

Name:SHIVENDRA PRATAP SINGH

Task id: **TKSD2**

My github repo link: <https://github.com/LETME2X/KALPANA>

To run this program the system must have : 1)Tkinter module
2)PIL

This project is based on GUI model used in Tkinter

I have found this following errors in the flowchart.

1)State 1 : If rocket altitude is = 725 ,the rocket will stay there to enter in State 2

2)State 1 : Rocket will not rise above 725 as 725 is max limit of altitude.

3)State 4A : In the altitude between 500 and 400 as payload 1 is released in starting of state 4 A ,it will not released again.

4)State 4 B : Again In the altitude between 400 and 5 as payload 2 is released in starting of state 4 B ,it will not released again.

I have done the code by fixing all the errors.

Code:

```
from tkinter import *          #importing tkinter in python

from tkinter.ttk import *

from PIL import ImageTk, Image  #importing PIL image module


win = Tk()                    #Creating window

counter = 0.000               #Assigning initial value to all the variables used....

start = False

state = [0,'x']

payload_alt = [0]*2

data = [0]*5

cansat_released = False

time = 0.0

act_buzzer = False

act_camera = False

sys_calibrated = False

received_CX = False

received_ST = False
```

```
rocket_rising = False
```

```
win.geometry("720x600") #Setting geometry of created window  
img = Image.open("kalpana.png") #importing background png file  
bg = img.resize((720, 600), Image.ANTIALIAS) #Resizing the png  
new_img = ImageTk.PhotoImage(bg)
```

```
canvas = Canvas(win, width=700, height=3500) #For printing image  
canvas.pack(expand=True)  
canvas.create_image(0, 0, image=new_img, anchor='nw')
```

```
def isittrue(func):  
    s = str("")  
    if func:          #Calling function of all the states  
        s = "YES"  
    else:  
        s = "NO"  
    return s
```

```
def isiton(func):  
    s = str("")  
    if func:  
        s = "ON"  
    else:  
        s = "OFF"  
    return s
```

```
def counter_label(label):    #function of time  
    def count():  
        global counter  
        counter += 0.001
```

```
s = str(counter)

label.config(text=s[:7])

label.after(1, count)    #after used for delay in time
```

```
count()
```

```
height = 0
```

```
t = 0
```

```
def alt(label,newWindow):
```

```
    def count():    #function for altitude
```

```
        global height
```

```
        global t
```

```
        global sys_calibrated
```

```
        global received_CX
```

```
        global received_ST
```

```
        global act_camera
```

```
        global act_buzzer
```

```
        global cansat_released
```

```
        text = Label(newWindow, text="CX : " + isiton(received_CX))
```

```
        text.place(x=50, y=475)
```

```
        text = Label(newWindow, text="Camera Activated : " + isittrue(act_camera))
```

```
        text.place(x=50, y=525)
```

```
        text = Label(newWindow, text="CanSat Released : " + isittrue(cansat_released))
```

```
        text.place(x=50, y=500)
```

```
        text = Label(newWindow, text="Buzzer : " + isiton(act_buzzer))
```

```
        text.place(x=50, y=550)
```

```
    if height<5:
```

```

text = Label(newWindow, text=" State : 0 ",font=('calibre', 15, 'bold'))
text.place(x=200, y=180)
if height == 1:
    text = Label(newWindow, text=" System Calibrating... ", font=('calibre', 10, 'bold'))
    text.place(x=300, y=180)
elif height == 2:
    text = Label(newWindow, text=" Receiving CX ON command...", font=('calibre', 10, 'bold'))
    text.place(x=300, y=205)
    if not received_CX:
        received_CX = True
elif height==4:
    text = Label(newWindow, text=" Receiving ST command...", font=('calibre', 10, 'bold'))
    text.place(x=300, y=230)
    if not sys_calibrated:
        sys_calibrated = True
    if not received_ST:
        received_ST = True
elif height>=5 and height<725:
    text = Label(newWindow, text=" State : 1 ", font=('calibre', 15, 'bold'))
    text.place(x=200, y=255)
    if height == 6:
        text = Label(newWindow, text=" Rocket Rising...", font=('calibre', 10, 'bold'))
        text.place(x=300, y=255)
    elif height == 10:
        text = Label(newWindow, text=" Collecting Data....", font=('calibre', 10, 'bold'))
        text.place(x=300, y=280)
elif t == 725:
    text = Label(newWindow, text=" State : 2 ", font=('calibre', 15, 'bold'))
    text.place(x=200, y=305)
    text = Label(newWindow, text=" Releasing CanSat...", font=('calibre', 10, 'bold'))
    text.place(x=300, y=305)

