In [ ]:

```
import tkinter as tk
from tkinter import messagebox
import time

class CountdownTimerApp:
    def __init__(self, root):
        self.root = root
        self.root.title("Countdown Timer")

        # Timer variables
        self.remaining_seconds = 0
        self.timer_running = False
        self.timer_paused = False

        # Create GUI components
        self.create_widgets()

    def create_widgets(self):
        # Input fields for hours, minutes, seconds
        tk.Label(self.root, text="Hours:").grid(row=0, column=0)
        self.hours_entry = tk.Entry(self.root, width=5)
        self.hours_entry.grid(row=0, column=1)

        tk.Label(self.root, text="Minutes:").grid(row=0, column=2)
        self.minutes_entry = tk.Entry(self.root, width=5)
        self.minutes_entry.grid(row=0, column=3)

        tk.Label(self.root, text="Seconds:").grid(row=0, column=4)
        self.seconds_entry = tk.Entry(self.root, width=5)
        self.seconds_entry.grid(row=0, column=5)

        # Start, Pause, Reset buttons
        self.start_button = tk.Button(self.root, text="Start", command=self.start_timer)
        self.start_button.grid(row=1, column=0, columnspan=2)

        self.pause_button = tk.Button(self.root, text="Pause", command=self.pause_timer, state=tk.DISABLED)
        self.pause_button.grid(row=1, column=2, columnspan=2)

        self.reset_button = tk.Button(self.root, text="Reset", command=self.reset_timer)
        self.reset_button.grid(row=1, column=4, columnspan=2)

        # Timer display
        self.timer_label = tk.Label(self.root, text="00:00:00", font=("Helvetica", 48))
        self.timer_label.grid(row=2, column=0, columnspan=6)

    def start_timer(self):
        if self.timer_running:
            return

        # Get the user input
        hours = int(self.hours_entry.get() or 0)
        minutes = int(self.minutes_entry.get() or 0)
        seconds = int(self.seconds_entry.get() or 0)

        self.remaining_seconds = hours * 3600 + minutes * 60 + seconds
        if self.remaining_seconds > 0:
            self.timer_running = True
            self.timer_paused = False
            self.update_timer()
            self.start_button.config(state=tk.DISABLED)
            self.pause_button.config(state=tk.NORMAL)
            self.reset_button.config(state=tk.NORMAL)
        else:
            messagebox.showwarning("Input Error", "Please enter a valid time.")

    def update_timer(self):
        if not self.timer_running:
            return

        if self.remaining_seconds > 0:
            hours, remainder = divmod(self.remaining_seconds, 3600)
            minutes, seconds = divmod(remainder, 60)
            self.timer_label.config(text=f"{hours:02}:{minutes:02}:{seconds:02}")
            self.remaining_seconds -= 1
            self.root.after(1000, self.update_timer)
        else:
            self.timer_label.config(text="00:00:00")
            messagebox.showinfo("Time's Up", "The countdown has finished!")
            self.reset_timer()

    def pause_timer(self):
        if self.timer_running:
            self.timer_running = False
            self.pause_button.config(state=tk.DISABLED)
            self.start_button.config(state=tk.NORMAL)
            self.reset_button.config(state=tk.NORMAL)

    def reset_timer(self):
        self.timer_running = False
        self.timer_paused = False
        self.remaining_seconds = 0
        self.timer_label.config(text="00:00:00")
        self.start_button.config(state=tk.NORMAL)
        self.pause_button.config(state=tk.DISABLED)
        self.reset_button.config(state=tk.DISABLED)

# Create and run the application
root = tk.Tk()
app = CountdownTimerApp(root)
root.mainloop()
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: