(master, text=e1.get()).grid(row=1, column=1)
    # ایجاد برچسب در ردیف یک
    Label(master, text="نام:").grid(row=0)
    # ایجاد ورودی متن
    e1 = Entry(master)
    # محل قرارگیری ورودی متن در ردیف یک و ستون دو
    e1.grid(row=0, column=1)
    # دکمه ای که با کلیک روی اون تابع فعال می شود
    Button(master, text='ارسال', command=show_entry_fields).grid(row=2, column=1,
        sticky=W, pady=4)
mainloop()
```

با کلیک روی دکمه تابع `show_entry_fields` فراخوانی میشه و متن از تکست توسط دستور `e1.get` گرفته میشه و به برچسب داده میشه و در برچسب نمایش داده میشه.

## دکمه ی تغییر متن و خروج:

در این برنامه تابعی نوشتیم که با کلیک روی دکمه متن رو به متن دیگه تغییر بده و همچنین تابعی نوشتیم که با کلیک روی اون پنجره بسته بشه.

```
from tkinter import *
```

```

تابعی برای بستن پنجره#
def stopProg(e):
    مقدار داخل تابع#
    root.destroy()
    تابعی برای تغییر متن#
def transfertext(e):
    label1.configure(text="متن جایگزین")
root=Tk()
button1=Button(root,text="خروج")
button1.pack()
button1.bind('<Button-1>',stopProg)
button2=Button(root,text="کلیک کنید")
button2.pack()
button2.bind('<Button-1>',transfertext)
label1=Label(root,text="متن اصلی")
label1.pack()
root.mainloop()

```

در tkinter فاصله هم مهم هست اگر رعایت نشه برنامه کار نمیکنه. با طریقه ی فاصله گذاشتن در برنامه های بالا دقت کنید. وقتی هر خط دستوری رو می نویسند enter کنید تا کرسر درست سر جای خودش قرار بگیره.

<Button-1> برای drag and drop کشیدن و انداخت.

<B1-Motion> برای دابل کلیک کردن است.

<ButtonRelease-1> کلیک روی دکمه ی سمت چپ ماوس.

<Double-Button-1> برای دابل کلیک کردن.

<Enter> وقتی اشاره گر ماوس میره روی دکمه، دکمه عمل میکنه.

<Leave> خارج شدن کرسر ماوس از روی دکمه .

<Configure> : تغییر اندازه.

<http://www.i-programmer.info/programming/python/5073-creating-the-python-ui-with-tkinter.html?start=1>

## ایجاد پنجره روی هم:

```

from tkinter import *
root = Tk()
top = Toplevel()
top.mainloop()

```

## ساخت فرم با اندازه ی عرض و ارتفاع مشخص:

```

from tkinter import *
root = Tk()
frame = Frame(root, width=500, height=300)
frame.pack()
root.mainloop()

```

## رنگ زمینه برای فرم:

```

from tkinter import *
root = Tk()

```



```

# اندازه ی پنجره
root.geometry("150x100")
frame = Frame(root, width=50, height=50, bg="blue")
frame.pack()
root.mainloop()

```

اگر دقت کنید فقط محدوده ای که متعلق به فرم هست آبی می شود.

## ایجاد دکمه روی فرم:

```

from tkinter import*
root = Tk()
# ایجاد فرم
nameform = Frame(root)
# نمایش فرم
nameform.pack()
# ایجاد دکمه
button1 = Button(nameform)
button1["text"]= "!متن روی دکمه"
# رنگ زمینه ی دکمه سبز است
button1["background"] = "green"
# نمایش دکمه
button1.pack()
root.mainloop()

```

## :Listbox

```

from tkinter import *
master = Tk()
listbox = Listbox(master)
listbox.pack()
listbox.insert(END, "متن جدید")
for item in ["متن اول", "متن دوم", "متن سوم", "متن چهارم"]:
    listbox.insert(END, "متن جدید")
mainloop()

```

## : OptionMenu