```

```

if not cansat_released:

    cansat_released = True

elif t>500:

    if t== 680:

        text = Label(newWindow, text=" State : 3 ", font=('calibre', 15, 'bold'))

        text.place(x=200, y=335)

    elif t== 630:

        text = Label(newWindow, text="Activating Camera... ", font=('calibre', 10, 'bold'))

        text.place(x=300, y=335)

    if not act_camera:

        act_camera = True

elif t>400 and t<500:

    text = Label(newWindow, text=" State : 4A ", font=('calibre', 15, 'bold'))

    text.place(x=200, y=370)

    if t == 480:

        text = Label(newWindow, text="PayLoad 1 Released ", font=('calibre', 10, 'bold'))

        text.place(x=320, y=370)

    elif t == 430:

        text = Label(newWindow, text="PayLoad Data... ", font=('calibre', 10, 'bold'))

        text.place(x=320, y=395)

elif t>5 and t<400:

    text = Label(newWindow, text=" State : 4B ",font=('calibre', 15, 'bold'))

    text.place(x=200, y=420)

    if t == 380:

        text = Label(newWindow, text="PayLoad 2 Released ", font=('calibre', 10, 'bold'))

        text.place(x=320, y=420)

    elif t == 290:

        text = Label(newWindow, text="PayLoad Data... ", font=('calibre', 10, 'bold'))

        text.place(x=320, y=445)

elif t<5 and t>0:

```

```

text = Label(newWindow, text=" State : 5 ", font=('calibre', 15, 'bold'))
text.place(x=200, y=470)

if t==4:

    text = Label(newWindow, text="Deactivating Camera... ", font=('calibre', 10, 'bold'))
    text.place(x=300, y=470)

    if act_camera:

        act_camera = False

elif t==3:

    text = Label(newWindow, text=" Activating Buzzer...", font=('calibre', 10, 'bold'))
    text.place(x=300, y=495)

    if not act_buzzer:

        act_buzzer = True

elif t==2:

    text = Label(newWindow, text=" Receiving CX OFF command...", font=('calibre', 10, 'bold'))
    text.place(x=300, y=520)

elif t == 1:

    text = Label(newWindow, text=" Telemetry OFF !!", font=('calibre', 10, 'bold'))
    text.place(x=300, y=550)

```

```

if height<=10: #condition for ascent of rocket

```

```

    height += 1
    s = str(height) + " m"
    label.config(text=s)
    label.after(1000, count)

```

```

elif height>=10 and height<20:

```

```

    height += 1
    s = str(height) + " m"
    label.config(text=s)
    label.after(500, count)

```

```
elif height >= 20 and height < 100:
```

```
    height += 1
```

```
    s = str(height) + " m"
```

```
    label.config(text=s)
```

```
    label.after(100, count)
```

```
elif height >= 100 and height < 660:
```

```
    height += 1
```

```
    s = str(height) + " m"
```

```
    label.config(text=s)
```

```
    label.after(10, count)
```

```
elif height >= 660 and height < 724:
```

```
    height += 1
```

```
    s = str(height) + " m"
```

```
    label.config(text=s)
```

```
    label.after(100, count)
```

```
elif height >= 724 and height < 725:
```

```
    height += 1
```

```
    s = str(height) + " m"
```

```
    label.config(text=s)
```

```
    label.after(2000, count)
```

```
    t = height
```

```
elif t <= 725 and t > 720: #condition for descent of rocket
```

```
    t -= 1
```

```
    s = str(t) + "m"
```

```
    label.config(text=s)
```

```
    label.after(500, count)
```

```
elif t <= 720 and t > 700:
```

```
    t -= 1
```

```
    s = str(t) + "m"
```

```
    label.config(text=s)
```

```
    label.after(100, count)
```

```
elif t<=700 and t>500:
```

```
    t -= 1
```

```
    s = str(t) + "m"
```

```
    label.config(text=s)
```

```
    label.after(50, count)
```

```
elif t<=500 and t>5:
```

```
    t -= 1
```

```
    s = str(t) + "m"
```

```
    label.config(text=s)
```

```
    label.after(10, count)
```

```
elif t<=5 and t>0:
```

```
    t -= 1
```

```
    s = str(t) + "m"
```

```
    label.config(text=s)
```

```
    label.after(1000, count)
```

```
elif t >= -20 and t <= 0:
```

```
    t -= 1
```

```
    print(t)
```

```
    if t == -20 :
```

```
        exit()
```

```
    label.after(100, count)
```

```
count()
```

```
def openNewWindow(): #for toplevel window
```

```
    newWindow = Toplevel(win)
```

```
    win.withdraw()
```

```
    newWindow.geometry("720x600")
```

```
    canva = Canvas(newWindow, width=700, height=3500)
```

```
    canva.pack(expand=True)
```



```
canva.create_image(0, 0, image=new_img, anchor='nw')
```

```
# texts on the toplevel window
```

```
text = Label(newWindow, text="Time Elapsed :")
```

```
text.place(x=550, y=475)
```

```
text = Label(newWindow, text="Altitude :")
```

```
text.place(x=550, y=500)
```

```
text = Label(newWindow, text="Air Pressure :")
```

```
text.place(x=550, y=525)
```

```
text = Label(newWindow, text="Temperature :")
```

```
text.place(x=550, y=550)
```

```
label = Label(newWindow)
```

```
label.place(x=640,y=475)
```

```
counter_label(label)
```

```
label2 = Label(newWindow)
```

```
label2.place(x=640, y=500)
```

```
alt(label2,newWindow)
```

```
button = Button(win, text='<<<LAUNCH>>>', width=25, command=openNewWindow)
```

```
button.place(x=280, y=280)
```

```
mainloop()
```