

ALGORITMIA E ESTRUTURAS DE DADOS

A BIBLIOTECA TKINTER / CUSTOMTKINTER

MESSAGEBOX

CTkMESSAGEBOX

LICENCIATURA EM
TECNOLOGIAS E SISTEMAS DE INFORMAÇÃO PARA A WEB
#ESMAD #P.PORTO

❖ Biblioteca Tkinter

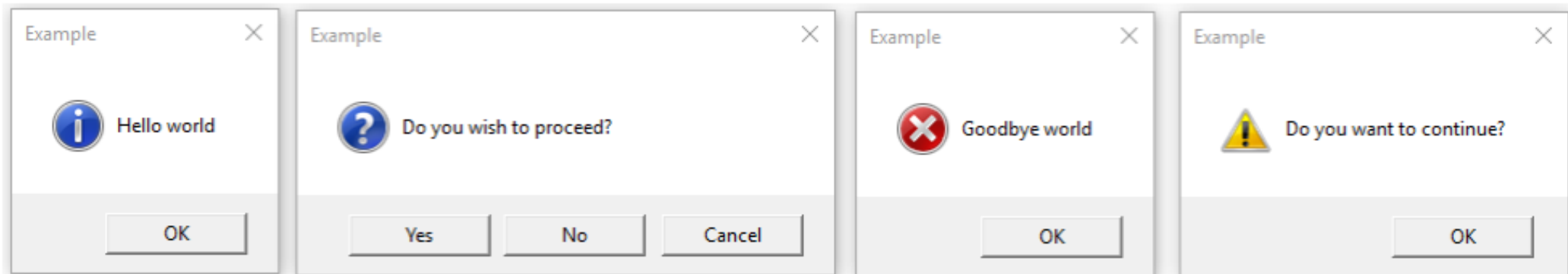
❑ **messagebox** – o módulo *messagebox* permite a implementação de mensagens em caixas de diálogo.

Este módulo providencia diversas funções para personalizar as caixas, tais como:

- ❑ `showinfo`
- ❑ `showwarning`
- ❑ `showerror`
- ❑ `askquestion`
- ❑ `askokcancel`
- ❑ `askyesno`
- ❑ `askretryignore`

```
exemplo3.py > ...  
1  # Biblioteca Tkinter: UI  
2  from tkinter import *  
3  from tkinter import messagebox  
4
```

```
>py -m pip install CTkMessageBox|
```



❖ Biblioteca Tkinter

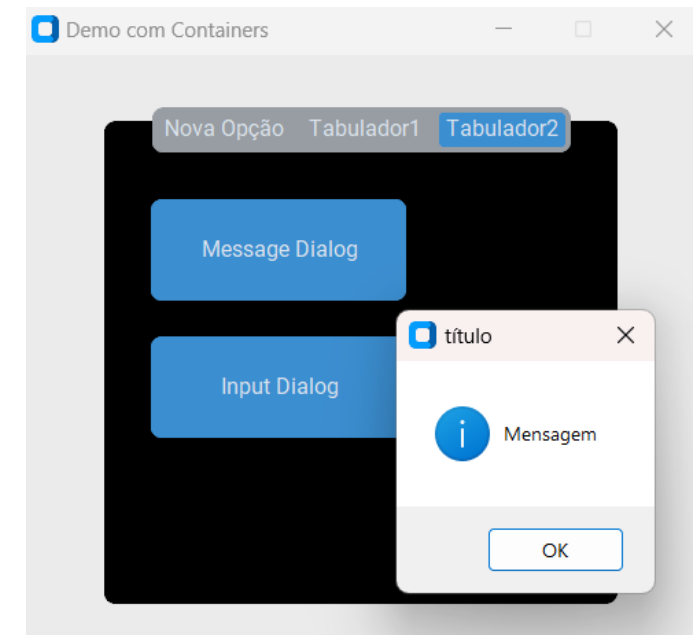
❑ **messagebox** – o módulo *messagebox* permite a implementação de mensagens em caixas de diálogo.

