

```
In [4]: from tkinter import *

root = Tk()

#-----situar al final (pq acaba el programa)

root.mainloop()
```

```
In [5]: from tkinter import *

root = Tk()

Label (root, text ="Hola Mundo").pack()
Label (root, text ="com vas?").pack()

#-----situar al final (pq acaba el programa)

root.mainloop()
```

```
In [7]: # https://docs.hektorprofe.net/python/interfaces-graficas-con-tkinter/widget-frame-

from tkinter import *
root = Tk()

# Hijo de root, no ocurre nada
frame = Frame(root)

# Empaqueta el frame en la raíz
frame.pack()

# También podemos añadir la configuración al crear el frame:
# Frame(root, width=480,height=320)

# Como no tenemos ningún elemento dentro del frame,
# no tiene tamaño y aparece ocupando lo mínimo posible, 0*0 px

# Color de fondo, background
frame.config(bg="lightblue")

# Podemos establecer un tamaño,
# la raíz se adapta al frame que contiene
frame.config(width=380,height=220)

root.mainloop()
```

```
In [8]: from tkinter import *

# Configuración de la raíz
root = Tk()
root.title("Hola mundo")
root.resizable(1,1)
root.iconbitmap('hola.ico')

frame = Frame(root, width=480, height=320)
frame.pack(fill='both', expand=1)
frame.config(cursor="pirate")
frame.config(bg="lightblue")
frame.config(bd=25)
frame.config(relief="sunken")

root.config(cursor="arrow")
root.config(bg="blue")
root.config(bd=15)
root.config(relief="ridge")

# Finalmente bucle de la aplicación
root.mainloop()
```

```
In [16]: # https://docs.hektorprofe.net/python/interfaces-graficas-con-tkinter/widget-Label-

from tkinter import *

root = Tk()

Label(root, text="¡Hola Mundo!").pack(anchor=NW)
Label(root, text="¡Otra etiqueta!").pack(anchor=CENTER)
Label(root, text="¡Última etiqueta!").pack(anchor=SE)

label = Label(root, text="¡Otra etiqueta!")
label.pack(anchor=CENTER)
label.config(fg="blue",      # Foreground
             bg="green",    # Background
             font=("Verdana",24))

texto = StringVar()
texto.set("Un nuevo texto por aquí")
label.config(textvariable=texto) # añadimos una variable de texto

imagen = PhotoImage(file=".\\img\\imagen.gif")
Label(root, image=imagen, bd=0).pack()

# Label(root, text="¡Hola Mundo!").pack()
# Label(root, text="¡Otra etiqueta!").pack()
# Label(root, text="¡Última etiqueta!").pack()

root.mainloop()
```

```
In [1]: #Widget Entry (Texto corto)

from tkinter import *
root = Tk()

entry = Entry(root)
entry.pack()

entry = Entry(root)
entry.pack(side=RIGHT)

label = Label(root, text="Nombre")
label .pack(side=LEFT)

entry2 = Entry(root)
entry2.pack(side=RIGHT)

label2 = Label(root, text="Apellidos")
label2 .pack(side=LEFT)

root.mainloop()
```

```
In [2]: from tkinter import *

# Configuración de la raíz
root = Tk()

label = Label(root, text="Nombre muy largo")
label.grid(row=0, column=0, sticky="w", padx=5, pady=5)

entry = Entry(root)
entry.grid(row=0, column=1, padx=5, pady=5)
entry.config(justify="right", state="normal")

label2 = Label(root, text="Contraseña")
label2.grid(row=1, column=0, sticky="w", padx=5, pady=5)

entry2 = Entry(root)
entry2.grid(row=1, column=1, padx=5, pady=5)
entry2.config(justify="center", show="?")

