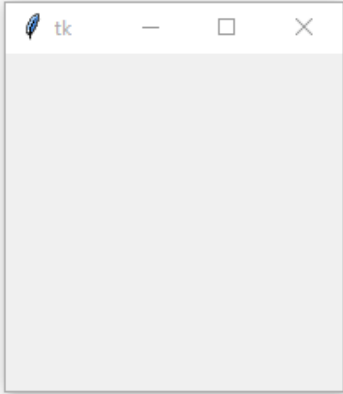


Python Tkinter Introducción (Interface gráfica)

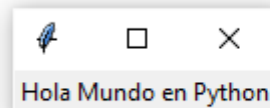
```
Python 3.5.0 Shell
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> from tkinter import *
>>> |
```

Ln: 4 Col: 4

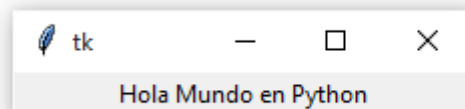
```
Python 3.5.0 Shell
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> from tkinter import *
>>> ventana = Tk()
>>>
```



```
>>> etiqueta = Label(ventana, text="Hola Mundo en Python")
>>> etiqueta.pack()
>>>
```



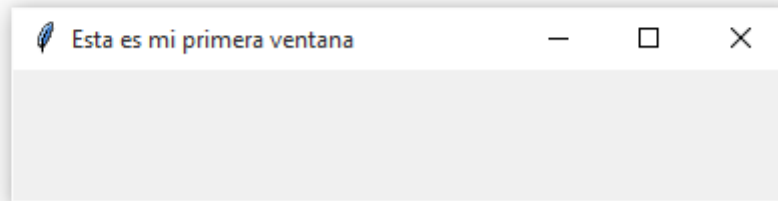
```
>>> etiqueta.mainloop()
```



Podemos interactuar con la ventana y hasta que no la cerremos no volverá a salir el cursor.

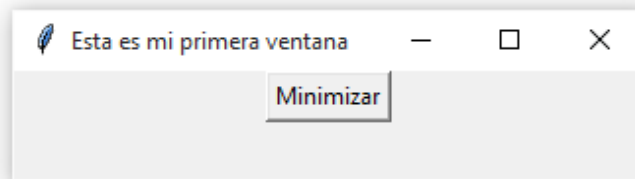
```
>>> etiqueta.mainloop()
>>> |
```

```
>>> ventana = Tk()
>>> ventana.title("Esta es mi primera ventana")
>>>
```



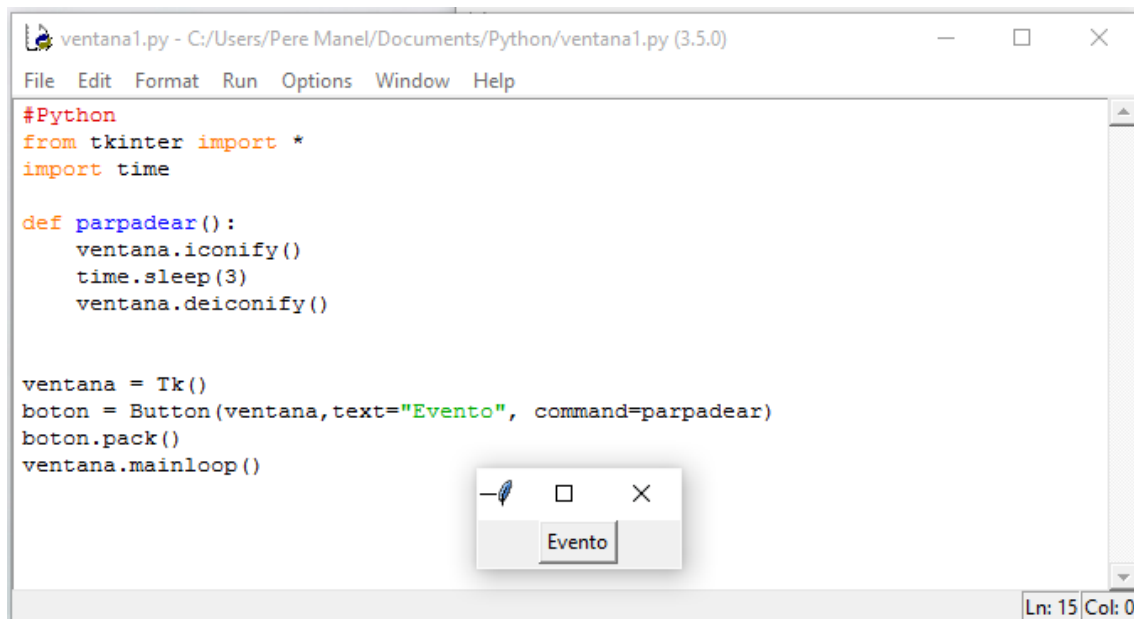
Llamamos de nuevo a la ventana y le damos un título.

```
>>> boton = Button(ventana, text="Minimizar", command=ventana.iconify)
>>> boton.pack()
>>>
```

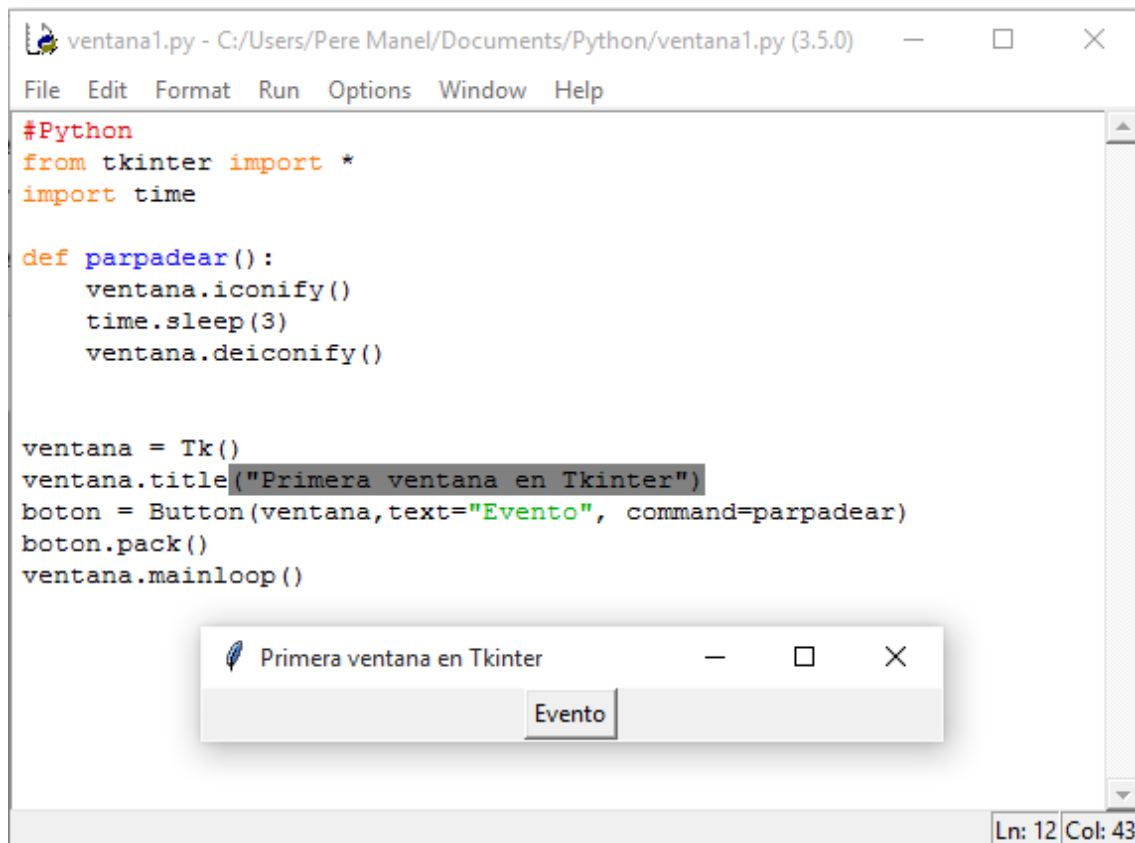


Agregamos un botón a la ventana y la instrucción 'command=ventana.iconify' lo que hace es minimizar.

Ahora con todo lo que hemos aprendido vamos a realizar un programa:

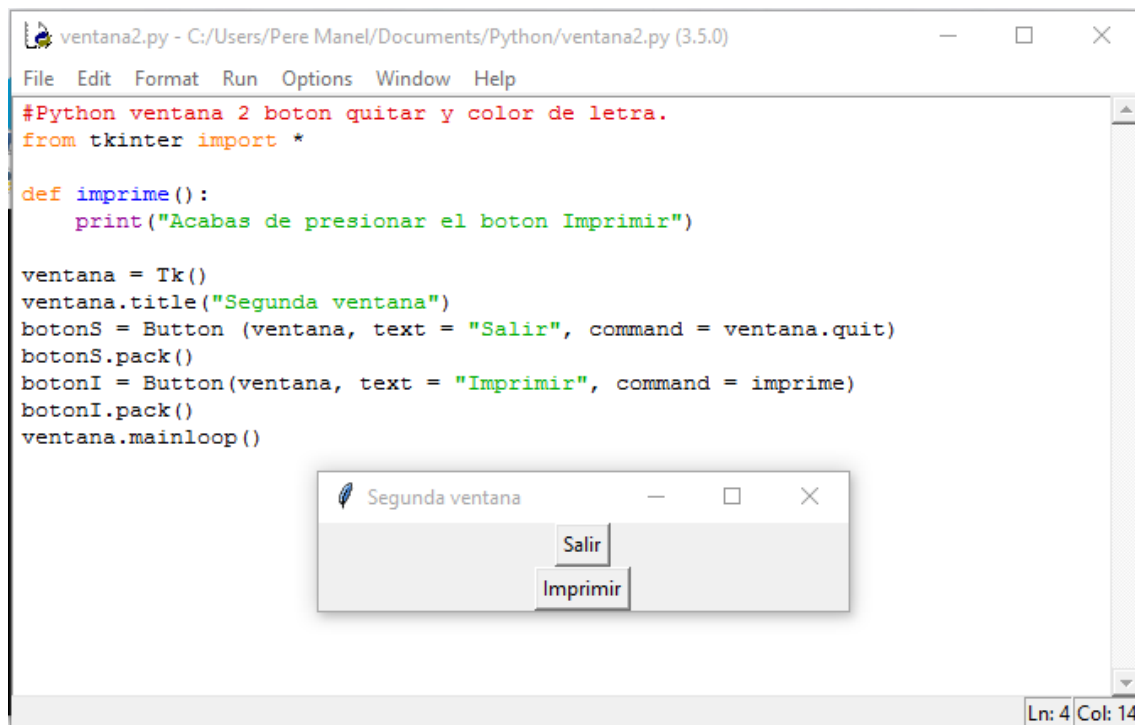


Este ejercicio hace que salga una ventana con un botón llamado Evento cuando le damos al botón esta ventana se minimiza durante 3 segundos y se vuelve a reestablecer, esto lo controla el método def parpadear():



Le hemos agregado un título en la ventana "Primera ventana en Tkinter".