```

from tkinter import *
master = Tk()
variable = StringVar(master)
# مقدار پیشفرض
variable.set("گزینه ی اول")
# کل مقادیر
w = OptionMenu(master, variable, "گزینه ی سوم", "گزینه ی دوم", "گزینه ی اول")
w.pack()
mainloop()

```

برنامه ی قبل برای رشته بود ولی این برنامه برای عدد صحیح است(int)

```

from tkinter import *
master = Tk()
variable = IntVar(master)

```

```

مقدار پیشفرض
variable.set(1)
کل مقادیر
w = OptionMenu(master, variable, 1, 2, 3)
w.pack()
mainloop()

```

## : Combobox

```

from tkinter import Tk, StringVar, ttk
root = Tk()
مقدار رشته ای
value = StringVar()
box = ttk.Combobox(root, textvariable=value, state='readonly')
مقادیر به صورت آرایه
box['values'] = ('صادق', 'مهدی', 'راوندی')
صفحه ی جاری یا پیشفرض ما صفحه ی اول باشد
# مقادیر آرایه همیشه از صفر شروع می شود
box.current(0)
ستون اول و ردیف اول
box.grid(column=0, row=0)
root.mainloop()
state='readonly' وضعیت فقط خواندنی. در صورتی که writeonly باشد می توانید به غیر از خواندن حتی داخل آن متنتان را بنویسید.

```

## ایجاد اسکرول:

```

from tkinter import *
root = Tk()
scrollbar = Scrollbar(root)
scrollbar.pack(side=RIGHT, fill=Y)
mainloop()

```

## ایجاد لغزنده:

```

from tkinter import *
master = Tk()
w = Scale(master, from_=0, to=42)
w.pack()
w = Scale(master, from_=0, to=200, orient=HORIZONTAL)
w.pack()
mainloop()
HORIZONTAL به معنای افقی است. که باعث میشه یک لغزنده ی افقی ایجاد بشه.
from_=0: لغزنده از این قسمت شروع میشود.
To : تا اینجا پایان میابد.

```

## ایجاد منو و زیر منو:

```

from tkinter import *

```

```

def tabenew():
    print ("گزینه ي جديد انتخاب شد")
def tabeopen():
    print ("گزینه ي باز کردن انتخاب شد")
def tabehelp():
    print ("راهنماي برنامه")
root = Tk()
menu = Menu(root)
root.config(menu=menu)

filemenu = Menu(menu)
menu.add_cascade(label="File", menu=filemenu)
filemenu.add_command(label="New", command=tabenew)
filemenu.add_command(label="Open...", command=tabeopen)
filemenu.add_separator()
filemenu.add_command(label="Exit", command=root.destroy)

helpmenu = Menu(menu)
menu.add_cascade(label="Help", menu=helpmenu)
helpmenu.add_command(label="About...", command=tabehelp)

mainloop()

```

با کلیک روی هر زیر منو تابع مربوط به آن زیر منو فراخوانی شده و متن متعلق به آن در ویرایشگر نمایش داده می شود.

<http://www.java2s.com>

## ایجاد تب :

```

import tkinter as tk
import tkinter.ttk as ttk

root = tk.Tk()
# محور عمودی + محور افقی + ارتفاع x عرض
root.geometry("%dx%d+%d+%d" % (300, 200, 100, 50))
root.title('عنوان')

nb = ttk.Notebook(root)
nb.pack(fill='both', expand='yes')

