**Python Minor Project**

**Create A Countdown Timer Using Python**

**Features To Include Reset/ Stop Pause /Resume**

**NAME: TUSHAR NITIN BHANSALI**

# Python program to illustrate a stop watch

# using Tkinter

# importing the required libraries

import tkinter as Tkinter

from datetime import datetime

import time

counter = 66600

running = False

def counter\_label(label):

    def count():

        if running:

            global counter

            # To manage the initial delay.

            if counter == 66600:

                display = "Starting..."

            else:

                tt = datetime.fromtimestamp(counter)

                string = tt.strftime("%H:%M:%S")

                display = string

            label['text'] = display  # Or label.config(text=display)

            # label.after(arg1, arg2) delays by

            # first argument given in milliseconds

            # and then calls the function given as second argument.

            # Generally like here we need to call the

            # function in which it is present repeatedly.

            # Delays by 1000ms=1 seconds and call count again.

            label.after(1000, count)

            counter += 1

    # Triggering the start of the counter.

    count()

# start function of the stopwatch

def Start(label):

    global running

    running = True

    counter\_label(label)

    start['state'] = 'disabled'

    stop['state'] = 'normal'

    reset['state'] = 'normal'

# Stop function of the stopwatch

def Stop():

    global running

    start['state'] = 'normal'

    stop['state'] = 'disabled'

    reset['state'] = 'normal'

    running = False

# Reset function of the stopwatch

def Reset(label):

    global counter

    counter = 66600

    # If rest is pressed after pressing stop.

    if running == False:

        reset['state'] = 'disabled'

        label['text'] = 'Welcome!'

    # If reset is pressed while the stopwatch is running.

    else:

        label['text'] = 'Starting...'

def \_update(self):

    """ Update the label with elapsed time. """

    if not self.\_isPause:

        self.\_elapsedtime = time.time() - self.\_pausedTime - self.\_start

        self.\_setTime(self.\_elapsedtime)

        self.\_timer = self.after(50, self.\_update)

        self.\_pauseStartTime = time.time()

    else:

        self.\_pausedTime = time.time() - self.\_pauseStartTime

root = Tkinter.Tk()

root.title("Stopwatch")

# Fixing the window size.

root.minsize(width=250, height=70)

label = Tkinter.Label(root, text="Welcome!", fg="black",

                      font="Verdana 30 bold")

label.pack()

f = Tkinter.Frame(root)

start = Tkinter.Button(f, text='Start', width=6, command=lambda: Start(label))

stop = Tkinter.Button(f, text='Stop', width=6, state='disabled', command=Stop)

reset = Tkinter.Button(f, text='Reset', width=6,

                       state='disabled', command=lambda: Reset(label))

resume = Tkinter.Button(f, text='resume', width=6,

                        command=lambda: \_update(label))

f.pack(anchor='center', pady=5)

start.pack(side="left")

stop.pack(side="left")

reset.pack(side="left")

resume.pack(side="left")

root.mainloop()

**OUTPUT:**

**INITIAL GUI:  
Graphical user interface, text, application

Description automatically generated**

**AFTER START:**

**Graphical user interface, text, application, email

Description automatically generated**

**STOP:**

**Graphical user interface, application, email

Description automatically generated**

**RESET:**

**Graphical user interface, application

Description automatically generated**