Este módulo providencia diversas funções para personalizar as caixas, tais como:

- ❑ showinfo
- ❑ showwarning
- ❑ showerror
- ❑ askquestion,
- ❑ askokcancel,
- ❑ askyesno,
- ❑ askretryignore

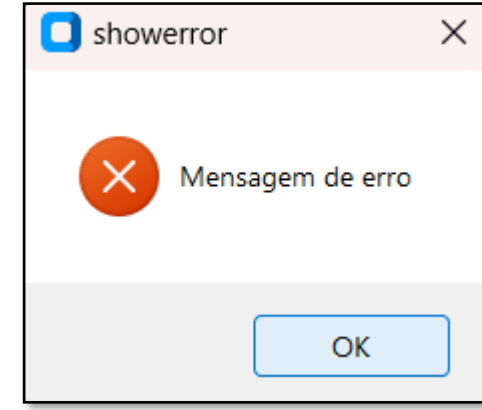
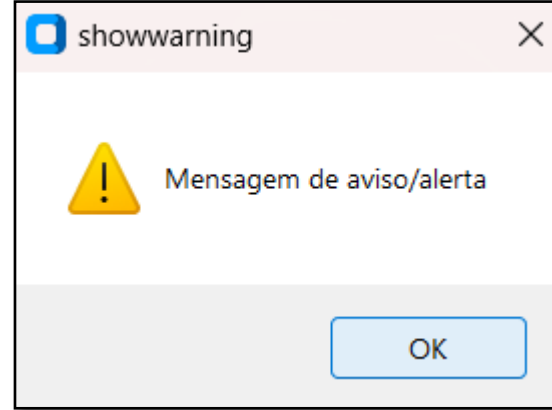
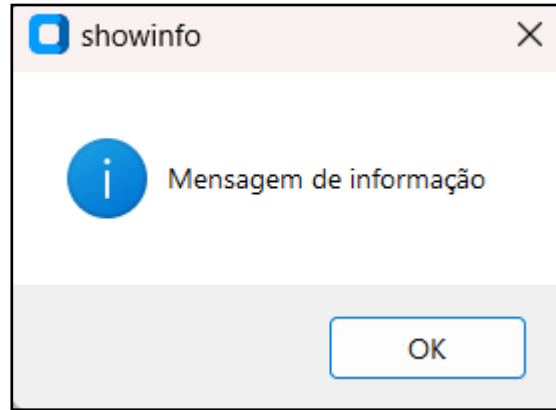
```
1 def messageDialog():  
2     messagebox.showinfo("título", "Mensagem")
```

```
exemplo3.py > ...  
1 # Biblioteca Tkinter: UI  
2 from tkinter import *  
3 from tkinter import messagebox  
4
```



❖ Biblioteca Tkinter

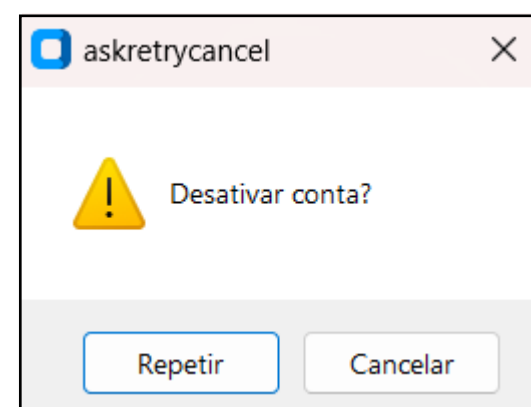
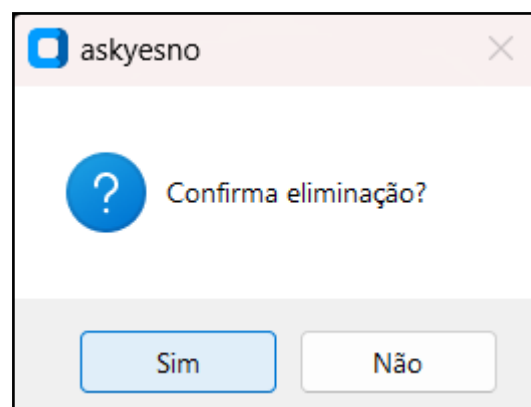
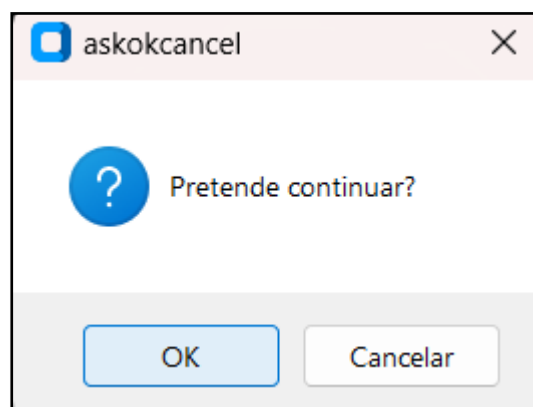
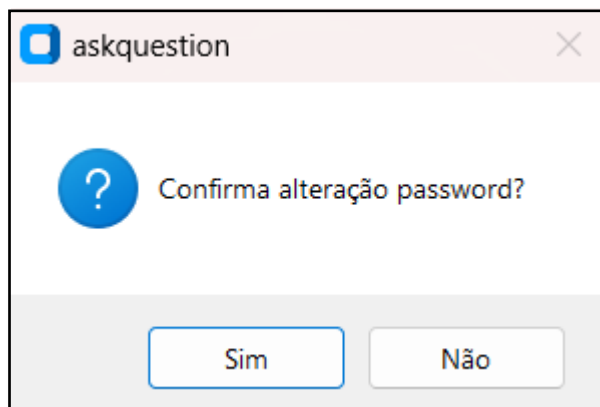
❑ messagebox



```
1 def messageDialog():  
2     messagebox.showinfo("showinfo", "Mensagem de informação")  
3     messagebox.showwarning("showwarning", "Mensagem de aviso/alerta")  
4     messagebox.showerror("showerror", "Mensagem de erro")  
5
```

❖ Biblioteca Tkinter

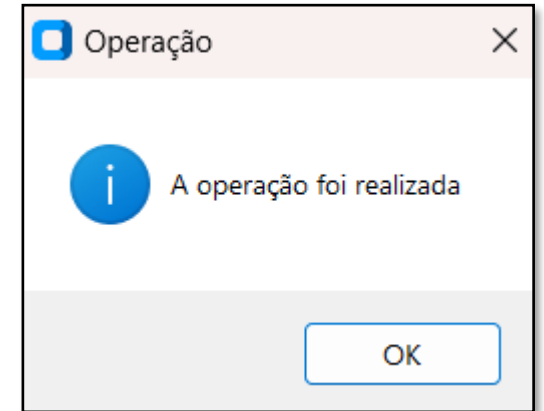
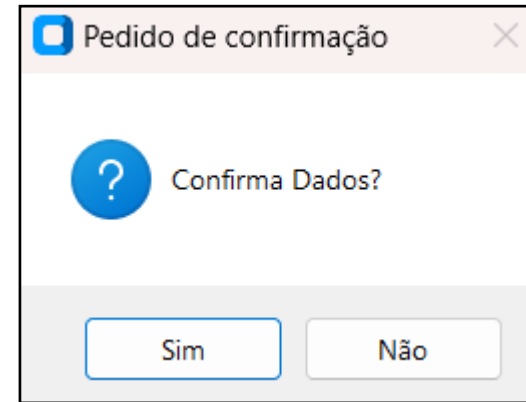
❑ messagebox



```
1 def messageDialog():
2     messagebox.askquestion("askquestion", "Confirma alteração password?")
3     messagebox.askokcancel("askokcancel", "Pretende continuar?")
4     messagebox.askyesno("askyesno", "Confirma eliminação?")
5     messagebox.askretrycancel("askretrycancel", "Desativar conta?")
6
```

❖ Biblioteca Tkinter

❑ messagebox



```
1  resposta = messagebox.askquestion("Pedido de confirmação", "Confirma Dados?")
2  if resposta== "yes":
3      messagebox.showinfo("Operação", "A operação foi realizada")
4  else:
5      messagebox.showinfo("Operação", "A operação foi cancelada")
6  #
7  resposta = messagebox.askokcancel("Pedido de confirmação", "Confirma Eliminar Registo?")
8  if resposta== "True":
9      messagebox.showinfo("Operação", "A operação foi realizada")
10 else:
11     messagebox.showinfo("Operação", "A operação foi cancelada")
```


❖ Biblioteca Tkinter

❑ messagebox

```
6
7 def remove_tree():
8     res= messagebox.askokcancel(title= "", message="Confirma eliminação de dados?")
9     if res == True:
10         tree.delete(*tree.get_children()) # REMOVE todas as linhas da tree
11
12
```

Consultas

Tipo de Movimento

☒ Entrada

☐ Saída

Por Utilizador

Número:

Consultar

Número	Data	Hora	Movimento
220	2021-12-29	16:00:38.329945	Entrada
120	2021-12-29	16:00:43.884554	Entrada
178	2022-01-05	14:53:39.240774	Entrada
220	2021-12-30	16:00:38.329945	Entrada
120	2021-12-30	16:00:43.884554	Entrada

Obter Dados

Remover Tudo

Número:

Data:

Hora:

Movimento:

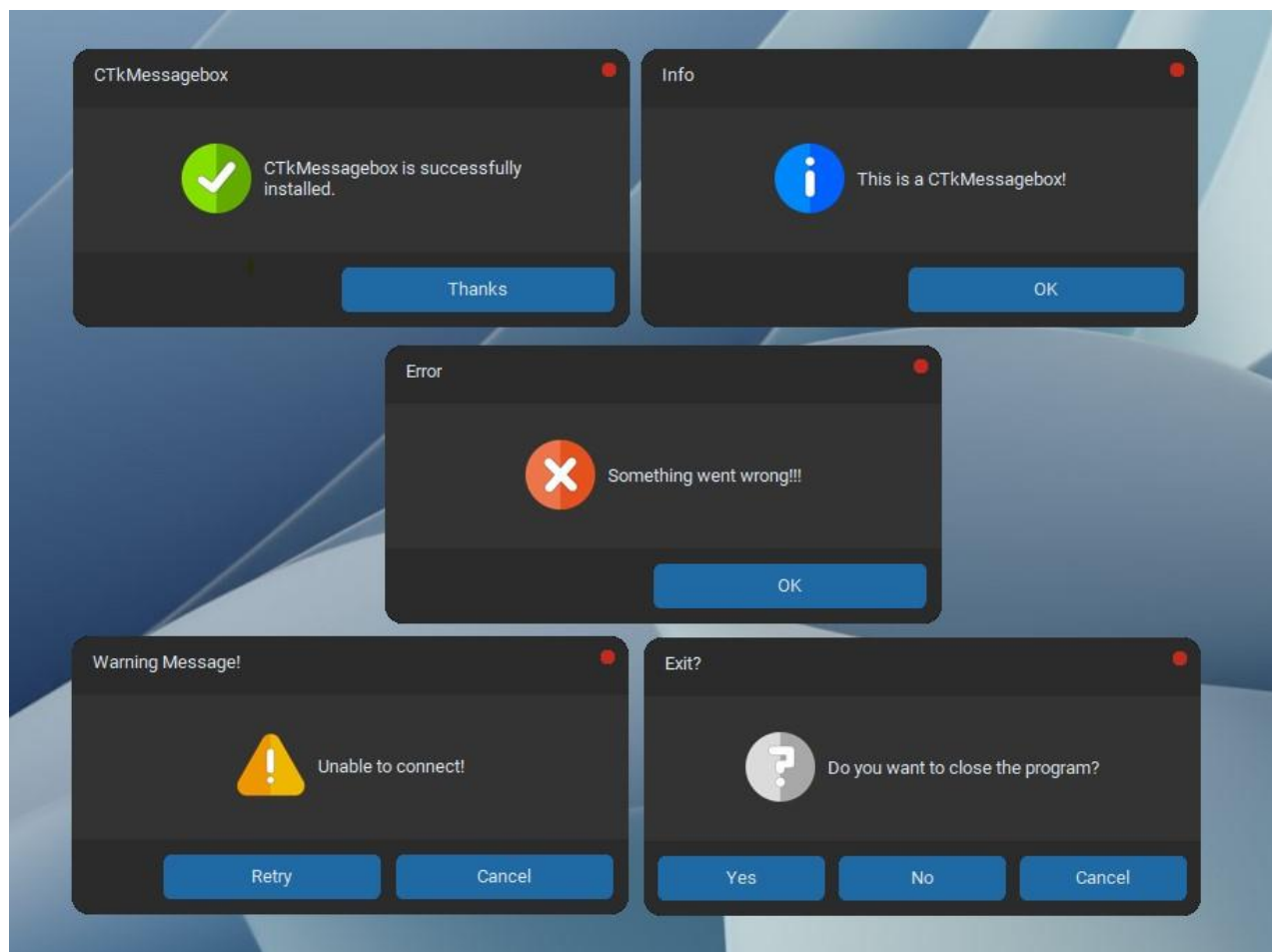
Confirma eliminação de dados?