# ایجاد فرم برای هر صفحه
f1 = tk.Frame(bg='red')
f2 = tk.Frame(bg='blue')
f3 = tk.Frame(bg='green')

# ایجاد صفحات
nb.add(f1, text='page1')
nb.add(f2, text='page2')
nb.add(f3, text='page3')

```

```
# این دکمه با توجه به اف یک به تب اول داده می شود
btn1 = tk.Button(f1, text='button1')
btn1.pack(side='left', anchor='nw', padx=50, pady=50)
root.mainloop()
```

pady =50 : باعث می شود که دکمه از سمت بالای پنجره فاصله بگیرد.  
 padx=50 : از سمت راست پنجره فاصله میگیرد.  
 X و y مختصات افقی و عمودی.

anchor='nw' : باعث می شود که دکمه در مختصات صفر از پنجره یعنی در گوشه ی بالا سمت چپ پنجره قرار بگیرد. یعنی شمال غربی.  
 مقادیری که anchor میگیرد:

s : جنوب.

n : شمال.

w : غرب.

e : شرق.

و شمال غربی و شرقی و جنوب غربی و شرقی .

<http://forums.devshed.com/python-programming-11/python-tkinter-tabs-708230.html>

## ایجاد خط:

```
from tkinter import*
master = Tk()
w = Canvas(master, width=300, height=200)
w.pack()
w.create_line(150, 80, 200, 200, fill="#476042", width=5)
mainloop()
```

create\_line : خط را با توجه به مختصات داده شده ایجاد میکند.  
 fill="#476042 : خط با این رنگ پر شود.  
 width=5 : ضخامت خط می باشد.

## ایجاد مستطیل:

```
from tkinter import *
master = Tk()
w = Canvas(master, width=200, height=100)
w.pack()
w.create_rectangle(50, 20, 150, 80, fill="#9e28f1")
mainloop()
```

create\_rectangle : برای ایجاد مستطیل .  
 fill : به معنای پر کردن و توپر می باشد.

#9e28f1 : کد رنگ هگزاد دسیمال است که این کد برای رنگ آبی است.  
 برای اینکه رنگ های بیشتری انتخاب کنید از فتوشاپ استفاده کنید .

برنامه ی فتوشاپ را باز کنید و به قسمت ابزار فتوشاپ رفته در پایین این قسمت روی تصویر زیر کلیک کنید:



حتی می تونید رنگ یک عکس رو هم بگیرید کافیه ابزار Eyedropper Tool رو انتخاب کنید و روی عکس کلیک کنید بعد دوباره مثل قبل و طبق تصویر بالا عمل کنید تا رنگ تصویر را بگیرید.

## ایجاد دایره :

```
from tkinter import *
master = Tk()
w = Canvas(master, width=150, height=150)
w.pack()
w.create_oval(50,50,100,100 ,fill="#9e28f1")
mainloop()
```

oval به معنای بیضی. می توانیم با دادن اندازه به آن یک دایره هم ایجاد کنیم.

## ایجاد چند ضلعی:

```
from tkinter import *
master = Tk()
w = Canvas(master, width=210, height=170)
w.pack()
points = [(20, 20), (50, 150), (200, 50)]
w.create_polygon(fill="blue", *points)
root.mainloop()
```

برای دادن مختصات نقاط  $x$  و  $y$  شروع و پایان خطوط از روش زیر استفاده می شود و به  $points$  داده می شود:

```
[(x1, y1), (x2, y2), (x3, y3),.....]
```

اینکه چند نقطه گذاشتم، خواستم بگم که شما می توانید نقاط  $x$  و  $y$  بیشتری داشته باشید .  
مثلا برای ساخت ستاره باید از  $x4, y4$  و بیشتر استفاده کنید.  
با روش بالا ما یک مثلث ایجاد کردیم..

## شی گرابی:

فلسفه ای که منجر به شکل گیری زبان های شی گرا شد، عبارت بود از این واقعیت که نحوه ی عملکرد مغز و شیوه ی دریافت اطلاعات از محیط پیرامون و پردازش آن اطلاعات (اندیشیدن)، شیوه ای شی گراست از همین رو می بایستی زبانی تعریف می شد که همین شیوه را مبنای کار خود قرار داده و دقیقا به همان شکل که اشیا در جهان خارج، دارای هویت و کارکرد مشخص و یگانه برای خود هستند و در عین حال با دیگر اشیا در ارتباطی تنگاتنگ و مستقیم اند، در زبان های برنامه نویسی نیز می بایستی این اصول کلی برقرار می شد تا به این شکل، برنامه نویسی

بتواند با بهره گیری از زبانی که به روش اندیشیدنش، نزدیکی بیشتری دارد، شی های مختلفی را تعریف نموده، این شی ها را در ارتباط با یکدیگر قرار داده و از شی های پویای تولید شده برای حل مساله ی پیش روی استفاده نماید.

از دیگر ویژگی های برنامه نویسی شی گرا، پیچیدگی کم، هزینه ی کم، امکان گسترش سریع برنامه با خطای کمتر نسبت به سایر پارادایم های برنامه نویسی است.

سه هدف اصلی برنامه نویسی مبتنی بر اشیاء:

- ۱) غلبه بر پیچیدگی
- ۲) مدیریت تغییر
- ۳) قابلیت استفاده مجدد.

### کلاس :

کلاس نوع خاصی از آرایه است که امکان ذخیره و جمع بندی تعداد زیادی المان را (که شامل داده ها و توابع هستند) برای یک شی جدید فراهم آورده و تحت یک نام مشترک ذخیره می کند. اما برخلاف آرایه که تنها قادر به ذخیره ی داده هایی هم جنس است، کلاس می تواند داده هایی از جنس های مختلف را ذخیره نماید.

منبع : ویکی پدیا.

[http://fa.wikipedia.org/wiki/شیء\\_گرا#cite\\_note-3](http://fa.wikipedia.org/wiki/شیء_گرا#cite_note-3)

می توانیم کدی رو چند بار در چند قسمت به کار ببریم . مثلا یک خصوصیت رو در چند قسمت استفاده کنیم.

مثلا برای دادن خصوصیت به دکمه :

```
from tkinter import *
class dokme:
    def __init__(self, master):
        fm = Frame(master)
        Button(fm, text='Left').pack(side=TOP)
        Button(fm, text='Center').pack(side=LEFT)
        Button(fm, text='Right').pack(side=LEFT)
        fm.pack()
```

```
root = Tk()
root.option_add('*font', ('verdana', 12, 'bold'))
root.title("oop")
display = dokme(root)
root.mainloop()
```

در مثال بالا یک سری خصوصیات رو دادیم به فونت همه ی دکمه ها. بدون اینکه مجبور بشیم به تک تک دکمه ها خصوصیت بدیم.

یک مثال ساده بدون شی گرای:

```
from tkinter import *
root = Tk()
frame = Frame(root)
frame.pack()
bottomframe = Frame(root)
```

```
bottomframe.pack( side = BOTTOM )
```

```
redbutton = Button(frame,font = "Verdana 10 bold", text="left", fg="red")  
redbutton.pack( side = LEFT)
```

```
bluebutton = Button(frame,font = "Verdana 10 bold", text="right", fg="blue")  
bluebutton.pack( side = LEFT )
```