Python Práctica con Botones



```
ventana2.py - C:/Users/Pere Manel/Documents/Python/ventana2.py (3.5.0)
File Edit Format Run Options Window Help
#Python ventana 2 boton quitar y color de letra.
from tkinter import *

def imprime():
    print("Acabas de presionar el boton Imprimir")

ventana = Tk()
ventana.title("Segunda ventana")
botonS = Button(ventana, text = "Salir", fg= "red" ,command = ventana.quit)
botonS.pack()
botonI = Button(ventana, text = "Imprimir", fg= "Blue", command = imprime)
botonI.pack()
ventana.mainloop()
```

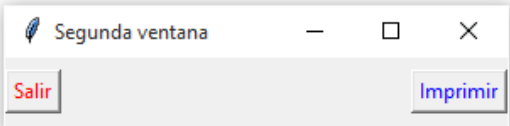


Ln: 14 Col: 0

```
ventana2.py - C:/Users/Pere Manel/Documents/Python/ventana2.py (3.5.0)
File Edit Format Run Options Window Help
#Python ventana 2 boton quitar y color de letra.
from tkinter import *

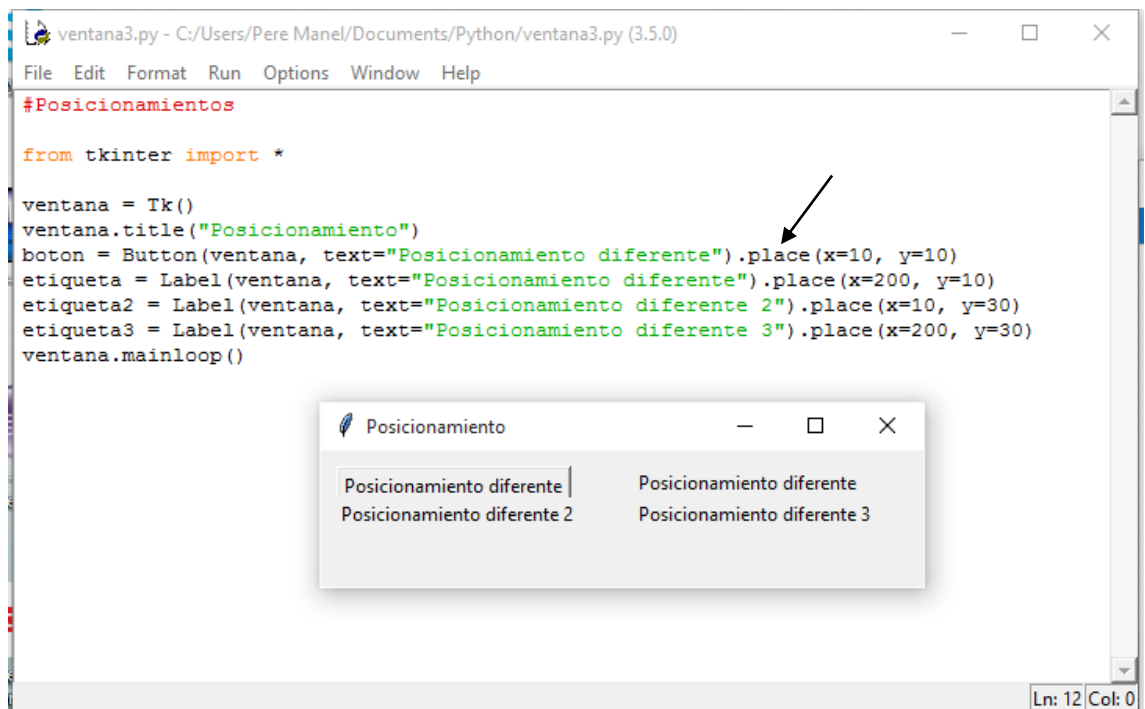
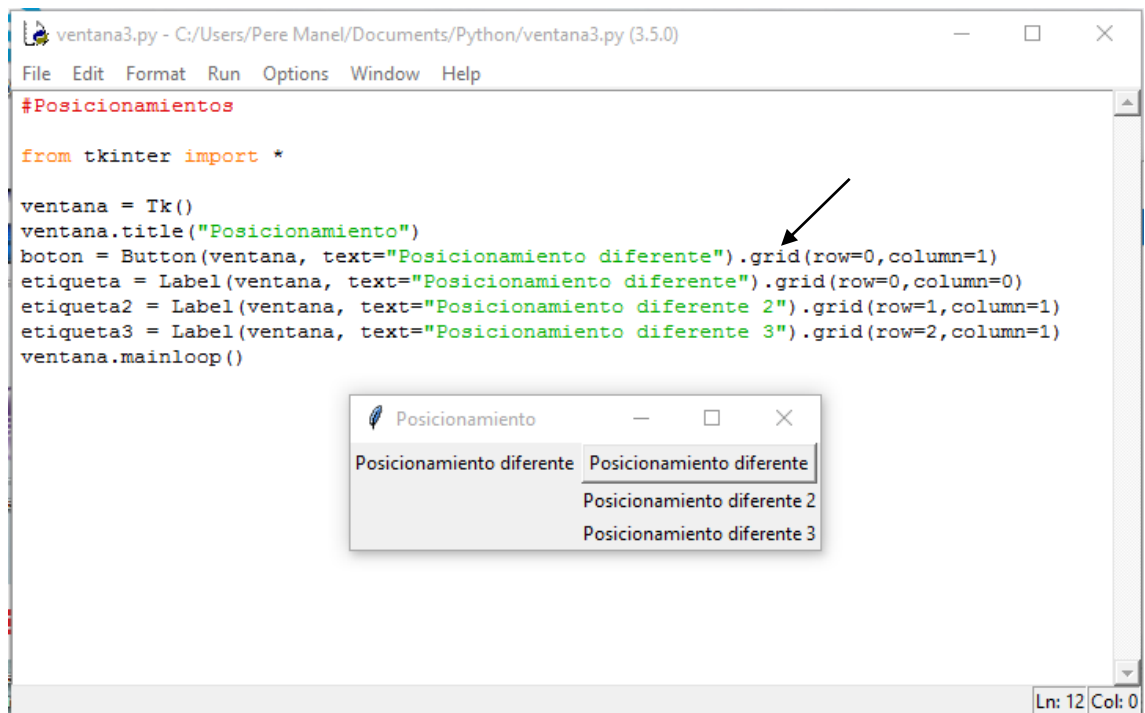
def imprime():
    print("Acabas de presionar el boton Imprimir")

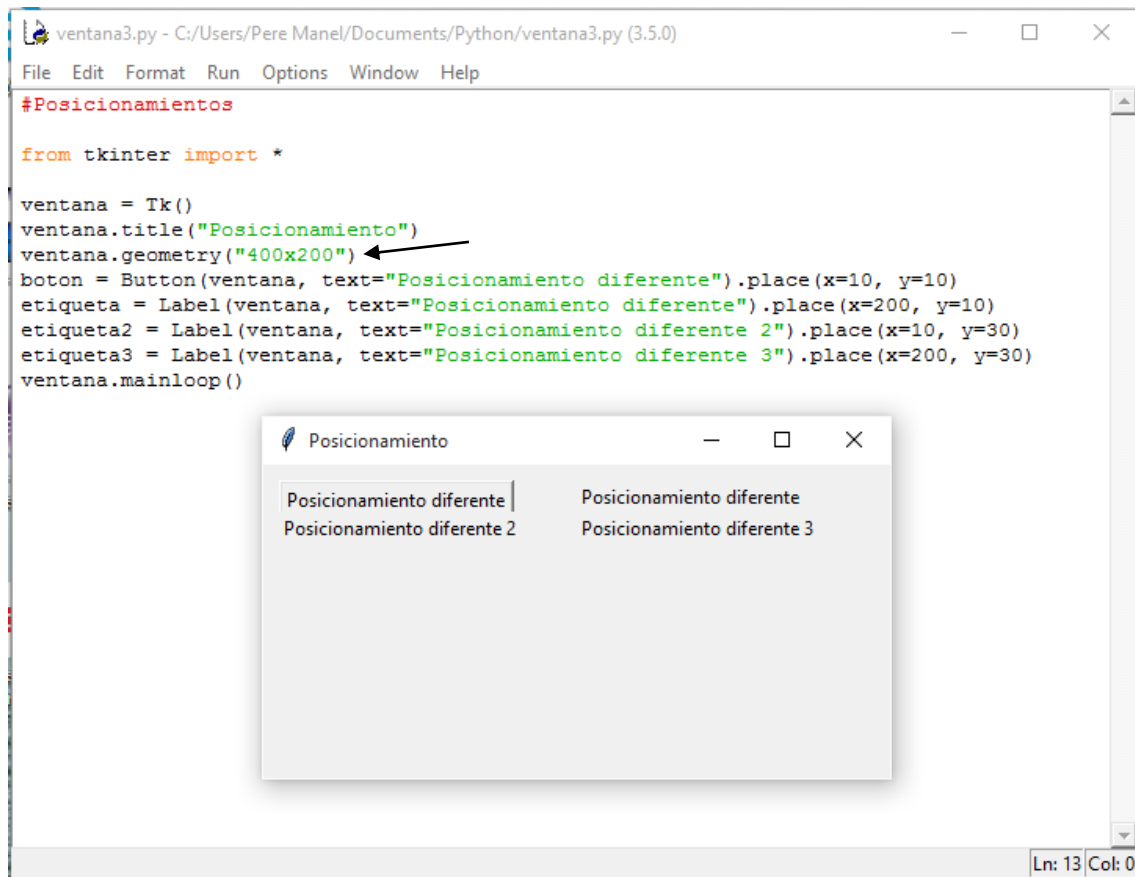
ventana = Tk()
ventana.title("Segunda ventana")
botonS = Button(ventana, text = "Salir", fg= "red" ,command = ventana.quit)
botonS.pack(side=LEFT)
botonI = Button(ventana, text = "Imprimir", fg= "Blue", command = imprime)
botonI.pack(side=RIGHT)
ventana.mainloop()
```



Ln: 12 Col: 22

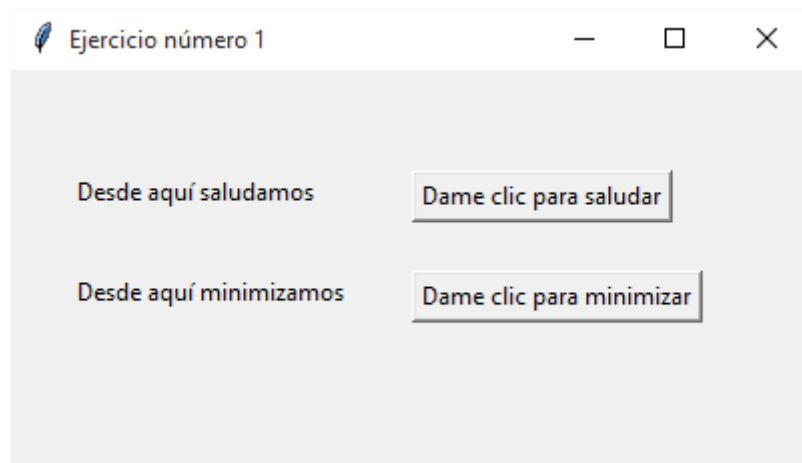
Python Posicionamientos





Ejercicio propuesto por el profesor.

Este tiene que ser el resultado:



Haciendo clic en 'Dame clic para saludar' que diga este mensaje por consola "Hola a todos".

Haciendo clic en 'Dame clic para minimizar' que la ventana se minimice.

Este será el código:

```
*ejercicio1.py - C:/Users/Pere Manel/Documents/Python/ejercicio1.py (3.5.0)*
File Edit Format Run Options Window Help

#Ejercicio 1

from tkinter import *

def saludo():
    print("Hola a todos")

def minimizar():
    ventana.iconify()

ventana = Tk()
ventana.title("Ejercicio número 1")
ventana.geometry("400x200")
etiqueta1 = Label(ventana, text="Desde aquí saludamos").place(x=30, y=50)
etiqueta2 = Label(ventana, text="Desde aquí minimizamos").place(x=30, y=100)
boton1 = Button(ventana, text="Dame clic para saludar", command=saludo).place(x=200, y=50)
boton2 = Button(ventana, text="Dame clic para minimizar", command=minimizar).place(x=200, y=100)
ventana.mainloop()

Ln: 19 Col: 0
```

Python Tkinter Entradas

```
ventana4.py - C:/Users/Pere Manel/Documents/Python/ventana4.py (3.5.0)
File Edit Format Run Options Window Help

#Ventana 4: Entradas

from tkinter import *

def saludar():
    print("Hola: "+nombre.get()+" "+ apellidoP.get()+" "+ apellidoM.get())

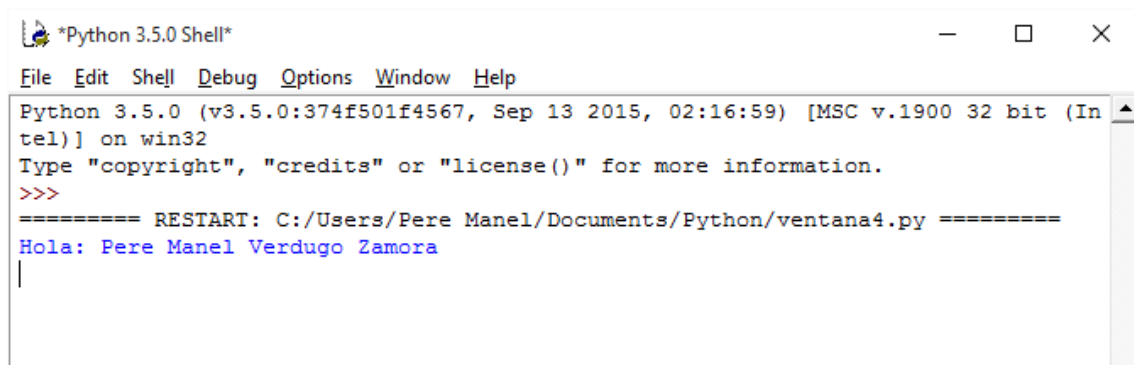
ventana = Tk()
nombre = StringVar()
apellidoP = StringVar()
apellidoM = StringVar()
nombre.set("Escribe tu nombre")
ventana.title("Entradas en tkinter")
ventana.geometry("400x400")
etiqueta1 = Label(ventana, text="Escribe tu nombre: ").place(x=10, y=10)
nombreCaja = Entry(ventana, textvariable=nombre).place(x=170, y=10)
etiqueta2 = Label(ventana, text="Escribe tu apellido Paterno: ").place(x=10, y=40)
apellidoPCaja = Entry(ventana, textvariable=apellidoP).place(x=170, y=40)
etiqueta3 = Label(ventana, text="Escribe tu apellido Materno: ").place(x=10, y=70)
apellidoMCaja = Entry(ventana, textvariable=apellidoM).place(x=170, y=70)
boton = Button(ventana, text="Saludo Personalizado ", command=saludar).place(x=10, y=100)
ventana.mainloop()

Ln: 12 Col: 31
```

Vamos a ejecutar.

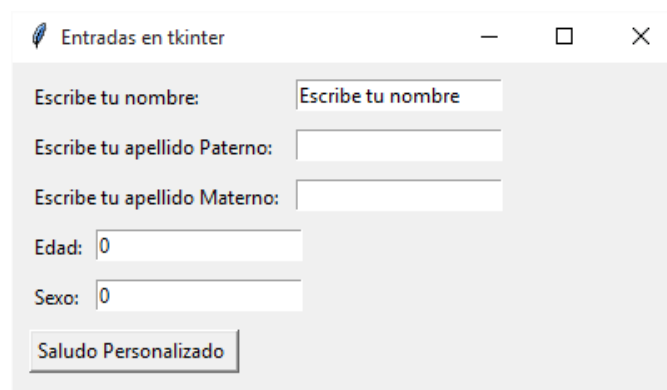
The screenshot shows a window titled "Entradas en tkinter". Inside, there are three labels with corresponding input boxes: "Escribe tu nombre:" with the text "Pere Manel", "Escribe tu apellido Paterno:" with the text "Verdugo", and "Escribe tu apellido Materno:" with the text "Zamora". Below these fields is a button labeled "Saludo Personalizado".

Seleccionaremos el botón 'Saludo Personalizado'.

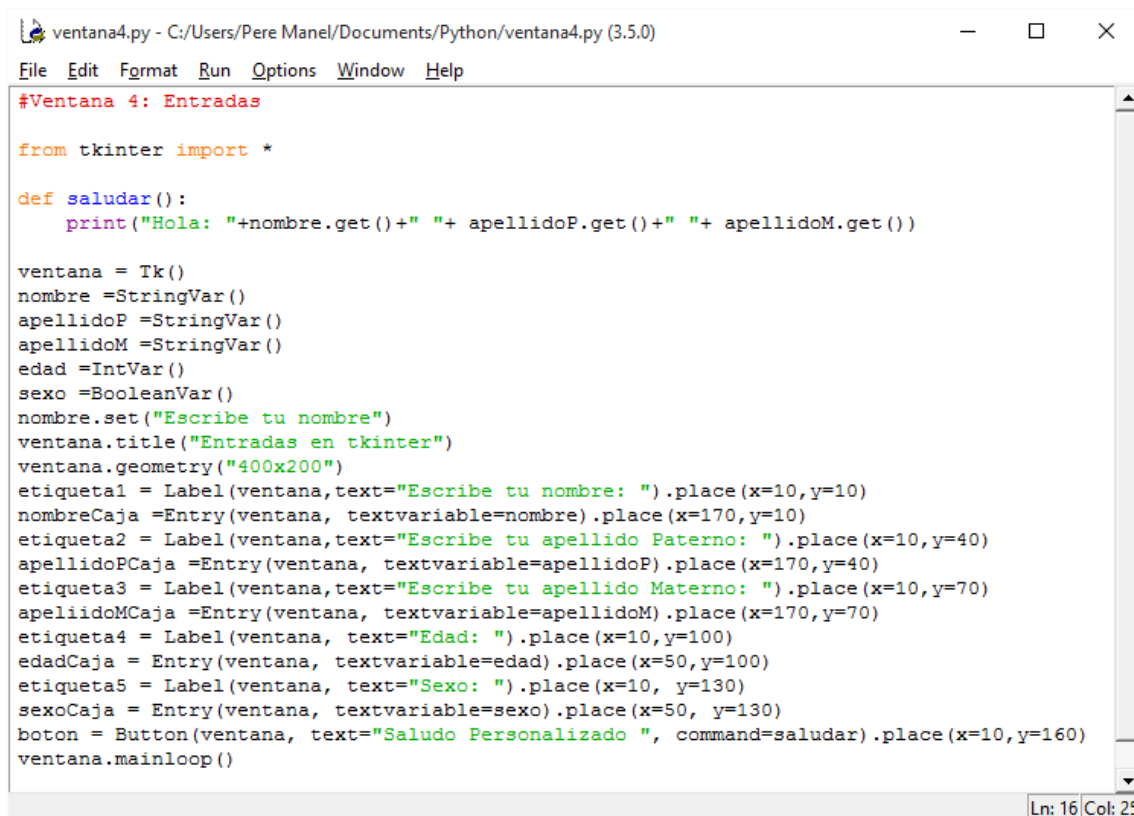


```
*Python 3.5.0 Shell*
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Pere Manel/Documents/Python/ventana4.py =====
Hola: Pere Manel Verdugo Zamora
|
```

Agregar la edad y el sexo.



Este será el código:



```
ventana4.py - C:/Users/Pere Manel/Documents/Python/ventana4.py (3.5.0)
File Edit Format Run Options Window Help
#Ventana 4: Entradas

from tkinter import *

def saludar():
    print("Hola: "+nombre.get()+" "+ apellidoP.get()+" "+ apellidoM.get())

ventana = Tk()
nombre =StringVar()
apellidoP =StringVar()
apellidoM =StringVar()
edad =IntVar()
sexo =BooleanVar()
nombre.set("Escribe tu nombre")
ventana.title("Entradas en tkinter")
ventana.geometry("400x200")
etiqueta1 = Label(ventana,text="Escribe tu nombre: ").place(x=10,y=10)
nombreCaja =Entry(ventana, textvariable=nombre).place(x=170,y=10)
etiqueta2 = Label(ventana,text="Escribe tu apellido Paterno: ").place(x=10,y=40)
apellidoPCaja =Entry(ventana, textvariable=apellidoP).place(x=170,y=40)
etiqueta3 = Label(ventana,text="Escribe tu apellido Materno: ").place(x=10,y=70)
apellidoMCaja =Entry(ventana, textvariable=apellidoM).place(x=170,y=70)
etiqueta4 = Label(ventana, text="Edad: ").place(x=10,y=100)
edadCaja = Entry(ventana, textvariable=edad).place(x=50,y=100)
etiqueta5 = Label(ventana, text="Sexo: ").place(x=10, y=130)
sexoCaja = Entry(ventana, textvariable=sexo).place(x=50, y=130)
boton = Button(ventana, text="Saludo Personalizado ", command=saludar).place(x=10,y=160)
ventana.mainloop()

Ln: 16 Col: 25
```


