



Programación

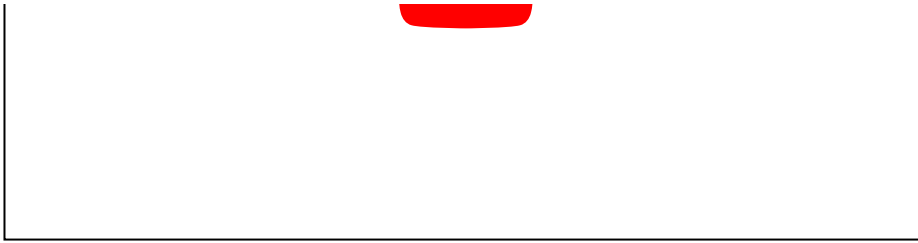


Fácil

¿Qué son los widgets? - El widget frame() y el método pack() - Tkinter desde cero - Capítulo 2

¿Qué son los widgets? - El widget Frame() y el ...





En este capítulo hablaremos un poco sobre **los widgets de Tkinter**. Para no saturarnos, veremos el **widget `Frame()`** y el **método `pack()`** que vamos a utilizar para empaquetar los widgets (ahora explicaré esto).

Hay muchas cosas que ver con **los widgets**, por lo que iremos viendo mucho más a lo largo del curso.

En casi cualquier interfaz gráfica, contamos con elementos comunes como son los botones, etiquetas, marcos, etc. Estos elementos están en cualquier programa, ya sea de Android, de Windows, Linux o Mac.

¿Qué es un `Frame()` de `tkinter`?

Lo primero que debes saber es **que es un `Frame()`**. Lo podemos traducir al español como marco. Este **`Frame()`**, va a ser un **contenedor para otros widgets en la ventana raíz**, la ventana root que creamos en el capítulo anterior.

Los widgets en Tkinter deben empaquetarse, es decir, decirle de alguna forma al programa, que este widget pertenece a la ventana raíz. Para esto, debemos utilizar **el método `pack()`**, el cual no te expliqué en el capítulo anterior.

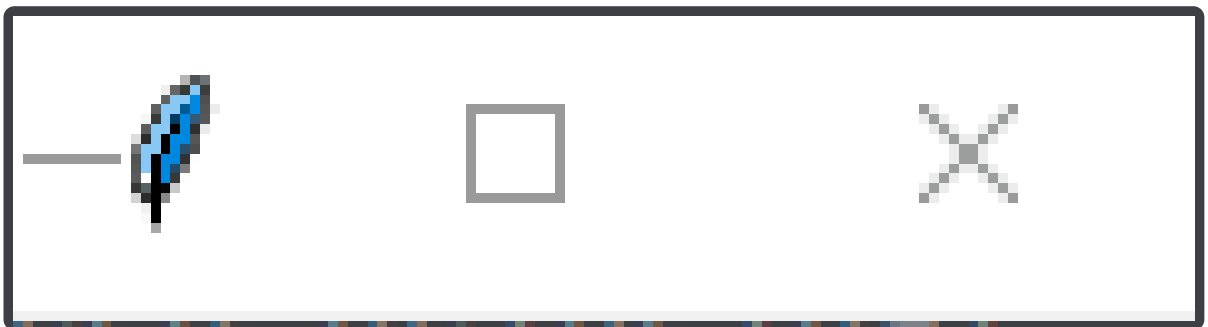
Si no utilizas este método, por muchos **widgets** que crees, no se van a mostrar en la ventana, así que deberás acostumbrarte a usarlo.

¿Cómo utilizar el widget Frame() de tkinter?

Código Python

```
1 from tkinter import *  
2 root = Tk()  
3  
4 marco_principal = Frame()  
5 marco_principal.pack()  
6  
7 root.mainloop()
```

Si ejecutamos esto así, nos queda una ventana sin nada, con un tamaño mínimo. Esto es porque no hemos especificado nada todavía.



¿Cómo redimensionar un Frame() de Tkinter?

