Project : Stop Watch / Clock

```python
import tkinter as tk

from datetime import datetime

import time

import threading


class ClockStopwatchApp:

    def __init__(self, root):

        self.root = root

        self.root.title("Clock & Stopwatch")

        self.root.geometry("400x300")



        self.clock_label = tk.Label(root, text="", font=("Helvetica", 16))

        self.clock_label.pack(pady=10)



        self.stopwatch_label = tk.Label(root, text="00:00:00", font=("Helvetica", 24))

        self.stopwatch_label.pack(pady=10)



        button_frame = tk.Frame(root)

        button_frame.pack(pady=5)



        self.start_button = tk.Button(button_frame, text="Start", width=8, command=self.start)

        self.start_button.grid(row=0, column=0, padx=5)



        self.stop_button = tk.Button(button_frame, text="Stop", width=8, command=self.stop)
```

```python
        self.stop_button.grid(row=0, column=1, padx=5)


        self.reset_button = tk.Button(button_frame, text="Reset", width=8,
command=self.reset)
        self.reset_button.grid(row=0, column=2, padx=5)



        self.running = False
        self.counter = 0



        self.update_clock()


    def update_clock(self):
        now = datetime.now().strftime("%H:%M:%S")
        self.clock_label.config(text="Clock: " + now)
        self.root.after(1000, self.update_clock)


    def update_stopwatch(self):
        if self.running:
            self.counter += 1
            time_str = time.strftime('%H:%M:%S', time.gmtime(self.counter))
            self.stopwatch_label.config(text=time_str)
            self.root.after(1000, self.update_stopwatch)


    def start(self):
        if not self.running:
            self.running = True
```

```python
            self.update_stopwatch()

    def stop(self):
        self.running = False

    def reset(self):
        self.running = False
        self.counter = 0
        self.stopwatch_label.config(text="00:00:00")


if __name__ == "__main__":
    root = tk.Tk()
    app = ClockStopwatchApp(root)
    root.mainloop()
```