Python Tkinter radiobutton

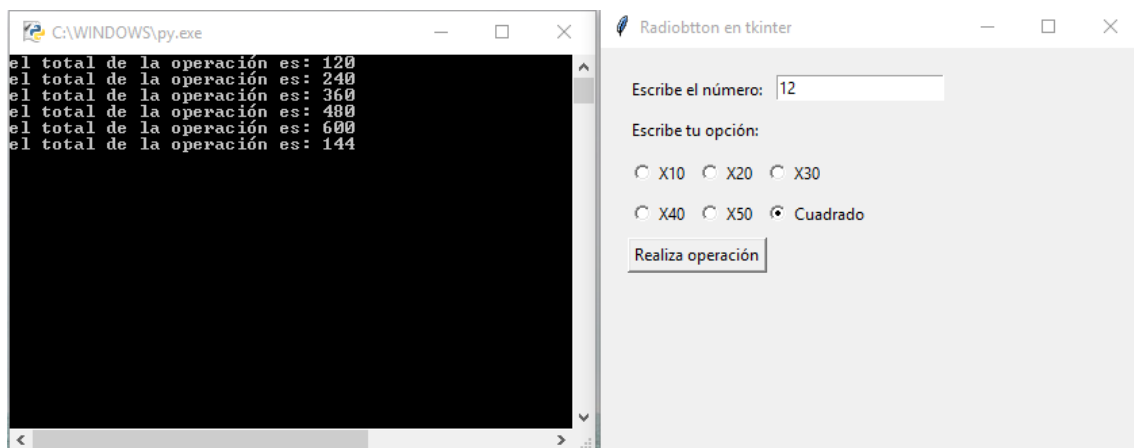
```
ventana5.py - C:/Users/Pere Manel/Documents/Python/ventana5.py (3.5.0)
File Edit Format Run Options Window Help

#Ventana 5: Radio

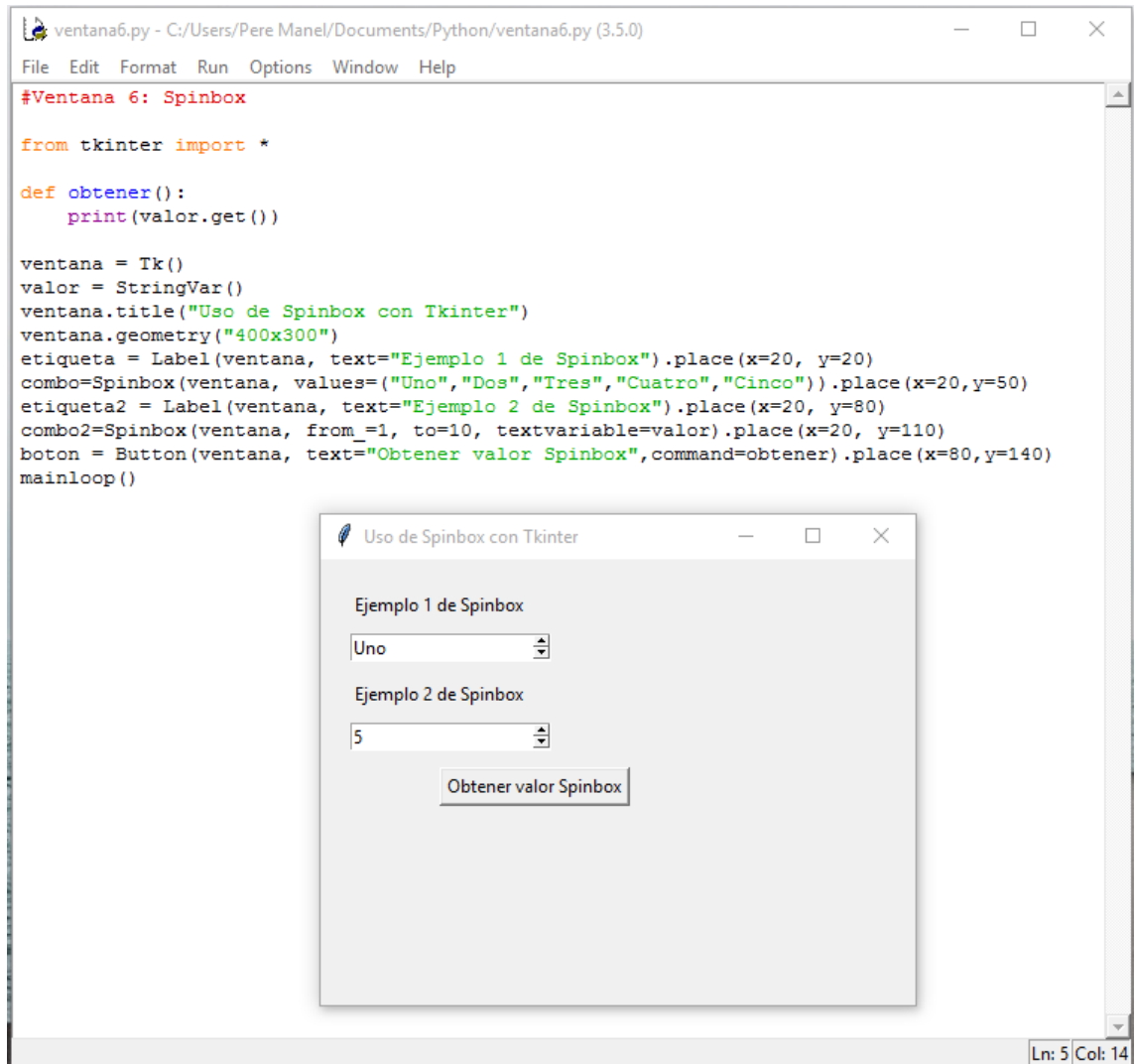
from tkinter import *

def operacion():
    numero = num.get()
    if opcion.get() == 1:
        total = numero * 10
    elif opcion.get() == 2:
        total = numero * 20
    elif opcion.get() == 3:
        total = numero * 30
    elif opcion.get() == 4:
        total = numero * 40
    elif opcion.get() == 5:
        total = numero * 50
    else:
        total = numero * numero
    print("el total de la operación es: " + str(total))

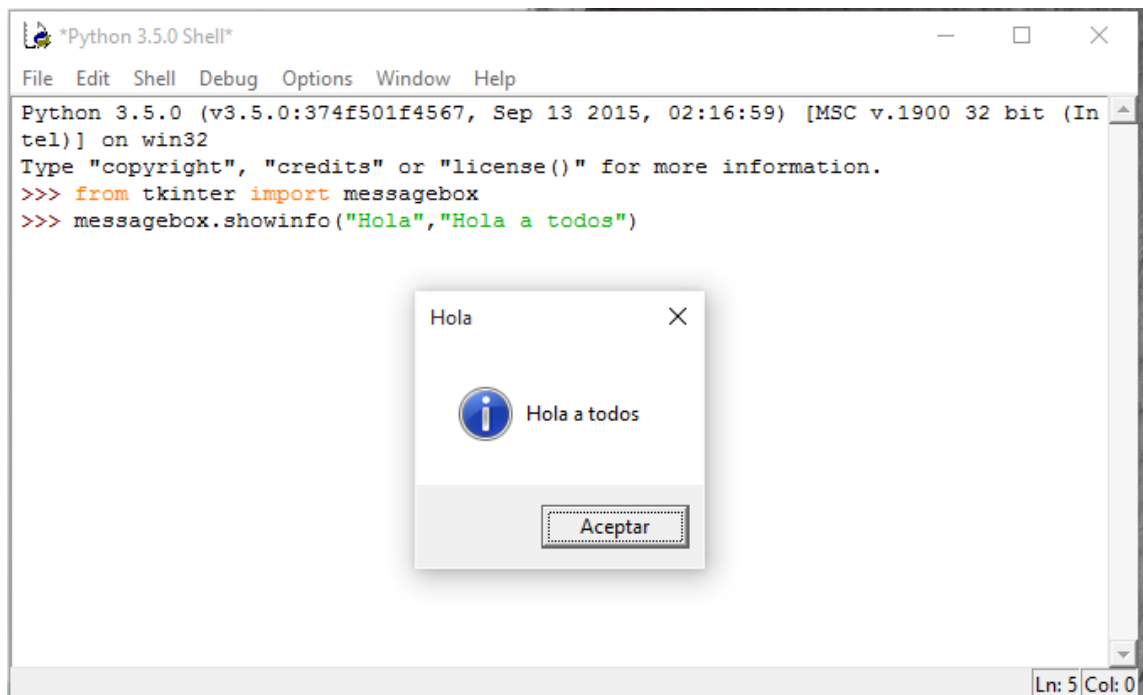
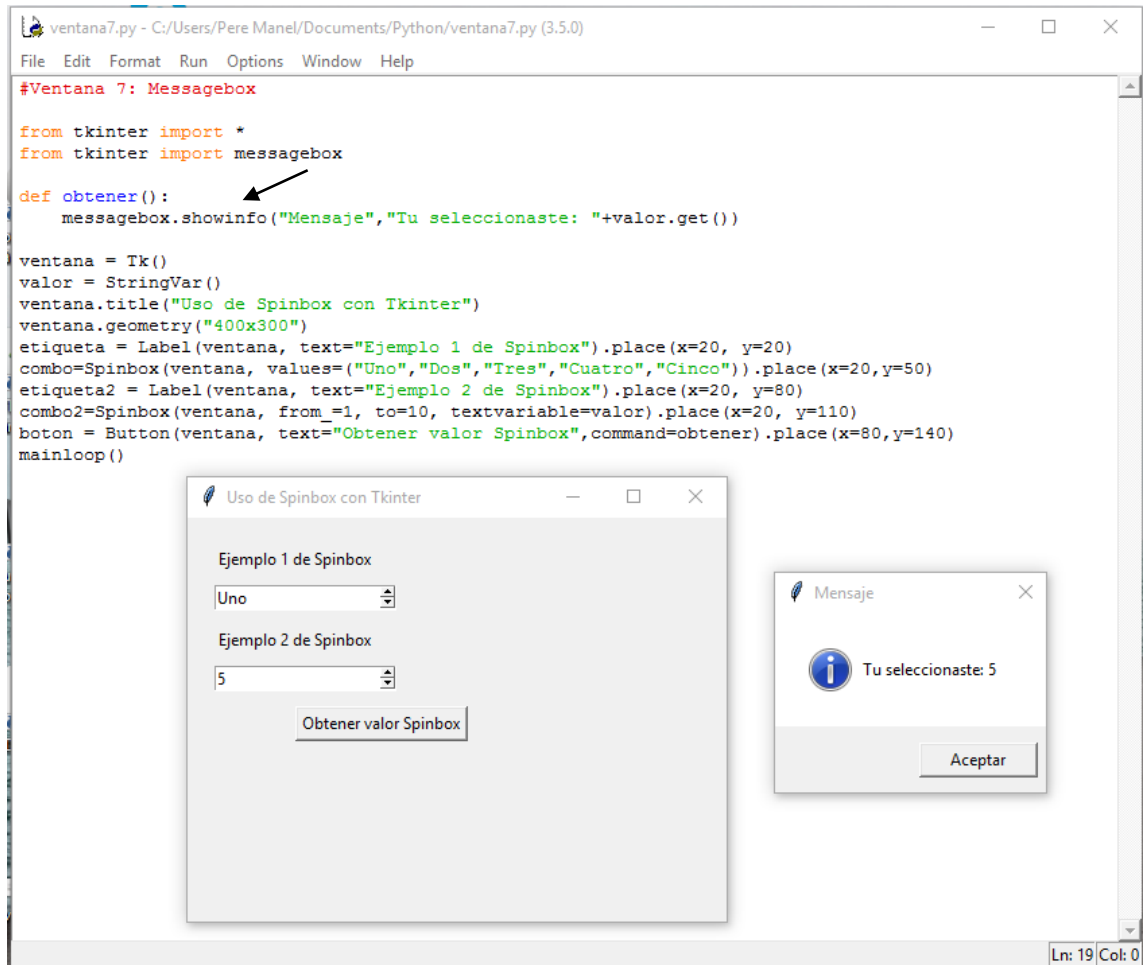
ventana = Tk()
opcion = IntVar()
num = IntVar()
ventana.title("Radiobttton en tkinter")
ventana.geometry("400x300")
etiquetal = Label(ventana, text="Escribe el número: ").place(x=20, y=20)
cajaNumero = Entry(ventana, textvariable=num).place(x=130, y=20)
etiqueta2 = Label(ventana, text="Escribe tu opción: ").place(x=20, y=50)
x10 = Radiobutton(ventana, text="X10", value=1, variable=opcion).place(x=20, y=80)
x20 = Radiobutton(ventana, text="X20", value=2, variable=opcion).place(x=70, y=80)
x30 = Radiobutton(ventana, text="X30", value=3, variable=opcion).place(x=120, y=80)
x40 = Radiobutton(ventana, text="X40", value=4, variable=opcion).place(x=20, y=110)
x50 = Radiobutton(ventana, text="X50", value=5, variable=opcion).place(x=70, y=110)
cuadrado = Radiobutton(ventana, text="Cuadrado", value=6, variable=opcion).place(x=120, y=110)
boton = Button(ventana, text="Realiza operación", command=operacion).place(x=20, y=140)
ventana.mainloop()
```

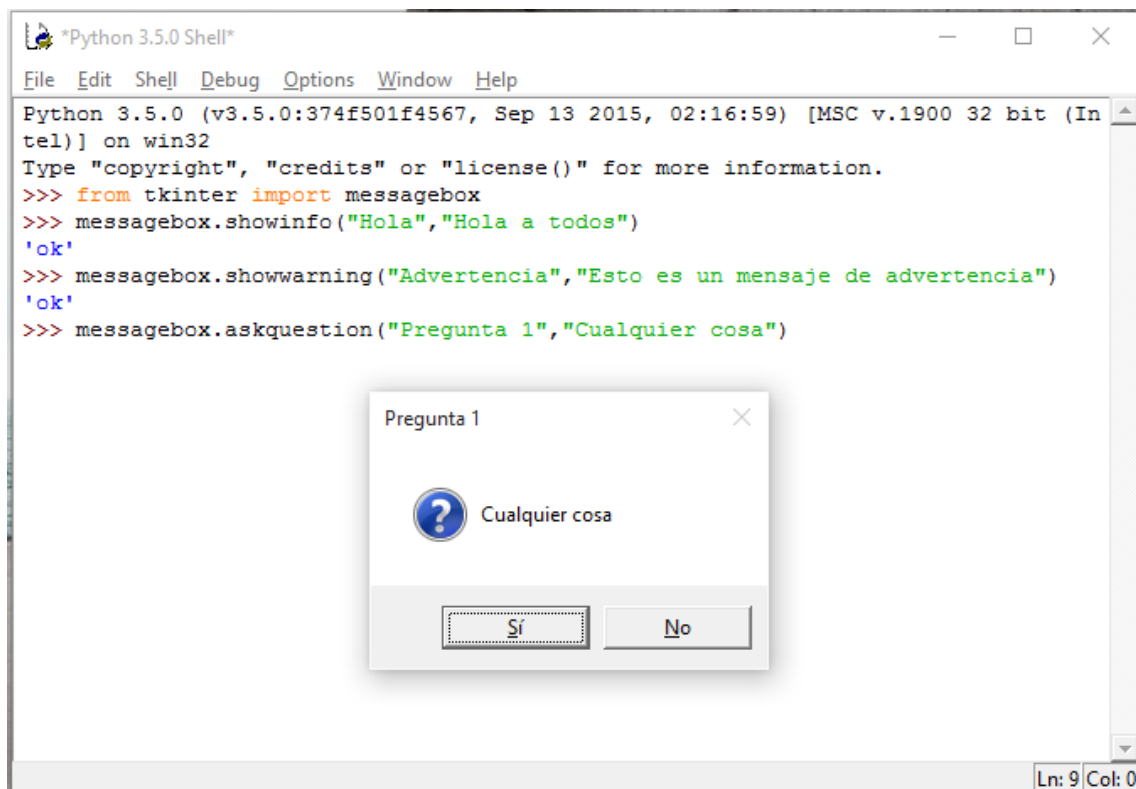
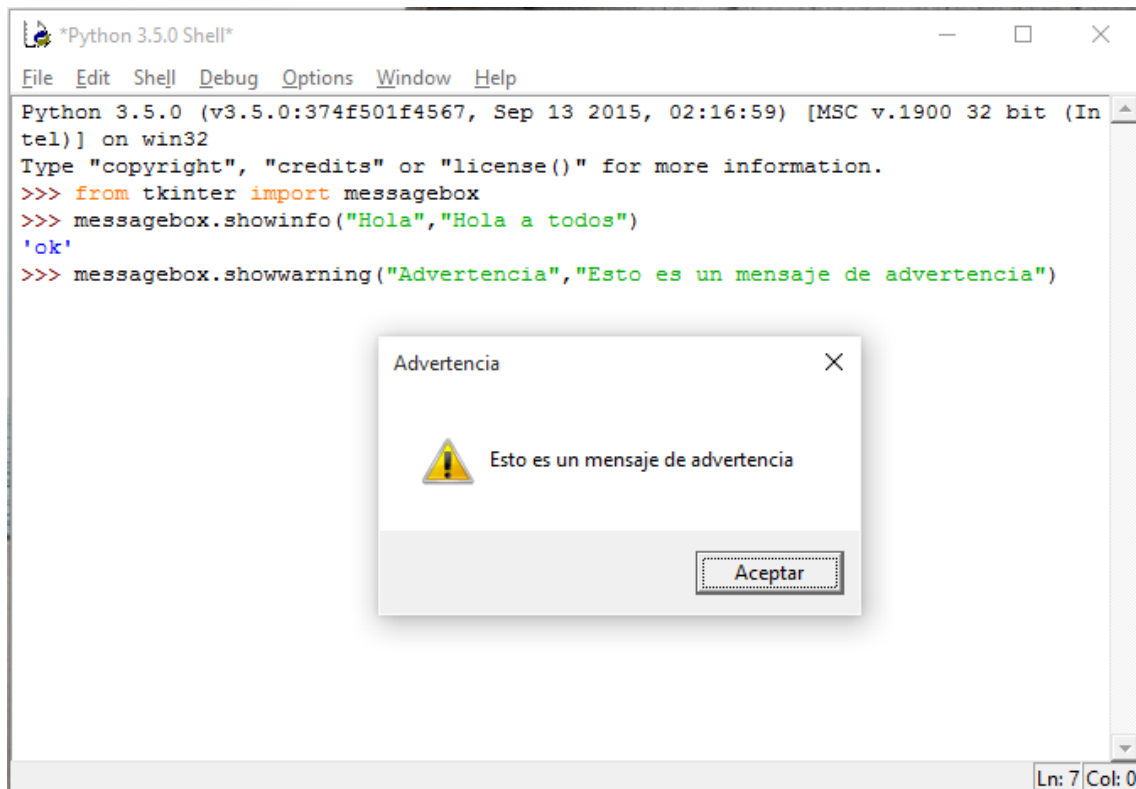


