

# python



## 05

**Ejemplo práctico**  
(Combobox y messagebox)

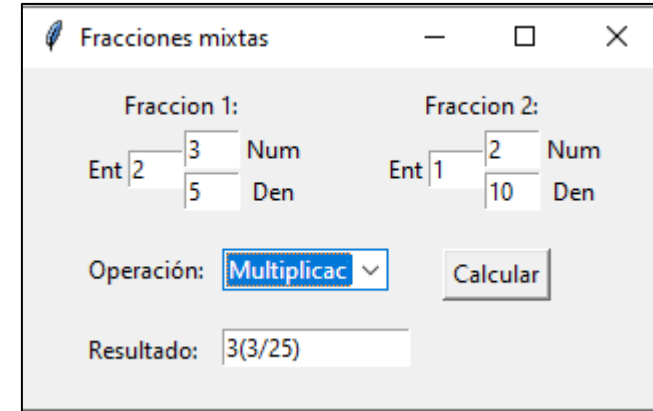
# Aplicaciones de Escritorio con

# Tkinter



# Contenido

- Operaciones
- Lista desplegable (Combobox)
- Cuadros de diálogo (MessageBox)



Fracciones mixtas

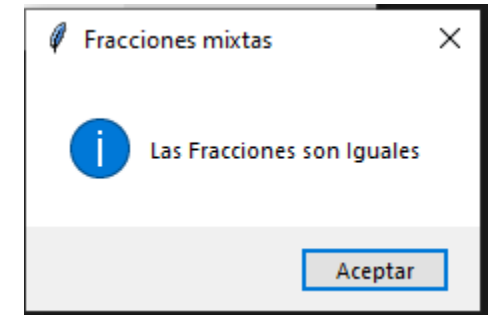
Fraccion 1: Ent 2 Num 3 Den 5

Fraccion 2: Ent 1 Num 2 Den 10

Operación: Multiplicac

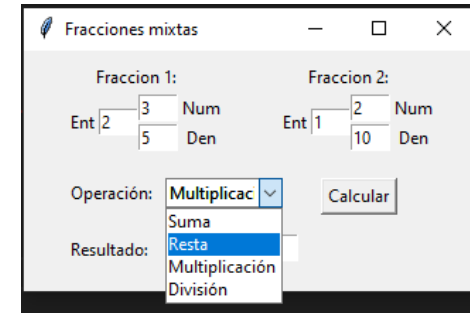
Calcular

Resultado: 3(3/25)



# Lista desplegable (Combobox)

- Combobox es una combinación entre una lista y una caja de texto. Puede actuar como una lista desplegable con determinadas opciones y, opcionalmente, permitir al usuario escribir un valor que no se encuentre en la lista.
- `tkinter.ttk.Combobox`, la cual hereda de `ttk.Entry`, por lo que todas las operaciones aplicables a una caja de texto también aplican para un combobox.
- `#importar la clase`
- `from tkinter.ttk import Combobox`



- #opciones que desplegará el combobox
- **opciones=["Suma", "Resta", "Multiplicación","División"]**
- Crear el combobox
- **cmbOpciones=Combobox(self, width="10", values=opciones, state="readonly")**
- **state= "disabled"    # "normal"**
- #Seleccionar un elemento
- **cmbOpciones.current(0)**
- #Obtener el elemento seleccionado
- **cmbOpciones.get()**
- #Obtener el índice del elemento seleccionado
- **cmbOpciones.current()**



# Lista desplegable (Combobox)

```
import tkinter as tk
from tkinter import ttk

def callbackFunc(event):
    print("New Element Selected")

app = tk.Tk()
app.geometry('200x100')

labelTop = tk.Label(app,
                    text = "Choose your favourite month")
labelTop.grid(column=0, row=0)

comboExample = ttk.Combobox(app,
                            values=[
                                "January",
                                "February",
                                "March",
                                "April"])

comboExample.grid(column=0, row=1)
comboExample.current(1)

comboExample.bind("<<ComboboxSelected>>", callbackFunc)

app.mainloop()
```

(eventos)

```
import tkinter as tk
from tkinter import ttk

def callbackFunc(event):
    print("New Element Selected")

app = tk.Tk()
app.geometry('200x100')

def changeMonth():
    comboExample["values"] = ["July",
                              "August",
                              "September",
                              "October"]

labelTop = tk.Label(app,
                    text = "Choose your favourite month")
labelTop.grid(column=0, row=0)

comboExample = ttk.Combobox(app,
                            values=[
                                "January",
                                "February",
                                "March",
                                "April"],
                            postcommand=changeMonth)

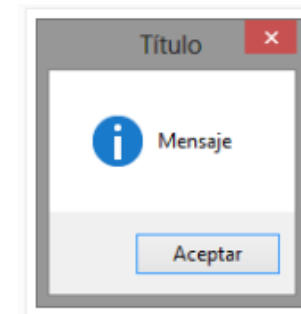
comboExample.grid(column=0, row=1)

app.mainloop()
```