# Finalmente bucle de la aplicación
root.mainloop()
```

```
In [29]: #Widget Text (Texto largo)

from tkinter import *
root = Tk()

texto = Text(root)
texto.pack()

texto.config(width=30, height=10)
texto.config(font=("Consolas",12), selectbackground="red", padx=5, pady=5)

root.mainloop()
```

In [31]: `#Widget Button (Botón)`

```
# Podemos crearlos y empacarlos en una línea
Button(root, text="Clícame").pack()
```

In [32]: `# Sin más complicación aquí tenemos un botón, ¿pero la gracia es que haga algo no?`
`# Para hacerlo tenemos que crear un parámetro command que ejecutará un código cuando`
`# Este código se lo pasaremos en forma de función.`
`# Si tenemos la ventana de comandos con la extensión .py podríamos hacer lo siguiente`

```
from tkinter import *

# Definimos una función a ejecutar al clic el botón
def hola():
    print("Hola mundo!")

root = Tk()

# Enlazamos la función a la acción del botón
Button(root, text="Clícame", command=hola).pack()

root.mainloop()
```

Hola mundo!

```
In [34]: from tkinter import *

# Funciones backend
def borrar():
    n1.set('')
    n2.set('')

def sumar():
    r.set( float( n1.get() ) + float(n2.get() ) )
    borrar()
def restar():
    r.set( float( n1.get() ) - float(n2.get() ) )
    borrar()

def multiplicar():
    r.set( float( n1.get() ) * float(n2.get() ) )
    borrar()

# Estructura del formulario
root = Tk()
root.config(bd=15) # borde exterior de 15 píxeles, queda mejor

# Tres StringVar para manejar los números y el resultado
n1 = StringVar()
n2 = StringVar()
r = StringVar()

Label(root, text="Numero 1").pack()
Entry(root, justify=CENTER, textvariable=n1).pack()

Label(root, text="\nNumero 2").pack()
Entry(root, justify=CENTER, textvariable=n2).pack()

Label(root, text="\nResultado").pack()
Entry(root, justify=CENTER, state=DISABLED, textvariable=r).pack()

Label(root).pack() # Separador

#Button(root, text="Sumar", command=sumar).pack()
Button(root, text="Sumar", command=sumar).pack(side=LEFT)
Button(root, text="Restar", command=restar).pack(side=LEFT)
Button(root, text="Multiplicar", command=multiplicar).pack(side=LEFT)

root.mainloop()
```

```
In [36]: #Widget Radiobutton (Radial)

from tkinter import *

def selec():
    monitor.config(text = "Opción {}".format(opcion.get() ) )

def reset():
    opcion.set(None)          # Reiniciamos el seleccionable
    monitor.config(text='')   # Reiniciamos la etiqueta

root = Tk()
root.config(bd=15)

opcion = IntVar() # Como StrinVar pero en entero

Radiobutton(root, text="Opción 1", variable=opcion,
             value=1, command=selec).pack()
Radiobutton(root, text="Opción 2", variable=opcion,
             value=2, command=selec).pack()
Radiobutton(root, text="Opción 3", variable=opcion,
             value=3, command=selec).pack()

Button(root, text="Reiniciar", command=reset).pack()

monitor = Label(root)
monitor.pack()

root.mainloop()
```

```
In [37]: from tkinter import *

def seleccionar():
    monitor.config(text="{}".format(opcion.get()))

def reset():
    opcion.set(None)
    monitor.config(text="")

# Configuración de la raíz
root = Tk()

opcion = IntVar()

Radiobutton(root, text="Opción 1", variable=opcion,
             value=1, command=seleccionar).pack()
Radiobutton(root, text="Opción 2", variable=opcion,
             value=2, command=seleccionar).pack()
Radiobutton(root, text="Opción 3", variable=opcion,
             value=3, command=seleccionar).pack()

monitor = Label(root)
monitor.pack()