Python Tkinter Spinbox

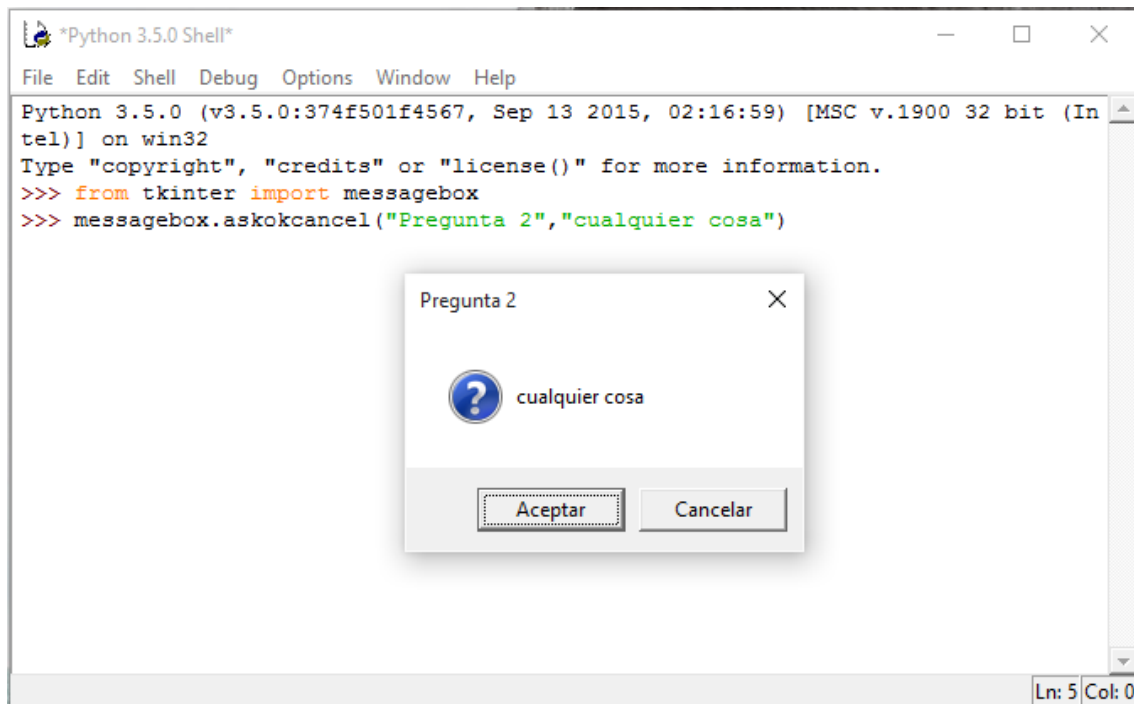


Python Tkinter MessageBox

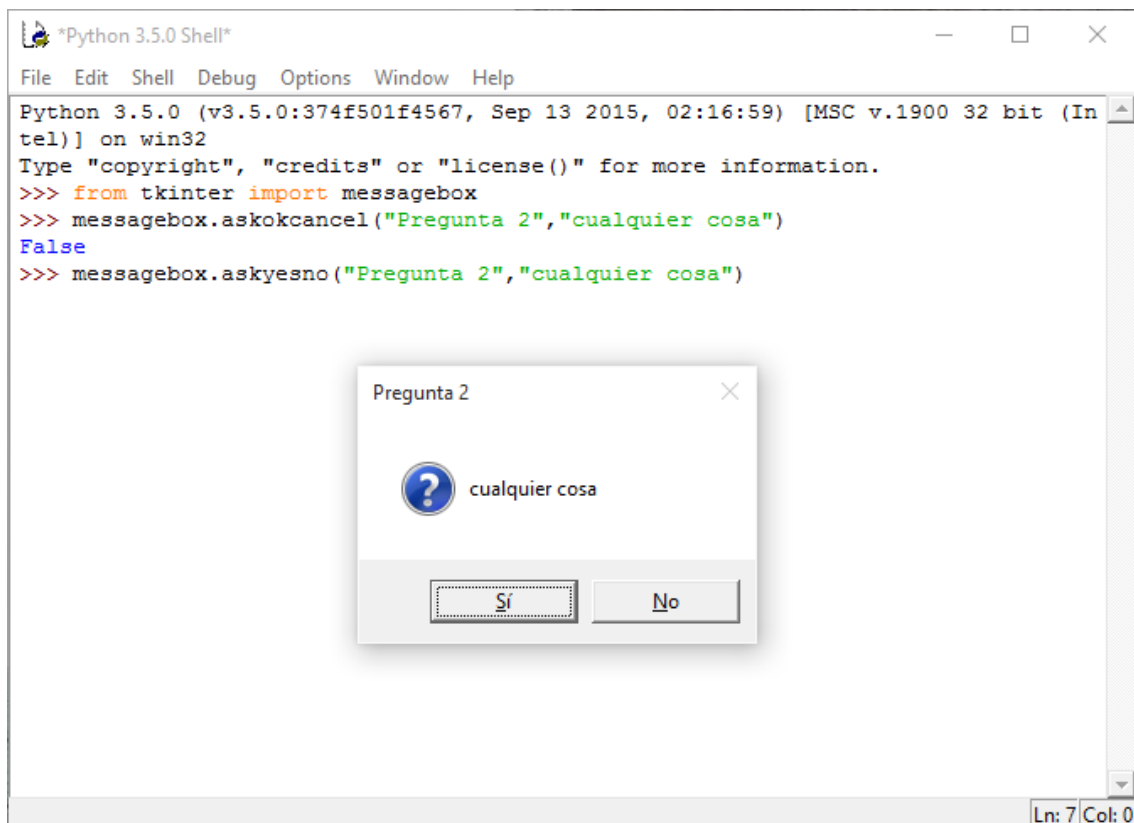




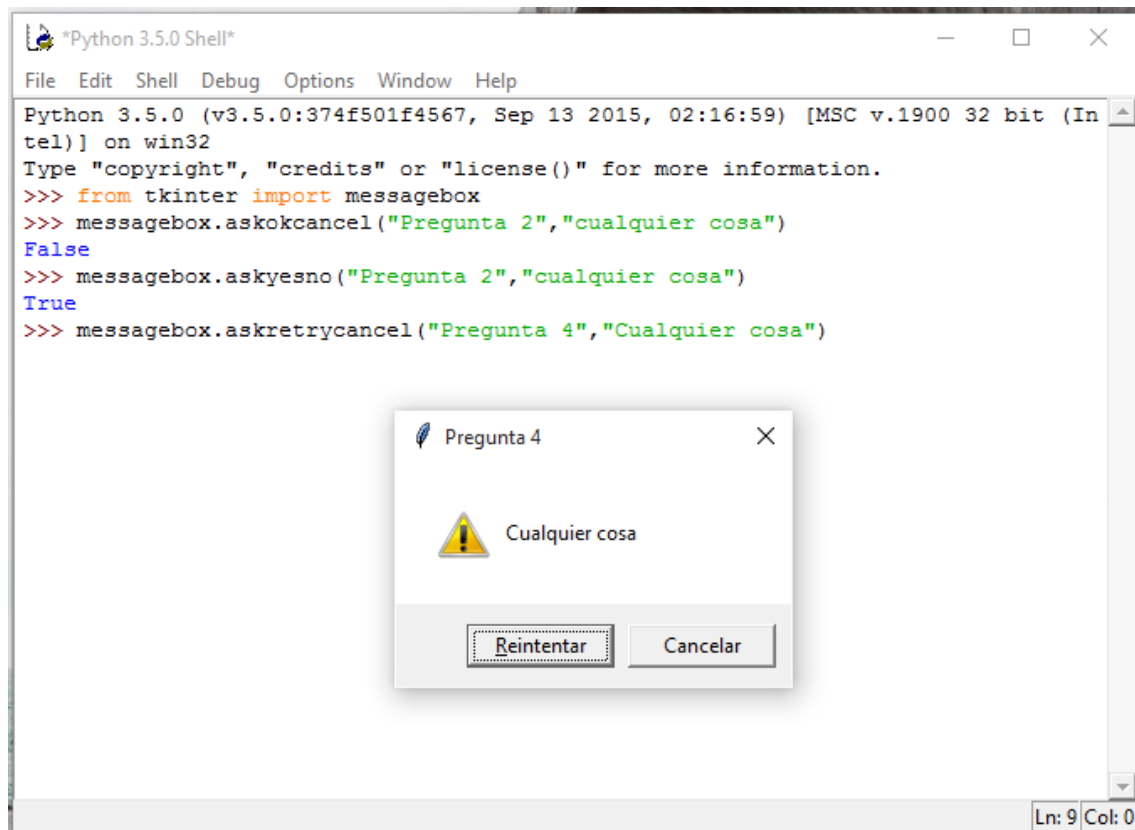
Según selecciones el botón 'Sí' o 'No' retornará una respuesta que será 'yes' o 'no' que luego nos servirá para controlar el programa.



Según selecciones el botón 'Aceptar' o 'Cancelar' retornará una respuesta que será 'True' o 'False' que luego nos servirá para controlar el programa.



A diferencia de askquestion que retorna un yes o un no el askyesno nos retorna True o False, si queremos trabajar con booleanos.



Reintentar retorna True y Cancelar retorna False.

Python Tkinter Agenda Parte I

```
ventana8.py - C:/Users/Pere Manel/Documents/Python/ventana8.py (3.5.0)
File Edit Format Run Options Window Help

#Agenda I

from tkinter import *
from tkinter import messagebox

lista = []

def guardar():
    print ("Hola")

def eliminar():
    print ("Hola")

ventana = Tk()
nombre = StringVar()
app = StringVar()
apm = StringVar()
correo = StringVar()
telefono = StringVar()
conteliminar = StringVar()
colorFondo = "#006"
colorLetra = "#FFF"
ventana.title("Agenda con archivos")
ventana.geometry("700x500")
ventana.configure(background = colorFondo)
etiquetaTitulo = Label(ventana, text="Agenda con Archivos",
    bg=colorFondo, fg=colorLetra).place(x=270,y=10)
etiquetaN = Label(ventana, text="Nombre", bg=colorFondo,
    fg=colorLetra).place(x=50, y=50)
cajaN = Entry(ventana, textvariable=nombre).place(x=150, y=50)
etiquetaApp = Label(ventana, text="Apellido Paterno", bg=colorFondo,
    fg=colorLetra).place(x=50, y=80)
cajaApp = Entry(ventana, textvariable=app).place(x=150, y=80)
etiquetaApm = Label(ventana, text="Apellido Materno", bg=colorFondo,
    fg=colorLetra).place(x=50, y=110)
cajaApm = Entry(ventana, textvariable=apm).place(x=150, y=110)
etiquetaT = Label(ventana, text="Teléfono", bg=colorFondo,
    fg=colorLetra).place(x=50, y=140)
cajaT = Entry(ventana, textvariable=telefono).place(x=150, y=140)
etiquetaC = Label(ventana, text="Correo", bg=colorFondo,
    fg=colorLetra).place(x=50, y=170)
cajaC = Entry(ventana, textvariable=correo).place(x=150, y=170)
etiquetaEliminar = Label(ventana, text="Teléfono: ", bg= colorFondo,
    fg=colorLetra).place(x=370, y=50)
spinTelefono = Spinbox(ventana, textvariable=conteliminar).place(x=450, y=50)
botoGuardar = Button(ventana, text="Guardar", command=guardar, bg="#009",
    fg="white").place(x=180, y=200)
botoEliminar = Button(ventana, text="Eliminar", command=eliminar, bg="#009",
    fg="white").place(x=490, y=80)

mainloop()

Ln: 51 Col: 0
```