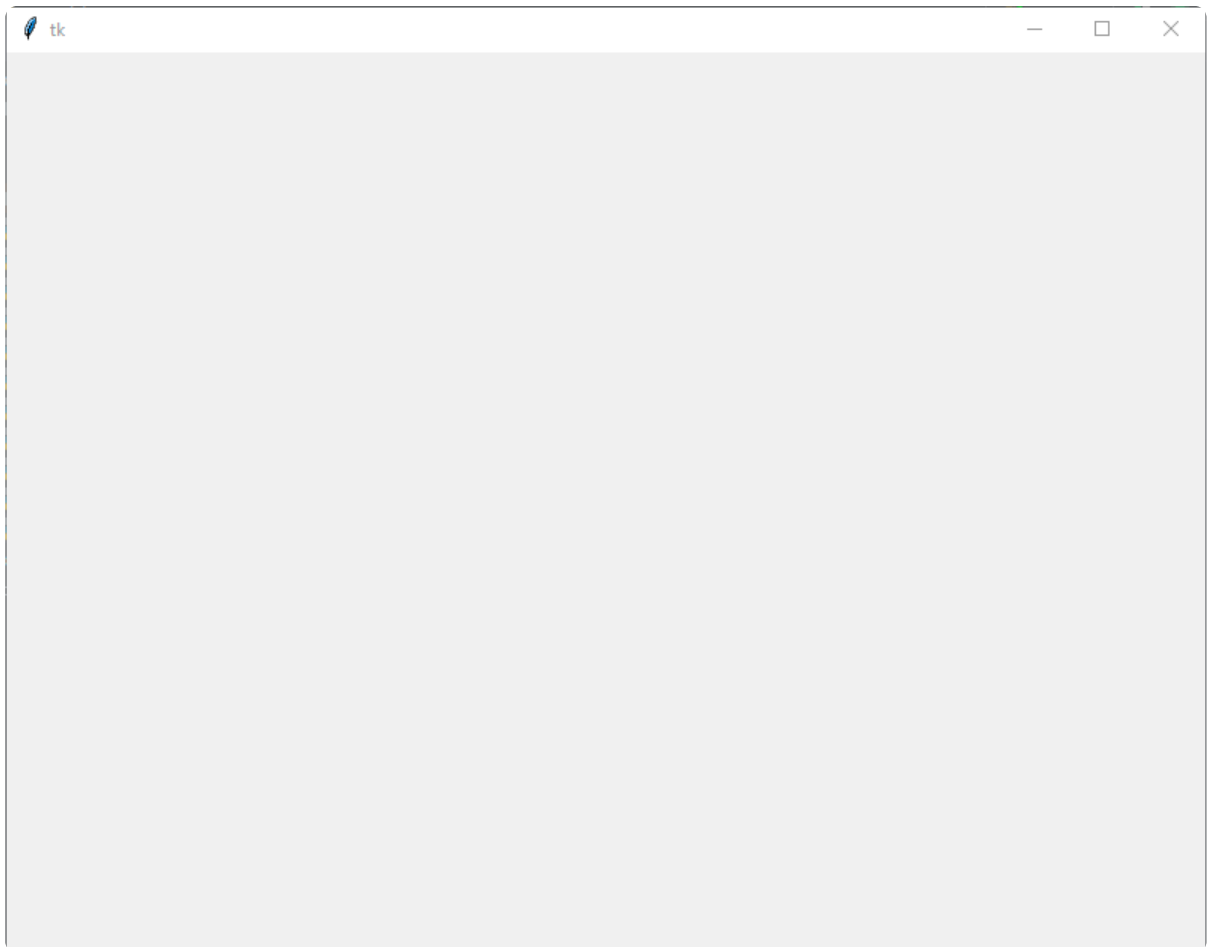
Los widgets, se pueden modificar muchísimo. Vamos a ver un ejemplo. Para redimensionar el Frame(), hay que utilizar `config()` y entre sus paréntesis especificar un `width=""` y un `height=""` para el ancho y el alto del marco.

Código Python

```
1 from tkinter import *LF
2 root = Tk()LF
3 LF
4 #Creamos el marcoLF
5 marco_principal = Frame()LF
6 LF
7 #Empaquetamos el marcoLF
8 marco_principal.pack()LF
9 LF
10 #Redimensionamos el marco el marcoLF
11 marco_principal.config(width="800", height="600")LF
12 LF
13 root.mainloop()
```

🖥️ Resultado 🖥️

La ventana ha crecido. Esto es porque el marco le esta haciendo adoptar sus dimensiones. La ventana raíz se está adaptando dinámicamente al tamaño total de los elementos que hay en ella.



