

¿ Como puedo detener o bloquear un ventana cuando abro una toplevel?

Formulada hace 1 año y 11 meses Modificada hace 1 año y 9 meses Vista 816 veces



1



Tengo una ventana y cuando presiono un botón se abre una Toplevel, pero quiero que la ventana principal se detenga por así decirlo, no quiero destruirla ni hacerla desaparecer quiero que mientras este abierta la Toplevel no se pueda acceder a la ventana principal hasta que la Toplevel se cierre.

no encontré nada parecido y relacionado con esto agradecería cualquier ayuda.



```
from tkinter import *

root = Tk()

def top():
    Toplevel()

Button(root, text="toplevel", command=top).pack()

root.mainloop()
```

python

tkinter

[Compartir](#) [Mejora esta pregunta](#) [Seguir](#)

formulada el 29 dic. 2020 a las 1:11

True	False
1	0
HIGH	LOW

Boolean 796 4 19

1 respuesta

Ordenado por:

Mayor puntuación (por defecto)



3



Por lo que entiendo usted está tratando de deshabilitar la interacción del usuario con la ventana principal o el padre mientras el hijo o el Toplevel esta abierto, en otras palabras se podría llamar una ventana modal.

La solución más fácil es mantener el foco del Toplevel y todos los eventos de interacción de la ventana inicial o principal a el Toplevel. Se debe tener el foco hasta que usted cierre el Toplevel y así reactivar de nuevo la ventana inicial. Su código le falta agregar el método `grab_set()` , y `focus_set()` en la función del Toplevel.

Lo correcto sería:

```
from tkinter import *

root = Tk()

def top():
    newTop = Toplevel()
    newTop.grab_set() # Mantiene el foco de la ventana hasta que se cierre y
    devuelve la interacción con la ventana principal el root en este caso.
    newTop.focus_set() # Mantiene el foco cuando se abre la ventana.
    newTop.mainloop()
```

```
Button(root, text="toplevel", command=top).pack()
```

```
root.mainloop()
```

Si usted quiere mantener un foco global que no permita el movimiento de las ventanas y su interacción la solución es la siguiente:

```
from tkinter import *

root = Tk()

def top():
    newTop = Toplevel()

    btn = Button(newTop, text="toplevel", command=newTop.destroy).pack() # Se
    agrega un botón para cerrar la ventana

    newTop.grab_set_global() # Obliga a las ventanas estar deshabilitadas y
    deshabilitar todos los eventos e interacciones con la ventana
                                # Cuidado los eventos se pueden deshabilitar.
    newTop.focus_set() # Mantiene el foco cuando se abre la ventana.
    newTop.mainloop()

Button(root, text="toplevel", command=top).pack()

root.mainloop()
```

Si usted quiere eliminar los botones de min/maximizar y solo dejar el botón de cerrar solo debe agregar `transient()`, debe especificar la ventana inicial en este caso es `root`. Un breve ejemplo es poner `newTop.transient(root)` debajo de `newTop.focus_set()`.

Compartir Mejora esta respuesta Seguir

respondida el 24 feb. 2021 a las 21:53



John Deb

686 1 12

es exactamente lo que buscaba. – [Boolean](#) el 24 feb. 2021 a las 23:09

-
- 1 Me alegra que le haya funcionado y servido, acabo también de encontrar una manera más sencilla si usted quiere mostrar como un cuadro de diálogo o ventana **modal**, solo debe establecer `root.attributes('-disabled', 1)` para impedir el acceso y ejecución de eventos de la ventana principal y `root.attributes('-disabled', 0)` si es para devolver el estado anterior de la ventana para poder tener el ciclo de eventos en la ventana principal al momento de destruir el `Toplevel`. – [John Deb](#) el 17 mar. 2021 a las 22:19 ✎
-