Button(root, text="Reiniciar", command=reset).pack()

# Finalmente bucle de la aplicación
root.mainloop()
```

```
In [38]: #Widget Checkbutton (Seleccionable)

from tkinter import *

root = Tk()
root.config(bd=15)

leche = IntVar()      # 1 si, 0 no
azucar = IntVar()     # 1 si, 0 no

Label(root, text="¿Cómo quieres el café?").pack()
Checkbutton(root, text="Con leche", variable=leche,
             onvalue=1, offvalue=0).pack()
Checkbutton(root, text="Con azúcar", variable=leche,
             onvalue=1, offvalue=0).pack()

root.mainloop()
```

```
In [41]: from tkinter import *

root = Tk()
root.config(bd=15)

leche = IntVar()
azucar = IntVar()

imagen = PhotoImage(file=".\\img\\imagen.gif")
Label(root, image=imagen).pack(side=LEFT)

frame = Frame(root).pack(side=RIGHT)
Label(frame, text="¿Cómo quieres el café?\n").pack(anchor=W)
Checkbutton(frame, text="Con leche", variable=leche,
             onvalue=1, offvalue=0).pack(anchor=W)
Checkbutton(frame, text="Con azúcar", variable=azucar,
             onvalue=1, offvalue=0).pack(anchor=W)

root.mainloop()
```



```
In [19]: from tkinter import *

def seleccionar():
    cadena = ""
    if (leche.get()):
        cadena += "Con leche"
    else:
        cadena += "Sin leche"

    if (azucar.get()):
        cadena += " y con azúcar"
    else:
        cadena += " y sin azúcar"

    monitor.config(text=cadena)

# Configuración de la raíz
root = Tk()
root.title("Cafetería")
root.config(bd=15)

leche = IntVar()    # 1 si, 0 no
azucar = IntVar()   # 1 si, 0 no

imagen = PhotoImage(file="imagen.gif")
Label(root, image=imagen).pack(side="left")

frame = Frame(root)
frame.pack(side="left")

Label(frame, text="¿Cómo quieres el café?").pack(anchor="w")
Checkbutton(frame, text="Con leche", variable=leche, onvalue=1,
             offvalue=0, command=seleccionar).pack(anchor="w")
Checkbutton(frame, text="Con azúcar", variable=azucar, onvalue=1,
             offvalue=0, command=seleccionar).pack(anchor="w")

monitor = Label(frame)
monitor.pack()

# Finalmente bucle de la aplicación
root.mainloop()
```

```
In [ ]: #Widget Menu (Menú)

from tkinter import *

# Configuración de la raíz
root = Tk()

menubar = Menu(root)
root.config(menu=menubar)

filemenu = Menu(menubar, tearoff=0)
filemenu.add_command(label="Nuevo")
filemenu.add_command(label="Abrir")
filemenu.add_command(label="Guardar")
filemenu.add_command(label="Cerrar")
filemenu.add_separator()
filemenu.add_command(label="Salir", command=root.destroy)

editmenu = Menu(menubar, tearoff=0)
editmenu.add_command(label="Cortar")
editmenu.add_command(label="Copiar")
editmenu.add_command(label="Pegar")

helpmenu = Menu(menubar, tearoff=0)
helpmenu.add_command(label="Ayuda")
helpmenu.add_separator()
helpmenu.add_command(label="Acerca de...")

menubar.add_cascade(label="Archivo", menu=filemenu)
menubar.add_cascade(label="Editar", menu=editmenu)
menubar.add_cascade(label="Ayuda", menu=helpmenu)

# Finalmente bucle de la aplicación
root.mainloop()
```

```
In [ ]: #Dialogs (Diálogos)

#ShowInfo
#Sirve para mostrar un diálogo de más información:

from tkinter import *
from tkinter import messagebox as MessageBox

def test():
    MessageBox.showinfo("Hola!", "Hola mundo") # título, mensaje

root = Tk()

Button(root, text = "Clícame", command=test).pack()

root.mainloop()
```

```
In [ ]: # ShowWarning
# Sirve para mostrar un diálogo con un mensaje de alerta:

MessageBox.showwarning("Alerta",
    "Sección sólo para administradores.")
