Para usar tkinter solamente necesitamos tres líneas de código:

from tkinter import \*

root = Tk()

root.mainloop()

Primero importamos el módulo, segundo creamos una instancia de la clase Tk(), como vemos es letra inicial mayúscula, por lo que indica presencia de una clase (esto sería como la clase ventana, por así decirlo) y luego le pasamos el metodo mainloop de la clase Tk() que lo que hace es crear un while con un break, bajo la condición de cerrar la ventana, o lo que es lo mismo, tocar la X

Tkinter y sus Clases y módulos:

Una cosa que no me había percatado es que en lo mas básico, todos y cada uno de los módulos y librerías de python son lo mismo y siguen las mismas reglas. Si lo veo como clases y métodos de clases con sus respectivos argumentos y funciones es mucho mas fácil aprender tkinter.

Por ejemplo si queremos agregar una label debemos primero crear una instancia de la clase Label, por eso se escriben las instancias con ,mayúscula, y luego se escriben sus métodos pertenecientes a cada una de las clases.

Hay que recordar que muchas veces los puntos significan métodos o funciones generales del modulo o la librería. Por ejemplo:-

label\_instance = Label(root)

Aquí creamos una instancia de la clase Label, la cual tiene como parámetro root, este parámetro significa en que ventana se encuentra ubicado, y es un parámetro que comparten todas las clases, por lo que imagino que hay una clase madre, llamémosla “widget”, que posee los parámetros que comparten las demás, aplicandose así la OOP de python.

Los Objetos de la clase Label, o de la clase entry, también compatrten métodos que son comunes para todos los widgets, como el metodo **config**, o el metodo **grid**, los cuales vienenn con parámetros que alteran el comportamiento de los widgets.

**Label\_instance.config (\*\*kwarg1, \*\*kwarg2)**

Variables de Tkinter:

Tkinter tiene sus propias variables, para trabajar con los datos que obtenemos del usuario, así como para también modificarlos, trabajar con ellos, imprimir un resultado y asignar una variable a este resultado, debemos de “crear” la variable dos veces. Lo que se hace es que primero se declara la verdadera variable, con un nombre elegido por nosotros, y se le da un tipo de variable como **StringVar().** ¡¡Y oh sorpresa!! Al parecer no estas creando una variable sino una clase, que por supuesto, tiene sus métodos. Luego de haber creado las instancias de la clase **StringVar()**, estas instancias pueden ser utilizadas para guardar valores ahí, y recuperarlos mas tarde con el metodo **.get()**, de las “variables”.

Visualmente se ve así:

variable = StringVar ()

entry = Entry (root, justify = “center”, **textvariable** = variable).pack()

luego para obtener el dato del usuario usamos **variable.get()**

Metódos importantes:

Unos métodos que son importante mencionar cuan hablamos de tkinter, son el metodo pack, grid o el config, a los cuales se les pasan diferentes \*\*kwargs, son kwargs y tienen dos asteriscos porque se pueden definir con al menos uno y no todos son necesarios, y como son calores de clave valor que pueden tomar diferentes valores como por ejemplo: “disabled” o “normal”, en caso de una entrada de texto, Pues eso, son \*\*kwargs o **K**ey-**W**ord-**ARG**ument**S**

Las configuraciones mas importantes se hacen en el metodo **config**

Botones, el uso de botones es el más importante por su capacidad de que le agreguen comandos, los “comandos” que no son mas que funciones, deben de estar declarados antes de la llamada del botón, o sea antes de que este sea declarado, porque hay que recordar que el intérprete lee línea a línea, y si llega a esa línea y trata de leer una función que aun no esta declarada, pues alzará un error. No es necesario que la función sea declarada antes de o después de la ventana TK(), ya que simplemente la estamos declarando, lo que es importante es que, a la hora de llamar la función a través del botón, todos sus componentes estén ya creados.

RadioBotones: Estos son botones de los cuales solamente se puede elegir uno en la clase Tk() en la que se encuentren, para poder poner más, había que crear frames, y entonces ponerlos en frames. Por defecto vienen clicados, y si, también podemos usar las “variables” te tkinter para poder guardar los datos recibidos por el usuario, que en este caso, no necesariamente tiene que ser todo el texto, lo podríamos guardar como opción, 1, 2 , 3, cosa que ya no sería una StringVar() sino una IntVar(). Los RadioBotones pueden compartir la misma variable, pero, debemos de también pasarles un argumento llamado valor para poder hacer independientes a cada uno de ellos cuando los cliquemos y así identificar cual de estos fue el seleccionado.

opcion = IntVar()

Radiobutton(root, text = “Opción 1”, variable = opción, value = 1).pack() Radiobutton(root, text = “Opción 1”, variable = opción, value = 1).pack() Radiobutton(root, text = “Opción 1”, variable = opción, value = 1).pack()

Si hacemos un get a la variable opcion, lo que vamos a obtener es uno de los valores, 1, 2 o 3.

Los Checkbutton son iguales con la única diferencia que puedes elegir entre uno y todos, este caso es particularmente interesante ya que los value solamente pueden ser 0 o 1, o sea, si o no, y en este caso si haríamos una variable independiente para cada uno de los checkbuttons

leche = IntVar() azúcar = IntVar()

Label(root, text = “¿Como quieres el café?”).pack()

Checkbutton (root, text = “Con leche”, variable = leche, onvalue = 1, offvalue = 0). pack () Checkbutton (root, text = “Con azucar”, variable = azucar, onvalue = 1, offvalue = 0). pack ()

Menus: Lo primero que tenemos que hacer es crear una instancia de la clase Menu. Los menus son una clase un tanto especial a la hora de ponerlos en la ventana ya que no se empaquetan porque estos tienen prioridad y siempre están por encima de todo, por lo que, para poder posicionarlos, es necesario pasarle a la instancia de la clase Tk(), **no a la instancia menu**, el metodo config y ahí le agregamos el menu como un parámetro \*\*kwargs con el key = menu y word = al valor de la instancia menu.

root = Tk()

menubar = Menu(root)

root.config (menu = menubar)

root.mainloop()

Luego que tenemos esto, podemos intuir que los diferentes menus van a contener otros menus, por lo que podríamos tratar a la clase menu como una subclase del tipo Tk() la cual contiene a otros, ya que la manera de la cual agregaríamos los demás menus es de la siguiente:

root = Tk()

menubar = Menu(root)

root.config (menu = menubar)

**filemenu = Menu(menubar)**

root.mainloop()

De esta manera los elementos ya han sido creados, ahora solamente necesitamos crear tanto el texto como la posición que ocuparán en el menu, que como simplemente estarán uno por debajo del otro simplemente los podemos empaquetar y ya está.

root = Tk()

menubar = Menu(root)

root.config (menu = menubar)

filemenu = Menu(menubar)

**menubar.add\_cascade(Label = “Archivo”, menu=filemenu)**

root.mainloop()

Algo a tomar en cuenta es que cuando creamos estos menus por defecto nos crea un elemento **tearoff**, el cual podemos simplemente declararlo como “0” y ya está.

Luego para agregar los botones, no los menus, debajo del menu podemos declarar estos comandos con el metodo **filemenu.add\_command** (Label = “Nuevo”), por ejemplo, de esta manera, lo que se esta creando no es otro menu, sino un elemento del menu que hará algo cuando ciclemos sobre el

Otro elemento que podemos agregar es un separador, o sea una línea que separa horizontalmente elementos del menu.

Popup avanzados:

Primeramente, los popups no aparecen cuando importamos todo de tkinter con el carácter “\*” sino que tenemos que importarlos de la siguiente manera:

from tkinter import messagebox as MessageBox

Aquí recordemos que lo que viene después de as es opcional, es como cuando abrimos un archivo y le damos un nombre para referirnos a el mas adelante y aplicarle metodos, lo mismo pasa con el messagebox.

Siguiendo esta misma analogía, hay otro módulo de tkinter que tenemos que importar directamente y que no viene incluido con el carácter “\*” que es el modulo colorchooser que se importaría de la manera siguiente:

from tkinter import colorchooser as ColorChooser

ColorChooser es la típica ventana emergente en al que podemos elegir un color en RGB o en formato #ffffff (hexadecimal) con el metodo:

color = ColorChooser.askcolor(title = “Elige un color”)

color en este caso s es una instancia de la clase askcolor del modulo ColorChooser el cual es realmente una tupla que contiene dos elementos de los cuales el primero es una tupla con los valores RGB, o sea una tupla con 3 flotantes, y el segundo es el valor del color en formato hexadecimal, en caso de no seleccionar ningún valor simplemente devuelve una tupla con dos valores del tipo None.