# Cuadros de diálogo (messagebox)

- Es una ventana con un **título**, un **mensaje**, un **ícono** y **uno o más botones**. Se emplea para informar al usuario sobre alguna cuestión o bien para permitirle tomar una decisión.
- Es una ventana modal (Los controles de la ventana no responden mientras se muestra un messagebox)
- `showinfo()` ícono azul, `showwarning()` amarillo o naranja, `showerror()`, rojo indicando error.

```
from tkinter import messagebox  
  
messagebox.showinfo(message="Mensaje", title="Título")
```



- `showinfo()`
- `showwarning()`
- `showerror()`



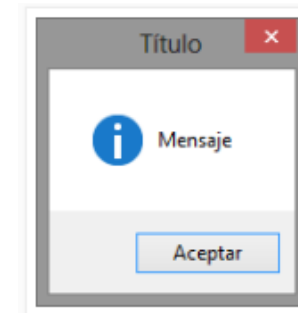
# Cuadros de diálogo (messagebox)

- `showinfo()`
- `showwarning()`
- `showerror()`
- `askquestion()`
- `askyesno()`
- `askokcancel()`
- `askyesnocancel()`
- `askretrycancel()`

- Es una ventana con un **título**, un **mensaje**, un **ícono** y **uno o más botones**. Se emplea para informar al usuario sobre alguna cuestión o bien para permitirle tomar una decisión.
- Es una ventana modal (Los controles de la ventana no responden mientras se muestra un messagebox)
- `showinfo()` ícono azul, `showwarning()` amarillo o naranja, `showerror()`, rojo indicando error.

```
from tkinter import messagebox

messagebox.showinfo(message="Mensaje", title="Título")
```

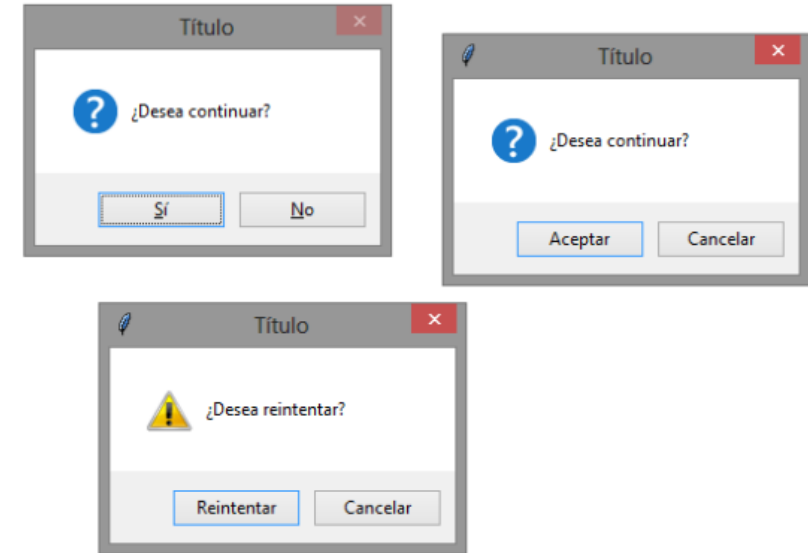


- `showinfo()`
- `showwarning()`
- `showerror()`



# Cuadros de diálogo (messagebox)

- Las **funciones de interrogación** retornan **True** o **False**, según se haya presionado «Aceptar» o «Cancelar», o bien «Sí» o «No».
- Cerrar la ventana es similar a haber cancelado, regresa False.



- `askyesno()`
- `askokcancel()`
- `askretrycancel()`

```
# Retornan True o False.  
print(messagebox.askyesno(message="¿Desea continuar?", title="Título"))  
print(messagebox.askokcancel(message="¿Desea continuar?", title="Título"))  
print(messagebox.askretrycancel(message="¿Desea reintentar?", title="Título"))
```





# Cuadros de diálogo (messagebox)

- **askquestion()** despliega los botones «Sí» y «No», su resultado es "yes" o "no".

```
# Retorna "yes" o "no".  
print(messagebox.askquestion(message="¿Desea continuar?", title="Título"))
```

- **askyesnocancel()** despliega los botones «Sí», «No» y «Cancelar», y los valores de retorno posibles son True, False y None, respectivamente. cerrar la ventana retorna None

```
# Retorna True, False o None.  
print(messagebox.asksynccancel(message="¿Desea continuar?", title="Título"))
```



# python



## 05

**Ejemplo práctico**  
(Combobox y messagebox)

# Aplicaciones de Escritorio con

# Tkinter

