

#M.Vamshidhar Reddy

```
import tkinter as tk
import datetime
import winsound as ws
```

```
class Countdown(tk.Frame):
    def __init__(self, master):
        super().__init__(master)
        self.create_widgets()
        self.show_widgets()
        self.seconds_left = 0
        self._timer_on = False
```

```
    def show_widgets(self):
        self.label.pack()
        self.entry.pack()
        self.start.pack()
```

```
self.stop.pack()  
self.reset.pack()
```

```
def create_widgets(self):  
    self.label = tk.Label(self, text="Enter  
the time in seconds.")  
    self.entry = tk.Entry(self,  
justify="center")  
    self.entry.focus_set()  
    self.reset = tk.Button(self, text="Reset  
Timer",  
  
command=self.reset_button)  
    self.stop = tk.Button(self, text="stop  
Timer",  
  
command=self.stop_button)  
    self.start = tk.Button(self, text="start  
Timer",  
  
command=self.start_button)  
  
def countdown(self):  
    self.label["text"] =
```

```
self.convert_seconds_left_to_time()
```

```
    if self.seconds_left:
```

```
        self.seconds_left -= 1
```

```
        self._timer_on = self.after(1000,  
self.countdown)
```

```
    else:
```

```
        self._timer_on = False
```

```
        ws.PlaySound("Alarm clock sound",  
ws.SND_FILENAME)
```

```
def reset_button(self):
```

```
    self.seconds_left = 0
```

```
    self.stop_timer()
```

```
    self._timer_on = False
```

```
    self.label["text"] = "Enter the time in  
seconds."
```

```
    self.start.forget()
```

```
    self.stop.forget()
```

```
    self.reset.forget()
```

```
    self.start.pack()
```

```
    self.stop.pack()
```

```
    self.reset.pack()
```

```
def stop_button(self):  
    self.seconds_left = int(self.entry.get())  
    self.stop_timer()
```

```
def start_button(self):  
    self.seconds_left = int(self.entry.get())  
    self.stop_timer()  
    self.countdown()  
    self.start.forget()  
    self.stop.forget()  
    self.reset.forget()  
    self.start.pack()  
    self.stop.pack()  
    self.reset.pack()
```

```
def stop_timer(self):  
    if self._timer_on:  
        self.after_cancel(self._timer_on)  
        self._timer_on = False
```

```
def convert_seconds_left_to_time(self):  
    return
```

```
datetime.timedelta(seconds=self.seconds_left)
```

```
if __name__ == "__main__":
```

```
    root = tk.Tk()
```

```
    root.resizable(False, False)
```

```
    countdown = Countdown(root)
```

```
    countdown.pack()
```

```
    root.mainloop()
```



