Arcilla, Andrew Sean D.

BSCS C204

FINALS LAB TASK 4

Problem:

Finals Lab Task 4. Python GUI using TKINTER

Note: Write your code following OOP code construct, you may use the attached simpleCalc.py program as guide.

Instructions: READ AND UNDERSTAND THE PROBLEM FIRST BEFORE DOING THE ACTUAL PROGRAM.

1. Design the form below
2. Problem Statement: The cost of a long Distance call is based on the destination, the time of day the call was made, as well as the distance of the call. The rates as as follows:

DAYTIME CALLS		NIGHTIME CALLS	
1. American Region	P 50 every 3 minutes	1. American Region	P 45 every 3 minutes
2. Asian Region	P 30 every 2 minutes	2. Asian Region	P 27 every 2 minutes
3. African Region	P 40 every 3 minutes	3. African Region	P 36 every 3 minutes
4. European Region	P 35 every 2 minutes	4. European Region	P 30 every 2 minutes

3. Make a program that will Allow the user to **Select Destination Code (between 1 – 4)** using ComboBox widget, A Time Code using radio buttons, And the Duration Of The Call in minutes and output the **TOTAL CHARGE**. – Validate user inputs by using **TRY EXCEPT block – Only numeric values are accepted.**

4. **Compute Button** should compute for the **TOTAL CHARGE**.

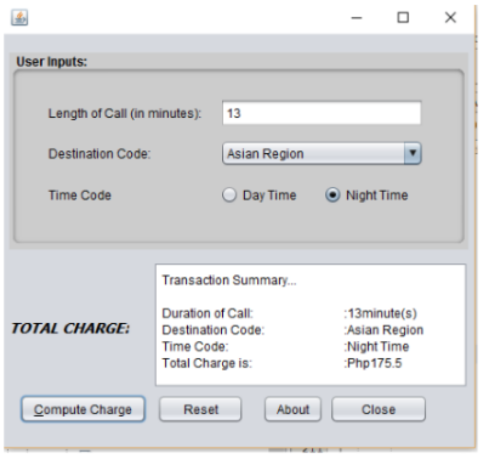
   4.1 Computations should be based on the table rates shown above. (The total charge is based on **Length of Calls**, **Destination Code** and **Time Code**)

   4.2. You may use the get () method of the comboBox to capture the selected option in your comboBox

5. **Reset Button** should clear the Radio Button Selection and the Text field entries should be cleared as well

6. **About button** should display a dialog with the message: "Hello I'm your Name"

7. See sample output below:


User Inputs:

Length of Call (in minutes): 13

Destination Code: Asian Region

Time Code: ( ) Day Time [✓ Night Time]

TOTAL CHARGE:

```
Transaction Summary...

Duration of Call:          :13minute(s)
Destination Code:          :Asian Region
Time Code:                 :Night Time
Total Charge is:           :Php175.5
```

Rubrics: Form Design and Layout : 10 points
Program Correctness : 40 points (Reset – 5 pts., About – 5 pts. , Compute – 30 pts.)

Source Code:

```python
from tkinter import ttk, messagebox
import tkinter as tk
class CallRateCalc:

    DAYTIME_RATES = {
        1: 50 / 3,
        2: 30 / 2,
        3: 20 / 4,
        4: 35 / 2
    }
    NIGHTTIME_RATES = {
        1: 45 / 3,
        2: 27 / 2,
        3: 15 / 3,
        4: 30 / 2
    }
    def compute_charge(self, code, minutes, time_code):
        if time_code == "Day":
            rate = self.DAYTIME_RATES[code]
        else:
            rate = self.NIGHTTIME_RATES[code]

        total = minutes * rate
        return round(total, 2)





class CallGUI:
    def __init__(self, master):
        self.master = master
        master.title("Long Distance Call Calculator")
        master.geometry("500x350")

        self.calculator = CallRateCalc()


        tk.Label(master, text="Length of Call (minutes):").place(x=30, y=30)
        self.entry_minutes = tk.Entry(master, width=20)
        self.entry_minutes.place(x=200, y=30)

        tk.Label(master, text="Destination Code:").place(x=30, y=70)
        self.combo_dest = ttk.Combobox(
            master,
            values=["1 - American Region", "2 - Asian Region",
                    "3 - African Region", "4 - European Region"],
            state="readonly",
            width=27
        )
        self.combo_dest.place(x=160, y=70)


        tk.Label(master, text="Time Code:").place(x=30, y=120)
        self.time_var = tk.StringVar()
        tk.Radiobutton(master, text="Day Time", variable=self.time_var,
                       value="Day").place(x=150, y=120)
        tk.Radiobutton(master, text="Night Time", variable=self.time_var,
                       value="Night").place(x=250, y=120)



        tk.Button(master, text="Compute", width=12,
                  command=self.compute_charge).place(x=50, y=170)

        tk.Button(master, text="Reset", width=12,
                  command=self.reset_fields).place(x=160, y=170)

        tk.Button(master, text="About", width=12,
                  command=self.show_about).place(x=270, y=170)

        self.output_box = tk.Text(master, width=55, height=10)
        self.output_box.place(x=30, y=210)



    def compute_charge(self):
        try:
            minutes = float(self.entry_minutes.get())
            if minutes <= 0:
                raise ValueError

        except:
            messagebox.showerror("Input Error", "Please enter a valid number
for minutes.")
            return

        if not self.combo_dest.get():
            messagebox.showwarning("Missing Selection", "Select a destination
code.")
            return

        if not self.time_var.get():
            messagebox.showwarning("Missing Selection", "Select a time
code.")
            return

        code = int(self.combo_dest.get()[0])
        time_code = self.time_var.get()

        total = self.calculator.compute_charge(code, minutes, time_code)




        self.output_box.delete("1.0", tk.END)
        self.output_box.insert(tk.END, "=== Transaction Summary ===\n")
        self.output_box.insert(tk.END, f"Duration of Call: {minutes}
minutes\n")
        self.output_box.insert(tk.END, f"Destination Code:
{self.combo_dest.get()}\n")
```

