

# Tutorial de Tkinter con Python 3

## Paso 1: Crear una ventana básica

Este es el primer paso para crear una interfaz gráfica. Se utiliza `tk.Tk()` para inicializar la ventana principal. `title()` establece el título y `geometry()` el tamaño.

```
import tkinter as tk

ventana = tk.Tk()
ventana.title("Mi primera ventana")
ventana.geometry("300x200")
ventana.mainloop()
```

## Paso 2: Añadir una etiqueta (Label)

Se usa `Label` para mostrar texto. El método `pack()` la coloca automáticamente en la ventana.

```
etiqueta = tk.Label(ventana, text="¡Hola, mundo!")
etiqueta.pack()
```

## Paso 3: Añadir un botón (Button)

El botón permite ejecutar una acción mediante una función cuando se pulsa.

```
def saludar():
    etiqueta.config(text="¡Hola desde el botón!")

boton = tk.Button(ventana, text="Saludar", command=saludar)
boton.pack()
```

## Ejercicio 1

Crea una ventana con una etiqueta y dos botones. Uno cambia el texto a 'Hola' y otro a 'Adiós'.

## Paso 4: Entrada de texto (Entry)

El widget `Entry` permite al usuario introducir texto. La función `.get()` se usa para obtener el contenido introducido.

```
def mostrar_nombre():
    nombre = entrada.get()
    etiqueta.config(text=f"Hola, {nombre}!")

entrada = tk.Entry(ventana)
entrada.pack()
boton = tk.Button(ventana, text="Aceptar", command=mostrar_nombre)
boton.pack()
```

## Paso 5: Caja de verificación (Checkbutton)

La casilla de verificación se representa con `Checkbutton`, y su estado se guarda con `BooleanVar`.

```
var = tk.BooleanVar()

check = tk.Checkbutton(ventana, text="Opción", variable=var, command=mostrar_estado)
check.pack()
```

## Tutorial de Tkinter con Python 3

```
def mostrar_estado():
    if var.get():
        etiqueta.config(text="Activado")
    else:
        etiqueta.config(text="Desactivado")
```

### Paso 6: Botones de opción (Radiobutton)

Los `Radiobutton` permiten seleccionar solo una opción entre varias. Se usa `StringVar` para guardar el valor seleccionado.

```
opcion = tk.StringVar(value="Opción 1")

radio1 = tk.Radiobutton(ventana, text="Opción 1", variable=opcion, value="Opción 1",
command=actualizar_etiqueta)
radio2 = tk.Radiobutton(ventana, text="Opción 2", variable=opcion, value="Opción 2",
command=actualizar_etiqueta)

radio1.pack()
radio2.pack()

def actualizar_etiqueta():
    etiqueta.config(text=f"Elegiste: {opcion.get()}")
```

### Paso 7: Lista desplegable (OptionMenu)

El widget `OptionMenu` permite al usuario elegir una opción de un menú desplegable.

```
seleccion = tk.StringVar(value="Python")
menu = tk.OptionMenu(ventana, seleccion, "Python", "Java", "C++", "JavaScript")
menu.pack()

boton = tk.Button(ventana, text="Mostrar", command=seleccionar)
boton.pack()

def seleccionar():
    etiqueta.config(text=f"Seleccionado: {seleccion.get()}")
```

### Paso 8: Cuadro de texto multilinea (Text)

El widget `Text` permite escribir varias líneas. `get("1.0", tk.END)` obtiene el contenido desde la primera línea hasta el final.

```
texto = tk.Text(ventana, height=5, width=40)
texto.pack()

boton = tk.Button(ventana, text="Leer texto", command=mostrar_contenido)
boton.pack()

def mostrar_contenido():
    contenido = texto.get("1.0", tk.END)
    etiqueta.config(text=f"Texto:\n{contenido.strip()}")
```

### Paso 9: Usar Frame para organizar contenido

El widget `Frame` permite agrupar otros widgets para organizarlos. Se puede usar con `.grid()` para alinearlos en filas y

## Tutorial de Tkinter con Python 3

columnas.

```
frame_superior = tk.Frame(ventana)
frame_superior.pack()

tk.Label(frame_superior, text="Nombre:").grid(row=0, column=0)
tk.Entry(frame_superior).grid(row=0, column=1)

tk.Label(frame_superior, text="Edad:").grid(row=1, column=0)
tk.Entry(frame_superior).grid(row=1, column=1)
```

### Ejercicio 2

Crea una aplicación que contenga:

- Una entrada de texto para el nombre.
- Un radiobutton para elegir el género.
- Una checkbox para aceptar términos.
- Un botón que al pulsarlo muestre un saludo personalizado.