```

```
In [ ]: # ShowError
# Sirve para mostrar un diálogo con un mensaje de error:

MessageBox.showerror("Error",
    "Ha ocurrido un error inesperado.")
```

```
In [ ]: # AskQuestion
# Sirve para mostrar un diálogo con una pregunta de Sí/No al usuario:

resultado = MessageBox.askquestion("Salir", "¿Está seguro que desea salir sin guard

if resultado == "yes":
    root.destroy() # Destruir, alternativa a quit
```

```
In [ ]: # AskOkCancel
# Sirve para mostrar un diálogo con una pregunta de Ok/Cancelar al usuario:

resultado = MessageBox.askokcancel("Salir",
    "¿Sobreescribir fichero actual?")

# if resultado == True:
# # Hacer algo
```

```
In [22]: # AskRetryCancel
# Sirve para mostrar un diálogo con una pregunta de Reintentar/Cancelar al usuario:

resultado = MessageBox.askretrycancel("Reintentar",
    "No se puede conectar")

# if resultado == True:
# # Hacer algo
```

```
In [24]: # AskColor
# Imaginad que queréis crear una aplicación y necesitáis que el usuario seleccione
# Pues para este caso tenemos una clase llamada colorchooser:

from tkinter import colorchooser as ColorChooser

def test():
    color = ColorChooser.askcolor(title="Elige un color")
    print(color)
```

```
In [26]: # AskOpenFile
# Y por último, un vistazo a la clase FileDialog,
# que nos permite realizar varias tareas como conseguir la ruta de un fichero para

from tkinter import filedialog as FileDialog

def test():
    fichero = FileDialog.askopenfilename(title="Abrir un fichero")
    print(fichero)
```

```
In [28]: # Sin embargo lo más interesante es que también podemos establecer otras opciones,
# por ejemplo un directorio inicial y un filtro de extensiones.
# Pero debéis tener cuidado con el directorio, ya que no todos los sistemas operati

file = FileDialog.askopenfilename(
    initialdir="C:",
    filetypes=(
        ("Ficheros de texto", "*.txt"),
        ("Todos los ficheros", "*.*")
    ),
    title = "Abrir un fichero."
)
```

```
In [29]: # AskSaveAsFile
# Por contra, también podemos hacer lo mismo para buscar un directorio donde guarda
# Aunque la lógica tras de esto ya es más compleja, el proceso es muy similar:

def test():
    ruta = FileDialog.asksaveasfile(title="Guardar un fichero")
    print(fichero )

#Este proceso equivaldría a abrir un fichero de texto plano con open(ruta,'w'),
#así que cuidado si abrís un fichero ya existente porque lo dejaréis sin nada, es d
```

```
In [30]: # En cuanto a los parámetros que podemos enviar tenemos por ejemplo para establecer
# (por si queréis escribir en append o en binario) y una extensión por defecto:

def test():
    fichero = FileDialog.asksaveasfile(
        title="Guardar un fichero", mode='w', defaultextension=".txt")

    if fichero is not None:
        fichero.write("Hola!")
        fichero.close()
```

```
In [ ]:
```

```
In [12]: #widget.bind (evento, manejador)

from tkinter import *
from tkinter import ttk
root = Tk()
l =ttk.Label(root, text="Starting...")
l.config(font=("Verdana",24))
l.grid()
l.bind('<Enter>', lambda e: l.configure(text='Moved mouse inside'))
l.bind('<Leave>', lambda e: l.configure(text='Moved mouse outside'))
l.bind('<1>', lambda e: l.configure(text='Clicked left mouse button'))
l.bind('<Double-1>', lambda e: l.configure(text='Double clicked'))
l.bind('<B3-Motion>',
lambda e: l.configure(text='right button drag to %d,%d' % (e.x, e.y)))
root.mainloop()
```