OK Cancelar

❖ Biblioteca Tkinter

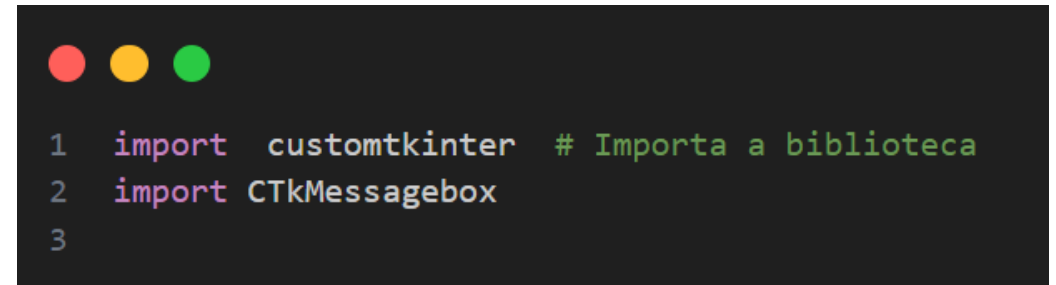
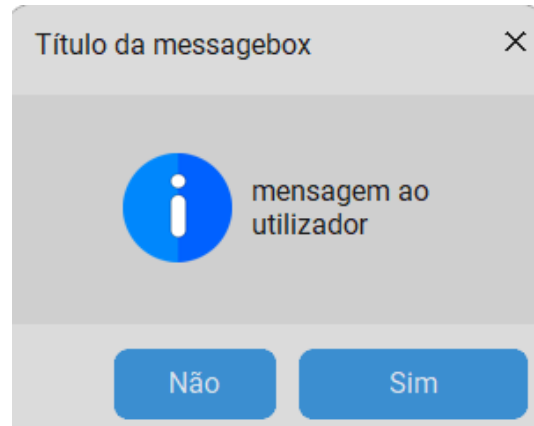
❑ CtkMessageBox

```
o>py -m pip install CtkMessageBox
```

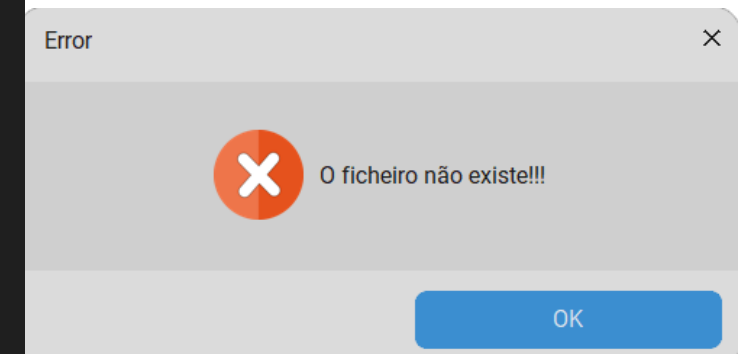
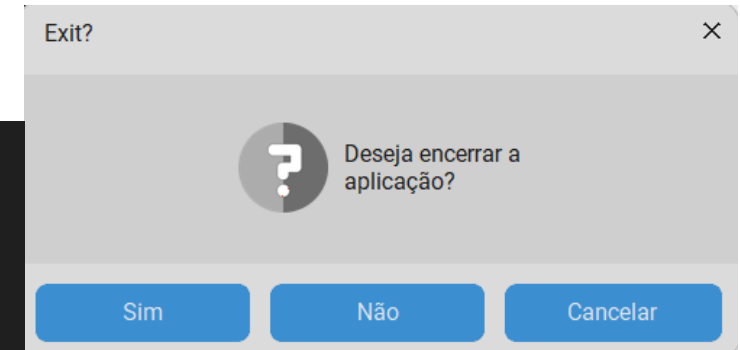


❖ Biblioteca Tkinter

CTkMessageBox



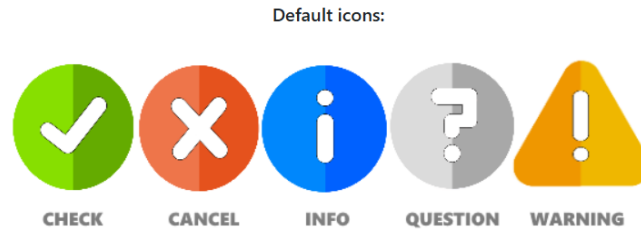
```
1 # messagebox Informação
2 CTkMessageBox.CTkMessageBox(app, width=200, height=200, title="Título da messagebox",
3                             message= "mensagem ao utilizador",
4                             option_1="Sim", option_2="Não")
5 # messagebox error
6 CTkMessageBox.CTkMessageBox(title="Error", message="O ficheiro não existe!!!", icon="cancel")
7
8 # messagebox ask question
9 msg = CTkMessageBox.CTkMessageBox(title="Exit?", message="Deseja encerrar a aplicação?",
10                                  icon="question", option_1="Cancelar", option_2="Não", option_3="Sim")
11 response = msg.get()
12
13 if response=="Sim":
14     app.destroy()
15 else:
16     print("Click 'Yes' to exit!")
```



❖ Biblioteca Tkinter

❑ CTkMessageBox

- ❑ title, message
- ❑ width, height
- ❑ fg_color, bg_color
- ❑ option_1 (default "OK"), option_2, option_3
- ❑ icon
- ❑ button_color, text_color



```
1 import customtkinter # Importa a biblioteca
2 import CTkMessageBox
3
```

