All the code for a tool tip

Start with the class.

import tkinter as tk

class ToolTip:

**def** \_\_init\_\_(self, widget, text):

self.widget = widget

self.text = text

self.tip\_window = None

self.widget.bind("<Enter>", self.show\_tip)

self.widget.bind("<Leave>", self.hide\_tip)

**def** show\_tip(self, event=None):

if self.tip\_window or not self.text:

return

x, y, \_, cy = self.widget.bbox("insert")

x += self.widget.winfo\_rootx() + 25

y += self.widget.winfo\_rooty() + cy + 20

self.tip\_window = tw = tk.Toplevel(self.widget)

tw.wm\_overrideredirect(True)

tw.wm\_geometry(f"+{x}+{y}")

label = tk.Label(tw, text=self.text, background="#ffffe0", relief="solid", borderwidth=1)

label.pack(ipadx=5, ipady=2)

**def** hide\_tip(self, event=None):

if self.tip\_window:

self.tip\_window.destroy()

self.tip\_window = None

You can use it like this:

root = tk.Tk()

btn = tk.Button(root, text="Hover me")

btn.pack(padx=20, pady=20)

ToolTip(btn, "This is a magical tooltip!")

root.mainloop()

The Dictionary:

tooltips = {

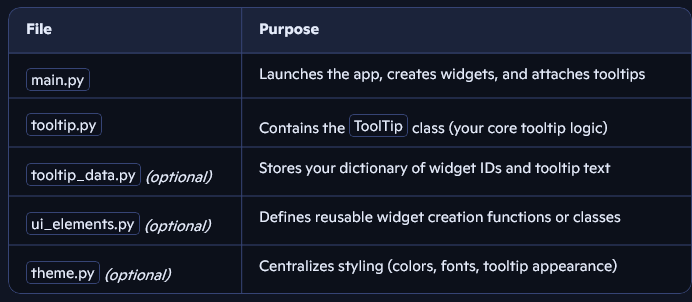
"save\_btn": "Save your progress",

"load\_btn": "Load a previous session",

"exit\_btn": "Exit the magical realm"

}

# Modular File Struction for the Tooltip project



Why Modular?

* **Scalability**: You can add new tooltip types (e.g. animated, delayed, styled) without cluttering your main file.
* **Reusability**: If Carly wants tooltips in another magical app, you can just import tooltip.py.
* **Readability**: Each file has a clear purpose—makes it easier to debug or expand.
* **Tutorial-friendly**: If you’re writing a series, each file can be introduced step-by-step.

🪄 Example: tooltip\_data.py

tooltips = { "save\_btn": "Save your progress", "load\_btn": "Load a previous session", "exit\_btn": "Exit the magical realm" }

Then in main.py:

from tooltip import ToolTip from tooltip\_data import tooltips ToolTip(save\_btn, tooltips["save\_btn"])

Would you like help scaffolding these files or setting up a basic folder structure? I can also help you write a \_\_init\_\_.py if you want to turn this into a mini package.