Cuando ejecutemos este será el resultado:

Agenda con archivos

Agenda con Archivos

Nombre

Apellido Paterno

Apellido Materno

Teléfono

Correo

Guardar

Teléfono:

Eliminar

Python Tkinter Agenda Parte II

#Agenda II

```
from tkinter import *
from tkinter import messagebox
```

```
lista = []
```

```
def guardar():
    n = nombre.get()
    ap = app.get()
    am = apm.get()
    c = correo.get()
    t = telefono.get()
    lista.append(n+"$"+ap+"$"+am+"$"+t+"$"+c)
    escribirContacto()
    messagebox.showinfo("Guardado", "El contacto ha sido guardado en la agenda")
    nombre.set("")
    app.set("")
    apm.set("")
    correo.set("")
    telefono.set("")
    consultar()
```

```
def eliminar():
    eliminado = conteliminar.get()
    removido = False
    for elemento in lista:
        arreglo = elemento.split("$")
        if conteliminar.get() == arreglo[3]:
            lista.remove(elemento)
            removido = True
    escribirContacto()
    consultar()
    if removido:
        messagebox.showinfo("Eliminar", "Elemento eliminado "+eliminado)
```

```
def consultar():
    r = Text(ventana, width=80, height=15)
    lista.sort()
    valores = []
    r.insert(INSERT, "Nombre\tApellidos P\t\tApellido M\t\tTeléfono\t\tCorreo\n")
    for elemento in lista:
        arreglo = elemento.split("$")
        valores.append(arreglo[3])
        r.insert(INSERT, arreglo[0]+"\\t"+arreglo[1]+"\\t\\t"+
```

```

        arreglo[2]+"\t\t"+arreglo[3]+"\t\t"+arreglo[4]+"\t\n")
r.place(x=20,y=230)
spinTelefono = Spinbox(ventana, value=(valores),textvariable=conteliminar).place(x=450,
y=50)
if lista ==[]:
    spinTelefono = Spinbox(ventana, value=(valores)).place(x=450,y=50)
r.config(state=DISABLED)

```

```

def iniciarArchivo():
    archivo = open("ag.txt","a")
    archivo.close()

```

```

def cargar():
    archivo = open("ag.txt","r")
    linea = archivo.readline()
    if linea:
        while linea:
            if linea[-1]=='\n':
                linea = linea[:-1]
            lista.append(linea)
            linea = archivo.readline()
    archivo.close()

```

```

def escribirContacto():
    archivo = open("ag.txt","w")
    lista.sort()
    for elemento in lista:
        archivo.write(elemento+"\n")
    archivo.close()

```

```

ventana = Tk()
nombre = StringVar()
app = StringVar()
apm = StringVar()
correo = StringVar()
telefono = StringVar()
conteliminar = StringVar()
colorFondo = "#006"
colorLetra = "#FFF"
iniciarArchivo()
cargar()
consultar()
ventana.title("Agenda con archivos")
ventana.geometry("700x500")
ventana.configure(background = colorFondo)
etiquetaTitulo = Label(ventana, text="Agenda con Archivos",
    bg=colorFondo, fg=colorLetra).place(x=270,y=10)

```

```

etiquetaN = Label(ventana, text="Nombre", bg=colorFondo,
    fg=colorLetra).place(x=50, y=50)
cajaN = Entry(ventana, textvariable=nombre).place(x=150, y=50)
etiquetaApp = Label(ventana, text="Apellido Paterno", bg=colorFondo,
    fg=colorLetra).place(x=50, y=80)
cajaApp = Entry(ventana, textvariable=app).place(x=150, y=80)
etiquetaApm = Label(ventana, text="Apellido Materno", bg=colorFondo,
    fg=colorLetra).place(x=50, y=110)
cajaApm = Entry(ventana, textvariable=apm).place(x=150, y=110)
etiquetaT = Label(ventana, text="Teléfono", bg=colorFondo,
    fg=colorLetra).place(x=50, y=140)
cajaT = Entry(ventana, textvariable=telefono).place(x=150, y=140)
etiquetaC = Label(ventana, text="Correo", bg=colorFondo,
    fg=colorLetra).place(x=50, y=170)
cajaC = Entry(ventana, textvariable=correo).place(x=150, y=170)
etiquetaEliminar = Label(ventana, text="Teléfono: ", bg= colorFondo,
    fg=colorLetra).place(x=370, y=50)
spinTelefono = Spinbox(ventana, textvariable=conteliminar).place(x=450, y=50)
botoGuardar = Button(ventana, text="Guardar", command=guardar, bg="#009",
    fg="white").place(x=180, y=200)
botoEliminar = Button(ventana, text="Eliminar", command=eliminar, bg="#009",
    fg="white").place(x=490, y=80)

```

mainloop()

Este será el resultado final:

Agenda con Archivos

Nombre

Apellido Paterno

Apellido Materno

Teléfono

Correo

Nombre	Apellidos P	Apellido M	Teléfono	Correo
Pere M	Verdugo	Zamora	678061103	pmverdugo@gmail.com
Piedad	Fernandez	Soriano	937153452	piedadfs@gmail.com

Ejercicio práctico

Con todo lo aprendido hasta ahora vamos a realizar un programa que os permite realizar el control de los libros que tengo que leer y los que me quedan pendientes de leer.

Esta será la ventana principal:

Relación de libros que tengo que leer

Título: Añadir libro

Autor:

Editorial:

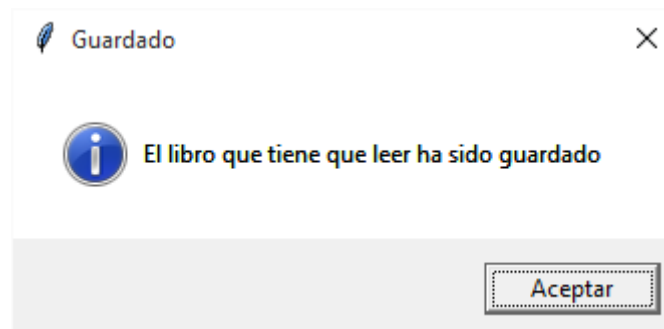
Nº de páginas:

Fecha límite:

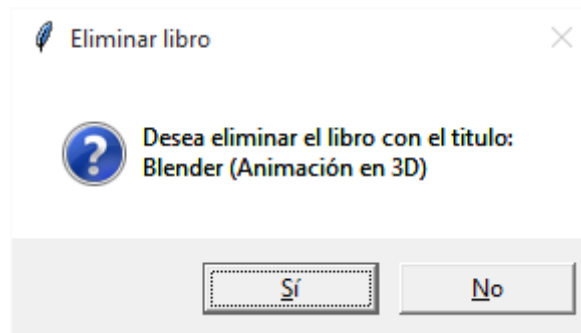
Título del libro
Blender (Animación en 3D) Autor:Pere Manel Editorial:Anaya Nº de páginas:450 Fecha límite:18-12-2015
FileMaker Pro 14 (Mi primera base de datos) Autor:Pere Manel Editorial:AppStore Nº de páginas:182 Fecha límite:30-10-2015
Introducción Windwos 10 Autor:Pere Manel Editorial:Ra-Ma Nº de páginas:250 Fecha límite:12-11-2015
Microsotf Word (Tratamiento de textos) Autor:Pere Manel Editorial:Anaya Nº de páginas:180 Fecha límite:15-12-2015
Programación en C

Título leído: Libro ya leído

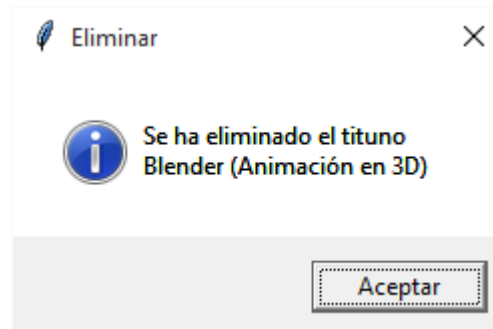
Cuando agregamos un nuevo libro nos tiene que salir el siguiente mensaje:



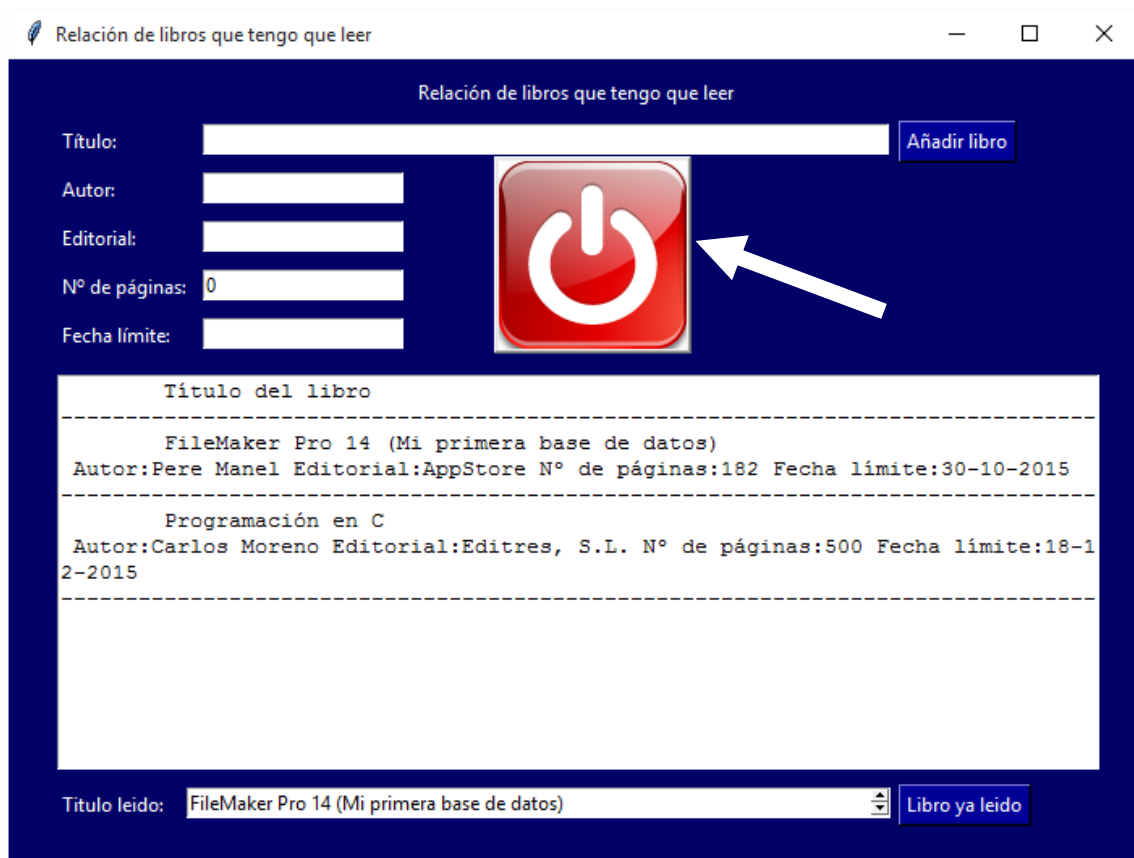
Ahora vamos a seleccionar un libro como se muestra con la fecha y seleccionaremos el botón "Libro ya leído".



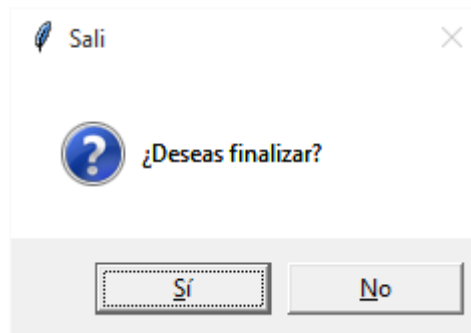
Saldrá una ventana que nos pedirá información, contestamos Sí.



Con otra ventana nos muestra que dicho libro ya se ha eliminado.



Para salir de la aplicación seleccionaremos el botón Salir.



Saldrá una ventana que nos pedirá confirmación, si seleccionamos Sí el programa se cerrará.

El código del programa:

```
#Este proyecto me permitirá llevar una relación de los libros
#que tengo que leer y saber en cualquier momento los libros
#que tengo pendientes por leer.
```

```
from tkinter import *
from tkinter import messagebox
```

```
libros = []
```

Importamos tkinter y tikenter messagebox.