¿Cómo se le puede dar un color de fondo a un Frame() de Tkinter?

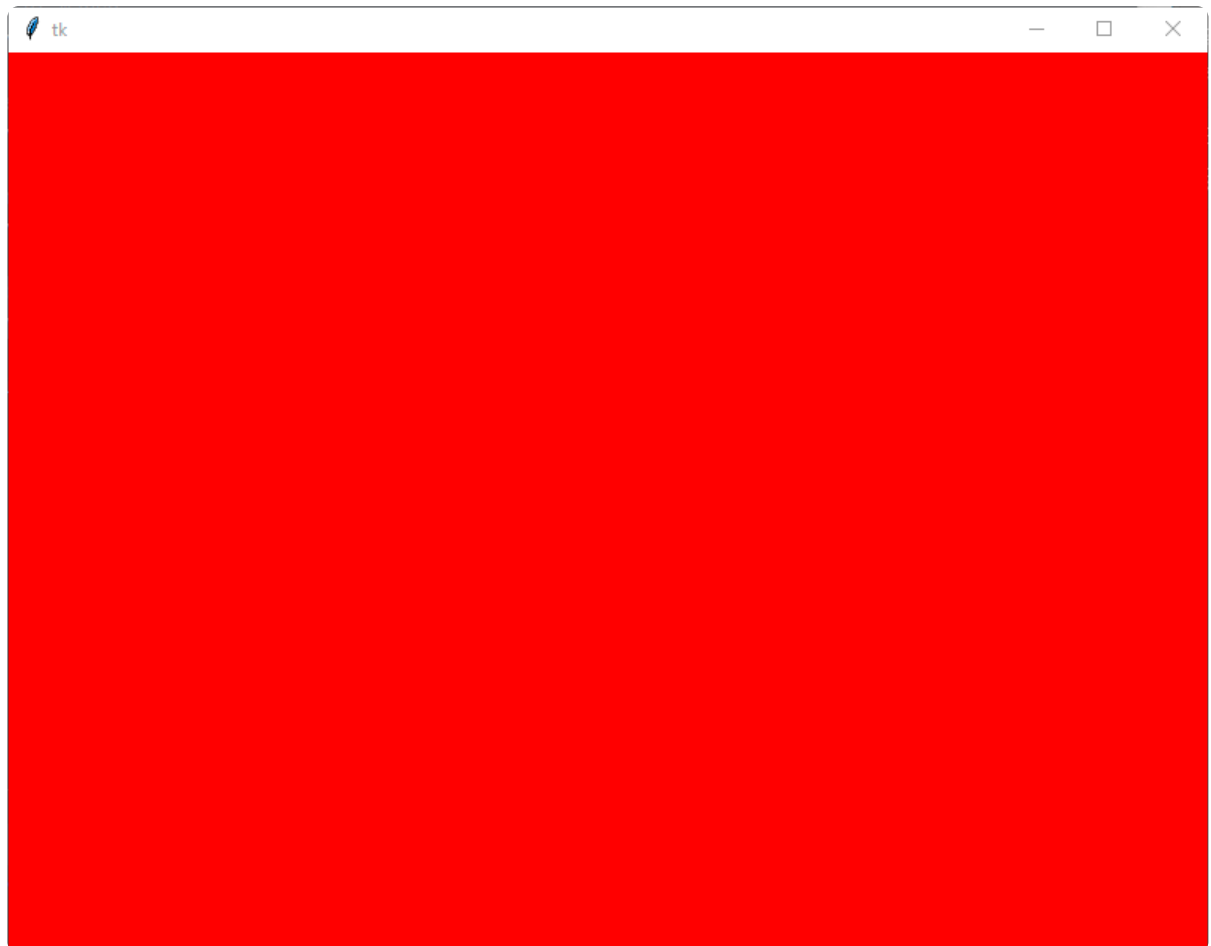
Ahora le vamos a dar un color para poder distinguir este Frame() de la propia ventana. Esto lo hacemos con la propiedad `bg=""` en el `config()`.

