---------- Python Tkinter Tutorial ----------

```
'''
Tk is a GUI toolkit used for creating desktop
applications that work on Windows, MacOS and Linux
Tk provides tons of widgets (Buttons, Scrollbars, etc.)
that are used to create applications.

Tcl (Tool Command Language) is a programming
language used for developing web & desktop applications
'''

# Get the standard library for Tk
from tkinter import *

# Get the newest widget themes from Tk 8.5
from tkinter import ttk


def get_sum(*args):
    try:
        # Cast string to a float
        num_1_val = float(num_1.get())
        num_2_val = float(num_2.get())

        # Set the value of solution to update
        # the entry box
        solution.set(num_1_val + num_2_val)
    except ValueError:
        pass


# Create the main window that holds all the widgets
root = Tk()

# Define the title for the window
root.title("Calculator")

# The frame surrounds the interface with the widgets
# A frame is used so the widgets and background
# colors are consistent
# Define padding for left top and right bottom
frame = ttk.Frame(root, padding="10 10 10 10")

# ---------- GRID GEOMETRY MANAGER  ----------
# The Grid manager is the most useful using a series
# of rows and columns for laying out widgets

# Each cell can only hold 1 widget, but a widget
# can cover multiple cells.

# rows start at 0, 1, ...
# columns start at 0, 1, ...
```

```
# sticky defines how the widget expands (N, NE, E, SE,
# S, SW, W, NW)

# Define that a grid should stick to the North, West,
# East and South sides of the frame
frame.grid(column=0, row=0, sticky=(N, W, E, S))

# Define that the frame should expand with the main window
# If columns and rows have the same weight they will
# expand at the same rate when the interface is expanded
root.columnconfigure(0, weight=1)
root.rowconfigure(0, weight=1)

# Define Tkinter string variables
num_1 = StringVar()
num_2 = StringVar()
solution = StringVar()

# Create entry box 7 characters long that has the value
# entered assigned to num_1
num_1_entry = ttk.Entry(frame, width=7,
                        textvariable=num_1)

# Place in the 1st column, 1st row
# W E means that the widget should expand horizontally
# with the surrounding interface
num_1_entry.grid(column=1, row=1, sticky=(W, E))

# Place a label with the value + in the 2nd column
ttk.Label(frame, text="+").grid(column=2, row=1,
                                sticky=(W, E))

# Create 2nd number entry box
num_2_entry = ttk.Entry(frame, width=7,
                        textvariable=num_2)
num_2_entry.grid(column=3, row=1, sticky=(W, E))

ttk.Button(frame, text="Add", command=get_sum).grid(column=1, row=2, sticky=W)

solution_entry = ttk.Entry(frame, width=7, textvariable=solution)
solution_entry.grid(column=3, row=2, sticky=(W, E))

# Put focus on the num_1 entry box
num_1_entry.focus()

# When the return button is pressed call the function calculate
root.bind('<Return>', get_sum)

# A loop that executes until the application exits
root.mainloop()
```