Definimos en array llamada libros =...[]

Definimos la función anadir():

```
def anadir():
    t = titulo.get()
    a = autor.get()
    e = editorial.get()
    np = NdePaginas.get()
    fl = fechalimite.get()
    libros.append(t+"$"+a+"$"+e+"$"+str(np)+"$"+fl)
    escribirLibro()
    messagebox.showinfo("Guardado", "El libro que tiene que leer ha sido guardado")
    titulo.set("")
    autor.set("")
    editorial.set("")
    NdePaginas.set("")
    fechalimite.set("")
    consultar()
```

Definimos la función escribirLibro():

```
def escribirLibro():
    archivo=open("biblioteca.txt", "w")
    libros.sort()
    for elemento in libros:
        archivo.write(elemento+"\n")
    archivo.close()
```

Definimos la función eliminarLibro():

```

def eliminarLibro():
    eliminado=eliminarLibro.get()
    removido=False
    for elemento in libros:
        arreglo = elemento.split("$")
        if eliminarLibro.get()==arreglo[0]:
            respuesta=messagebox.askyesno("Eliminar libro",
            ["Desea eliminar el libro con el titulo:\n"+eliminado])
            if respuesta:
                libros.remove(elemento)
                removido = True
                escribirLibro()
                consultar()
                respuesta=""
    if removido:
        messagebox.showinfo("Eliminar","Se ha eliminado el tituno\n"+eliminado)

```

Definimos la función salir():

```

def salir():
    sa=messagebox.askyesno("Sali","¿Deseas finalizar?")
    if sa==True:
        quit()

```

Definimos la función iniciarArchivo():

```

def iniciarArchivo():
    archivo = open("biblioteca.txt","a")
    archivo.close()

```

Definimos la función cargar():

```

def cargar():
    archivo=open("biblioteca.txt","r")
    linea = archivo.readline()
    if linea:
        while linea:
            if linea[-1]=='\n':
                linea=linea[:-1]
            libros.append(linea)
            linea=archivo.readline()
        archivo.close()

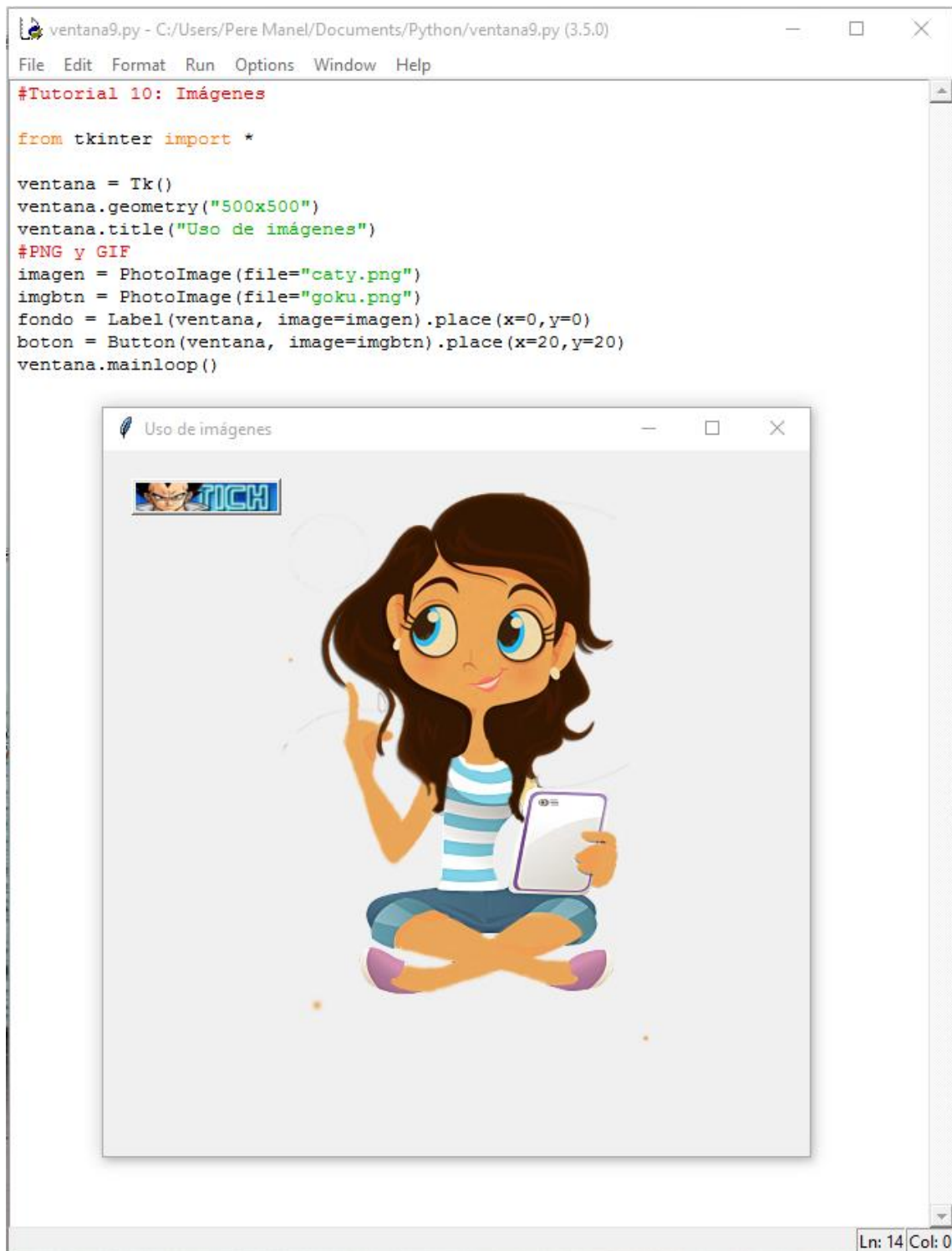
```

Definimos la función consultar()

La parte principal del programa:

Nota: la \ que verás al final de algunas líneas es para decir que la siguiente línea es la continuación de la anterior, nos permite partir líneas que son muy largas pero que no modifique la ejecución del programa.

Python Tkinter Imágenes

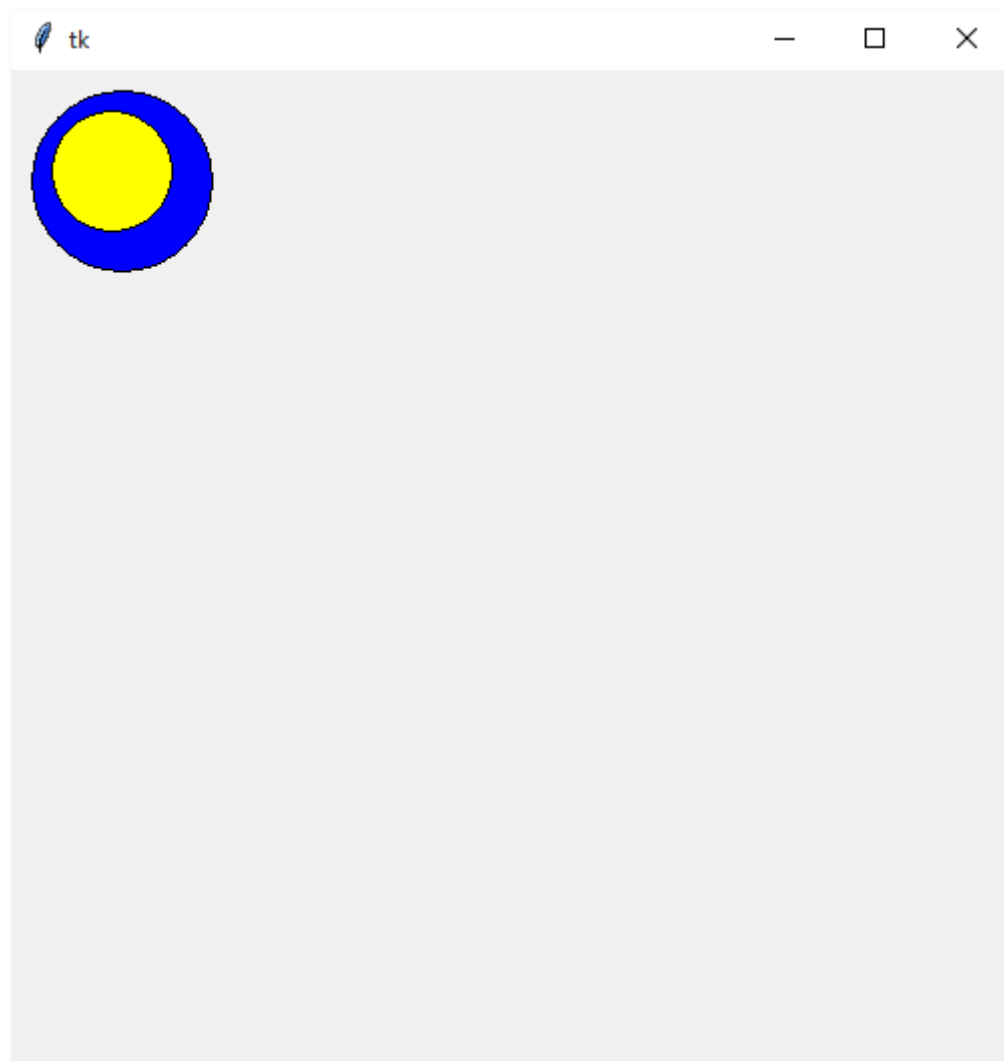


Python Tkinter Canvas Parte 1

```
*ventana10.py - C:/Users/Pere Manel/D...  -  □  ×
File Edit Format Run Options Window Help
#Tutorial 10: Canvas Parte 1

from tkinter import *

ventana = Tk()
c = Canvas(ventana, width=200, height=200)
ventana.geometry("500x500")
#rgb nivel rojo, nivel verde, nivel azul
#color = "#009"
#c.create_oval(10,10,100,100,fill=color)
#c.create_oval(10,10,100,100,fill="blue")
#c.create_oval(20,20,80,80,fill="yellow")
c.place(x=0,y=0)
cr
ventana.mainloop()
|
Ln: 16 Col: 0
```

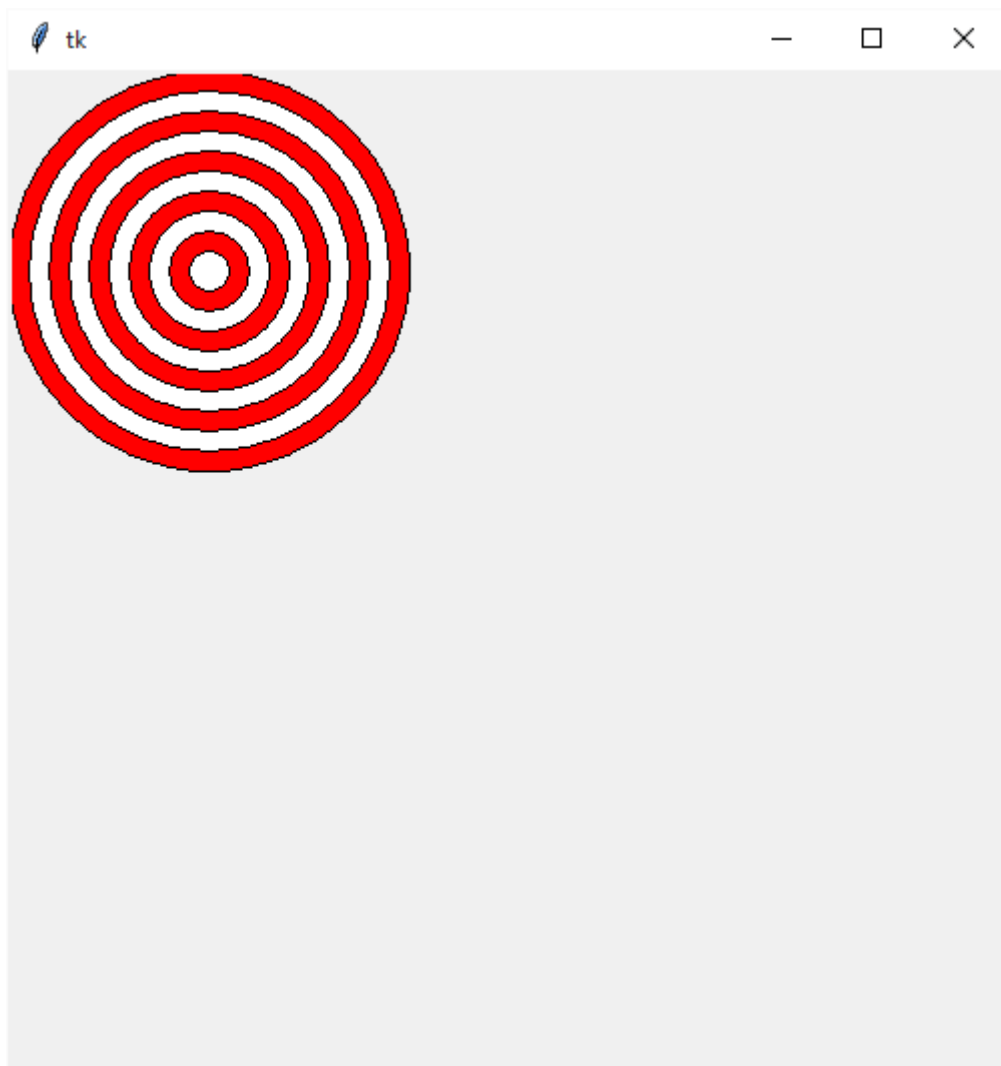


```
ventana10.py - C:/Users/Pere Manel/Documents/Python/ventana10.py (3.5.0)
File Edit Format Run Options Window Help
#Tutorial 10: Canvas Parte 1

from tkinter import *

ventana = Tk()
c = Canvas(ventana, width=200, height=200)
ventana.geometry("500x500")
for i in range (0,10):
    if i%2 ==0:
        c.create_oval(i*10,i*10,200-(i*10),200-(i*10),fill="red")
    else:
        c.create_oval(i*10,i*10,200-(i*10),200-(i*10),fill="white")
c.place(x=0,y=0)
ventana.mainloop()
```

