

```

1  # Import required libraries
2  from tkinter import *
3  from tkinter import ttk
4  from tkinter import filedialog
5  from tkinter import messagebox
6  from ttkthemes import ThemedStyle
7  import serial
8  import serial.tools.list_ports
9  import datetime
10 import threading
11 import multiprocessing
12 import os
13 import csv
14
15 # Debugging variable; if set to 1, debugging messages will print to the console
16 dbg = 0
17
18 # Setup main window
19 gRoot = Tk()
20 gRoot.config(bg="white")
21 gRoot.geometry("1080x640")
22 gRoot.title("Black Box Connect")
23
24 # Style configuration
25 sty = ThemedStyle(gRoot)
26 sty.set_theme('radiance')
27
28 # Configure main window grid
29 gRoot.columnconfigure(0,weight=1)
30 gRoot.rowconfigure(0,weight=1)
31
32 # Create a frame for connection settings
33 gFrame = ttk.LabelFrame(gRoot,text="Connection Setting",padding=10, style='TFrame')
34 gFrame.grid(column=1,row=1, sticky=(W,E))
35
36 # Create a frame for command buttons
37 gFrameCmd = ttk.LabelFrame(gRoot,text="List of commands",padding=10, width=130 , style='TFrame')
38 gFrameCmd.grid(column=1,row=3, sticky=(N, S, E, W))
39
40 # Create a frame for COM messages
41 gFrame21 = ttk.Frame(gRoot,padding=10, style='TFrame')
42 gFrame21.grid(column=1,row=2, sticky=(W, E, N))
43 gRoot.resizable(0,0)
44
45 # Configure weight of columns and rows for gFrame
46 for x in range(10):
47     gFrame.columnconfigure(x,weight = x)
48     gFrame.rowconfigure(x,weight = x)
49
50 # Add labels to frames
51 labell=ttk.Label(gFrame, text = "Serial Console")
52 labell.grid(column=2,row=0)
53
54 # Configure style for label2
55 sty.configure("label2.TLabel",borderwidth=4,relief="ridge",foreground="red",ipadx=10)
56 label2=ttk.Label(gFrame,sty="label2.TLabel", text = "Select Com Port")
57 label2.grid(column=1,row=1, sticky = (N,E,W,S))
58
59 # Fetch available COM ports
60 ports = serial.tools.list_ports.comports()
61 com_port_list = [com[0] for com in ports]
62 com_port_list.insert(0,"Select an Option")
63 if dbg == 1:
64     print(com_port_list)
65
66 # Variable initializations for dropdowns
67 com_value_inside = StringVar()
68 baud_value_inside = StringVar()
69
70 # Baud rate dropdown menu
71 baud_menu = ttk.OptionMenu(gFrame,baud_value_inside,"select baud rate","9600",

```

```

72         '19200','28800','38400','57600','76800')
73     baud_menu.grid(column=3, row=1, sticky = (E))
74
75     # Function to update the available COM port list
76     def com_port_list_update():
77         # ... [rest of the function]
78
79     # Function to print serial data
80     def serial_print():
81         # ... [rest of the function]
82
83     # Serial command functions
84     def CONFIG():
85         # ... [rest of the function]
86     # ... [similar command functions]
87
88     # Function to manage serial connection
89     def serial_connect(com_port,baud_rate):
90         # ... [rest of the function]
91
92     counter1 = 0;
93
94     # More command functions
95     def SHUTDOWN():
96         # ... [rest of the function]
97     # ... [similar command functions]
98
99     # Function to close the serial connection
100    def serial_close():
101        # ... [rest of the function]
102
103    # Function to close the application and the serial connection
104    def power_off():
105        # ... [rest of the function]
106
107    # Function to fetch and display the user's selections
108    def submit_value():
109        # ... [rest of the function]
110
111    # Listbox to display serial data
112    Lb2 = Listbox(gFrame21, width = 130, height=20, xscrollcommand = 1)
113    Lb2.grid(column=1, row = 1, sticky = W+E)
114    # ... [rest of the scrollbar configuration]
115
116    # Function to clear the listbox
117    def clear_listbox():
118        Lb2.delete(0,END)
119
120    # Command buttons for the gFrameCmd frame
121    subBtn = ttk.Button(gFrameCmd,text="GNSS live data",command = GLIVE, width=15)
122    # ... [other button declarations]
123
124    # Command buttons for the gFrame frame
125    subBtn = ttk.Button(gFrame,text="submit",command = submit_value)
126    # ... [other button declarations]
127
128    # Menu function that does nothing (possibly for further development or debugging)
129    def donothing():
130        # ... [rest of the function]
131
132    # Function to save the displayed data to a file
133    def save():
134        # ... [rest of the function]
135
136    # Function to display about info (GitHub link in this case)
137    def About_me():
138        # ... [rest of the function]
139
140    # Menu configuration
141    menubar = Menu(gRoot)
142    # ... [rest of the menu configuration]
143

```

```
144     # Protocol for window close button
145     gRoot.protocol("WM_DELETE_WINDOW", power_off)
146
147     # Attach the menu to the main window
148     gRoot.config(menu=menubar)
149
150     # Start the GUI loop
151     gRoot.mainloop()
152
```