Código Python

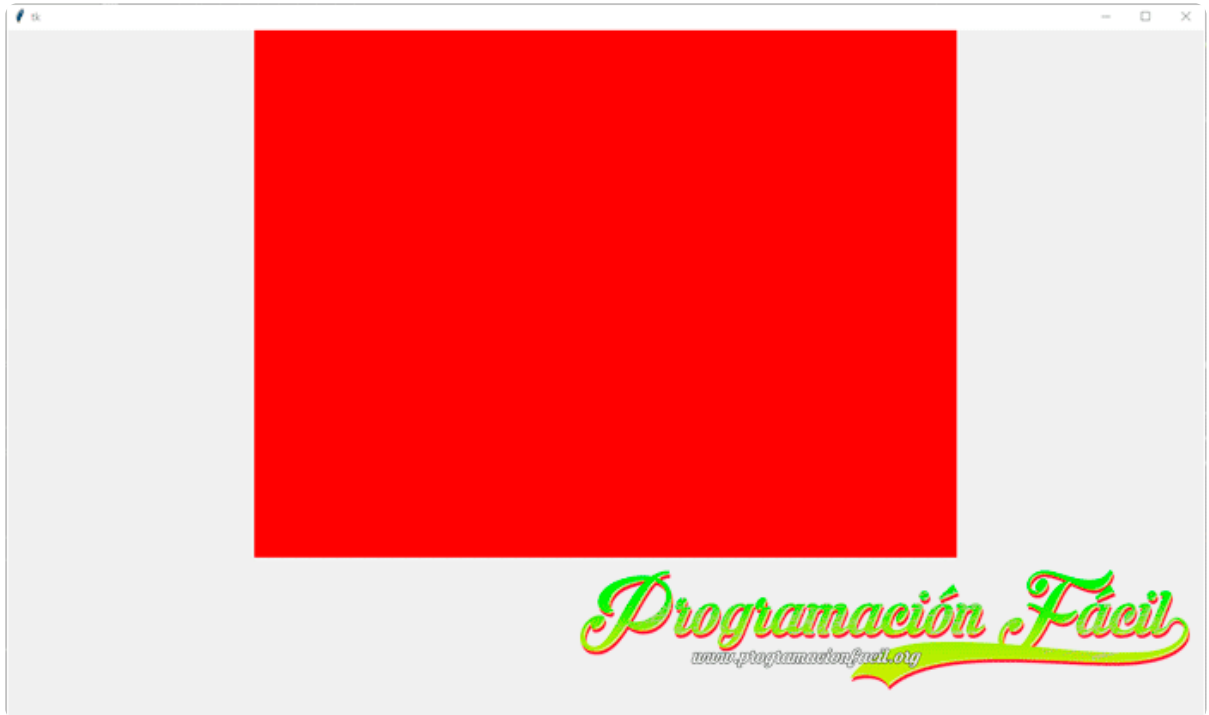
```
1 | marco_principal.config(bg="red")
```

Resultado

Por el momento, ya vemos el color que le he indicado.



Ahora podemos agrandar la ventana raíz y ver que ocupa el marco y que la ventana se está ajustando a él.



Voy a dejar el capítulo aquí, no quiero darte demasiados conceptos de golpe.
Seguiremos viendo mucho más en el siguiente capítulo.

Suscríbete a mi canal de YouTube para apoyarme

Si te ha gustado este curso y crees que el trabajo merece la pena, te agradeceré eternamente que te suscribas a mi canal de YouTube para apoyarme y que pueda seguir haciendo cursos gratuitos.

Además, si te encanta la programación, tienes un montón más de cursos gratuitos para ver.

No solo eso, podrás participar enviándome comentarios con tus sugerencias para temas específicos o cursos completos o incluso las dudas que tengas y las intentaré ir resolviendo en los cursos que estén todavía abiertos.



Programación Fácil

YouTube

12 K

Comentarios

Si te quedan dudas sobre el temario, sobre Python, o cualquier otra cosa relacionada o simplemente quieres agradecer, aquí tienes tu sitio para dejar tu granito de arena. Gracias por tus comentarios y por darle vida a este sitio web.

Nickname

Email (optional)

Reply...

Comment

[Comments powered by Cusdis](#)