```
blackbutton = Button(bottomframe,font = "Verdana 10 bold", text="Center",  
fg="black")  
blackbutton.pack( side = BOTTOM)
```

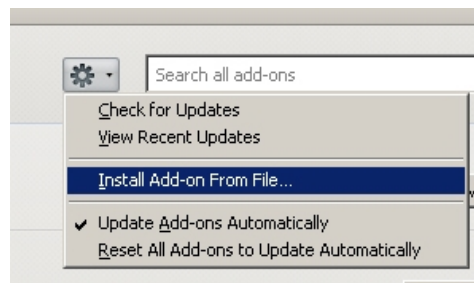
```
root.mainloop()
```

## بانک اطلاعاتی:

میخواهیم مقادیر متن را داخل یک جدول در بانک اطلاعاتی ذخیره کنیم برای اینکار از بانک اطلاعاتی SQLite استفاده میکنیم. برنامه ی SQLite Manager 0.7.7 که از Extensions های فایرفاکس هست خیلی کار ما را در کار با بانک اطلاعاتی راحت می کند. برای دانلود این برنامه به آدرس زیر بروید:

<https://addons.mozilla.org/en-US/firefox/addon/sqlite-manager/>

بعد از دانلود ، مرورگر فایرفاکس را باز کنید و به منوی Tools و سپس Add ons رفته و گزینه ی Extensions را انتخاب کنید. و طبق تصویر زیر با استفاده از گزینه ی Install Add-on From file برنامه ی SQLite Manager 0.7.7 را انتخاب کنید و نصب کنید.



بعد برای انتخاب این برنامه به منوی Tools رفته و گزینه ی SQLite Manager را انتخاب کنید. برای ایجاد دیتابیس به منوی Database و سپس New Database را انتخاب کنید. نام دیتابیس را مثلا ravandi وارد کنید و ok کنید. برنامه از شما خواهد خواست که فایل را در جایی ذخیره کنید ما در آدرس زیر فایل را ذخیره کردیم:

F:/sql/ravandi.sqlite

شما هم هر جایی دوست داشتید ذخیره کنید ولی بعداً باید در برنامه درست آدرس دهی کنید. بعد در سمت چپ روی Tables راست کلیک کرده و گزینه ی create table را انتخاب کنید. در پنجره ی باز شده و در قسمت Table Name نام جدول را مثلاً tkinter وارد کنید. حالا میخواهیم سه فیلد بسازیم. در قسمت Column Name اولین فیلد را با نام id می نویسیم. که شماره ی رکورد را ثبت میکند. از نوع (Data Type) integer (عدد صحیح) یعنی فقط مقدار عدد به این فیلد وارد شود.

Primary key را Yes می کنیم که فیلد اصلی ما باشد. بعد گزینه ی Autoinc را هم انتخاب می کنیم که شماره ی هر رکورد پشت سر هم با ایجاد هر رکورد ایجاد بشه. فیلد بعدی با نام مثلاً name از نوع varchar و تیک Allow Null را بر میداریم یعنی نمی خواهیم خالی باشد. فیلد بعدی

هم lastname از نوع varchar و تیک Allow Null را هم بر میداریم. البته اگر Null باشد در حالت پیشفرض Null است. یعنی خالی است و اگر فعال هم باشد مشکلی نیست.

Column Name	Data Type	Primary Key?	Autoinc?	Allow Null?	Unique?	Default Value
id	INTEGER	<input checked="" type="checkbox"/> Yes	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	
name	VARCHAR	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	
lastname	VARCHAR	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	

حالا وقتی روی علامت بعلاوه ی Table کلیک میکنید نام جدول را می بینید که با کلیک راست روی آن و زدن گزینه ی Drop table می توانید جدول را پاک کنید. برای وارد کردن یک متن به داخل جدول از روش زیر استفاده کنید :

```
import sqlite3
# اتصال به بانک اطلاعاتی
conn = sqlite3.connect('F:/sql/ravandi.sqlite')
c = conn.cursor()
# وارد کردن یک رکورد به جدول
c.execute("""insert into tkinter (name,lastname) values ('حسینیان راوندی','محمدرضا')""")
# این دستور برای مطمئن شدن از ارسال مقادیر است
conn.commit()
# بستن اتصال به دیتابیس
conn.close()
رکورد اینجا منظور اون مقادیری است که با هر بار اجرای برنامه وارد جدول می شوند. یعنی هر بار مقادیری که ارسال میکنیم ثبت می شود. و می شود یک رکورد.
حالا می توانید به جای مقادیر محمدرضا و حسینیان راوندی مقادیر دیگری را قرار بدهید . و وقتی دکمه ی F5 را می زنید یک مقدار جدید به جدول اضافه می شود و شماره ی رکورد یعنی شماره ی فیلد id به طور خودکار یک عدد اضافه می شود. چون تیک گزینه ی Autoinc را زدیم.
برای دیدن رکوردهای ثبت شده در برنامه ی SQLite Manager ، اول جدول مورد نظر یعنی tkinter را در سمت چپ انتخاب کنید بعد در سمت راست روی تب Browse & Search کلیک کنید و برای دیدن رکورد های ثبت شده روی دکمه ی Search کلیک کنید. مقادیر ثبت شده در جدول را مشاهده می کنید.
```

## وارد کردن متن به جدول با کمک فرم:

منبع: <<

<http://bytes.com/topic/python/answers/901064-help-combining-tkinter-mysql>