Ln: 8 Col: 20

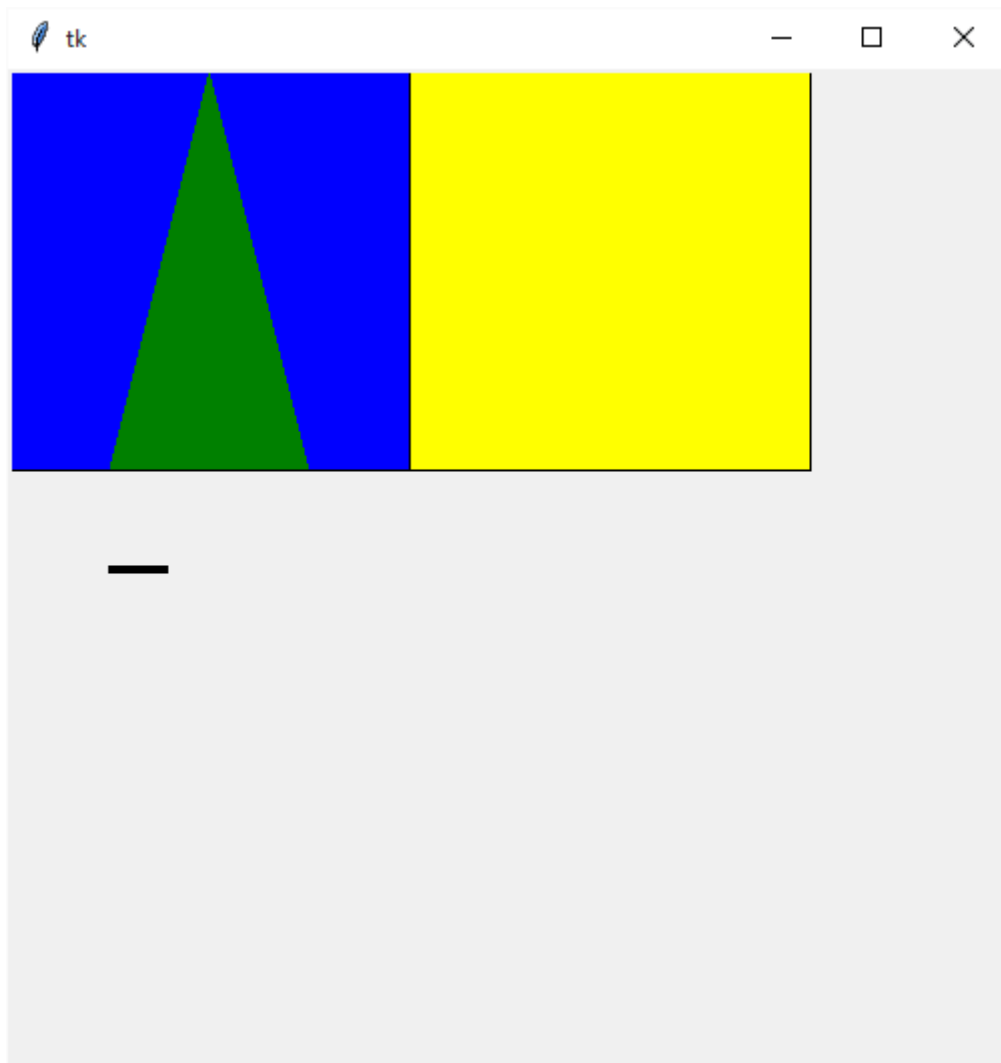


Python Tkinter Canvas Parte 2

```
ventana11.py - C:/Users/Pere Manel/Documents/Python/ventana11.py (3.5.0)
File Edit Format Run Options Window Help
#Tutorial 10: Canvas Parte 2
from tkinter import *

ventana =Tk()
c = Canvas(ventana, width=500, height=500)
ventana.geometry("500x500")
c.place(x=0,y=0)
c.create_rectangle(0,0,200,200,fill="blue")
c.create_rectangle(200,0,400,200,fill="yellow")
c.create_line(50,250,80,250,width=4.0) #Por defecto 1.0
c.create_polygon(100,0,150,200,50,200,fill="green")
ventana.mainloop()
```

Ln: 9 Col: 0



En la siguiente página web podrás encontrar una paleta de colores:

http://es.bleach.wikia.com/wiki/Ayuda:Paleta_de_colores

The screenshot shows the 'Ayuda:Paleta de colores' page on the Bleach Wiki. The page features a color palette with two main sections: 'Colores Rojos' (Red Colors) and 'Colores Verdes' (Green Colors). Each section contains a list of named colors with their corresponding hex and rgb values. The 'Colores Rojos' section includes colors like IndianRed, LightCoral, Salmon, DarkSalmon, LightSalmon, Crimson, Red, FireBrick, DarkRed, and Colores Rosas. The 'Colores Verdes' section includes colors like GreenYellow, Chartreuse, LawnGreen, Lime, LimeGreen, PaleGreen, LightGreen, MediumSpringGreen, SpringGreen, MediumSeaGreen, SeaGreen, ForestGreen, Green, and DarkGreen. The page also has a sidebar with 'Actividad Reciente' (Recent Activity) and a top navigation bar with 'Wiki', 'Bleach', 'Media', 'Comunidad', and 'Acerca del Wiki'.

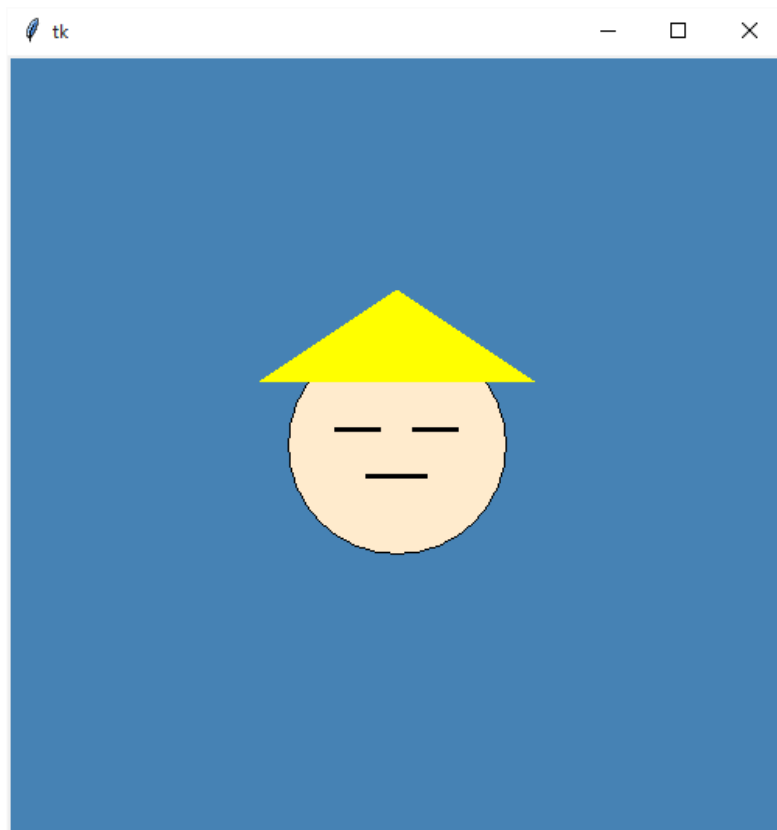
Colores Rojos			Colores Verdes		
IndianRed	#CD5C5C	rgb(205,92,92)	GreenYellow	#ADFF2F	rgb(173,255,47)
LightCoral	#F08080	rgb(240,128,128)	Chartreuse	#7FFF00	rgb(127,255,0)
Salmon	#FA8072	rgb(250,128,114)	LawnGreen	#7CFC00	rgb(124,252,0)
DarkSalmon	#E9967A	rgb(233,150,122)	Lime	#00FF00	rgb(0,255,0)
LightSalmon	#FFA07A	rgb(255,160,122)	LimeGreen	#32CD32	rgb(50,205,50)
Crimson	#DC143C	rgb(220,20,60)	PaleGreen	#98FB98	rgb(152,251,152)
Red	#FF0000	rgb(255,0,0)	LightGreen	#90EE90	rgb(144,238,144)
FireBrick	#B22222	rgb(178,34,34)	MediumSpringGreen	#00FA9A	rgb(0,250,154)
DarkRed	#8B0000	rgb(139,0,0)	SpringGreen	#00FF7F	rgb(0,255,127)
Colores Rosas			MediumSeaGreen	#3CB371	rgb(60,179,113)
Pink	#FFC0CB	rgb(255,192,203)	SeaGreen	#2E8B57	rgb(46,139,87)
LightPink	#FFB6C1	rgb(255,182,193)	ForestGreen	#228B22	rgb(34,139,34)
HotPink	#FF69B4	rgb(255,105,180)	Green	#008000	rgb(0,128,0)
DeepPink	#FF1493	rgb(255,20,147)	DarkGreen	#006400	rgb(0,100,0)
MediumVioletRed	#C71585	rgb(199,21,133)	YellowGreen	#9ACD32	rgb(154,205,50)

```
ventana11.py - C:\Users\Pere Mane\Documents\Python\ven...
File Edit Format Run Options Window Help

#Tutorial 10: Canvas Parte 2
from tkinter import *

ventana = Tk()
c = Canvas(ventana, width=500, height=500)
ventana.geometry("500x500")
c.place(x=0,y=0)
c.create_rectangle(0,0,500,500, fill="#4682B4")
c.create_oval(180,180,320,320,fill="#FFEB3D")
c.create_polygon(160,210,340,210,250,150, fill="yellow")
c.create_line(210,240,240,240, width=3.0)
c.create_line(260,240,290,240, width=3.0)
c.create_line(230,270,270,270, width=3.0)
ventana.mainloop()

Ln: 15 Col: 0
```



Pythoon Tkinter Canvas Parte 3

```
ventana11.py - C:\Users\Pere Manel\Documents\Python\ventana11.py (3.5.0)
File Edit Format Run Options Window Help
#Tutorial 11: Canvas Parte 3 Intesección con widgets
from tkinter import *

def sombrero():
    c.create_polygon(160,210,340,210,250,150, fill=color.get())

def fondo():
    c.create_rectangle(0,0,500,500, fill=color.get())
    c.create_oval(180,180,320,320,fill="#FFEB CD")
    c.create_polygon(160,210,340,210,250,150, fill="yellow")
    c.create_line(210,240,240,240, width=3.0)
    c.create_line(260,240,290,240, width=3.0)
    c.create_line(230,270,270,270, width=3.0)

ventana =Tk()
color = StringVar()
color1 = StringVar()
c = Canvas(ventana, width=500, height=500)
ventana.geometry("500x500")
c.place(x=0,y=0)
c.create_rectangle(0,0,500,500, fill="#4682B4")
c.create_oval(180,180,320,320,fill="#FFEB CD")
c.create_polygon(160,210,340,210,250,150, fill="yellow")
c.create_line(210,240,240,240, width=3.0)
c.create_line(260,240,290,240, width=3.0)
c.create_line(230,270,270,270, width=3.0)
combo = Spinbox(ventana, values=("blue", "yellow", "red", "black","white"),
                textvariable=color).place(x=20, y=20)
boton = Button(ventana, text="cambiar sombrero", command=sombrero).place(x=165,y=18)
boton1 = Button(ventana, text="cambiar fondo", command=fondo).place(x=285,y=18)
ventana.mainloop()

Ln: 28 Col: 16
```