```
from tkinter import *
import sqlite3
conn = sqlite3.connect('F:/sql/ravandi.sqlite')
cursor = conn.cursor ()
def SaveData():
    f = a.get()
    g = b.get()
```



```

    cursor.execute ("""INSERT INTO tkinter (name, lastname) VALUES("%s", "%s")"""%(
f,g))
    conn.commit()
    conn.close()
root = Tk()
ent_frame = Frame(root)
Label(ent_frame, text="FIRST NAME:").pack(side=LEFT)
a = Entry(ent_frame, width=15)
a.pack(side=LEFT)
Label(ent_frame, text="LAST NAME:").pack(side=LEFT)
b = Entry(ent_frame, width=15)
b.pack(side=LEFT)
ent_frame.pack()
Button(root, text="Save",command=SaveData).pack(side=BOTTOM)
mainloop()

```

>>

توضیحات: با کلیک روی دکمه تابع فراخوانی همیشه و مقادیر رو از ورودی ها با get() میگیره و میده به خط insert و اونم وارد جدول میکنه.

## نمایش رکوردهای ثبت شده:

```

import sqlite3
connection = sqlite3.connect('F:/sql/ravandi.sqlite')
cursor = connection.cursor()
#انتخاب همه ی رکوردهای جدول برای نمایش
cursor.execute('SELECT * FROM tkinter')
for row in cursor:
    print (row[0], row[1], row[2])
connection.commit()
connection.close()
علامت ستاره * برای تمام رکوردهاست. اگر میخواهید فقط مقادیر یک فیلد نمایش داده شود فقط نام همان فیلد را بنویسید. در سطر بعدی یک حلقه ی for ایجاد می شود و مقادیر رکوردهای داخل جدول پشت سر هم چاپ می شود.
print (row[0], row[1], row[2])
خط بالا شماره های رکورد ها و مقادیری که به فیلدهای name و lastname داده شده را نمایش می دهد.

```

حالا برنامه ای می نویسیم که مقادیر یک فیلد از جدول را روی برجسب نمایش دهد:

```

from tkinter import *
import sqlite3
connection = sqlite3.connect('F:/sql/ravandi.sqlite')
cursor = connection.cursor()
cursor.execute('SELECT * FROM tkinter')
root = Tk()
namefram= Frame(root)
for row in cursor:
    mname=row[1]
    Label(namefram, text=mname).pack()

```

```
connection.commit()
connection.close()
namefram.pack()
mainloop()
```

امدیم مقادیر row[1] (که همیشه مقادیر فیلد name) را دادیم به متغیر mname و این متغیر را دادیم به برجسب تا نمایش داده بشه.

**برای دیدن بقیه ی دستورات sql به این آدرس بروید:**

<http://www.w3schools.com/sql/>

اگر کتاب دیگری بنویسم یا بخوام در مورد مطالب این کتاب مطلبی بنویسم و یا اصلاحی انجام بدم در آدرس خوراک خوان سایتتم به آدرس زیر اطلاع میدم:

<http://pachenar.ir/rss-learn.php>

منابع:

<http://www.ferg.org>

<http://www.i-programmer.info>

<http://www.python-course.eu>

<http://www.ferg.org>

<http://www.tutorialspoint.com>

<http://stackoverflow.com>

<http://docs.python.org>