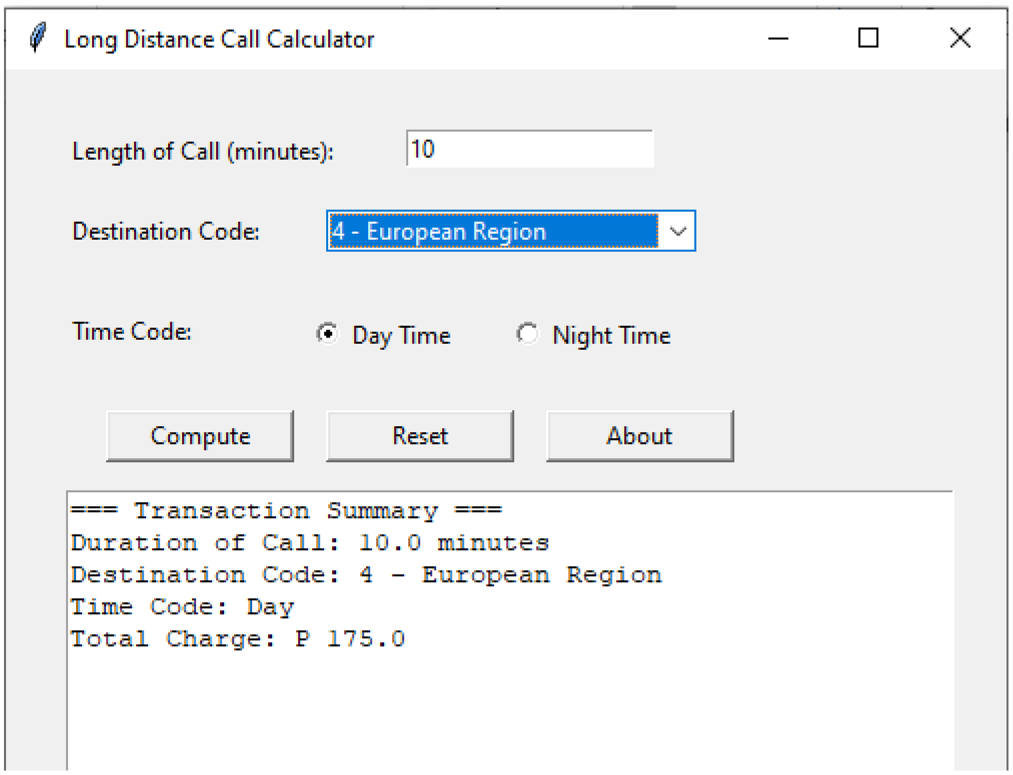
```
        self.output_box.insert(tk.END, f"Time Code: {time_code}\n")
        self.output_box.insert(tk.END, f"Total Charge: P {total}\n")

    def reset_fields(self):
        self.entry_minutes.delete(0, tk.END)
        self.combo_dest.set("")
        self.time_var.set("")
        self.output_box.delete("1.0", tk.END)
    def show_about(self):
        messagebox.showinfo("About", "Hello! I'm your helper.")




root = tk.Tk()
app = CallGUI(root)
root.mainloop()
```

Sample Output:



```
=== Transaction Summary ===
Duration of Call: 10.0 minutes
Destination Code: 4 - European Region
Time Code: Day
Total Charge: P 175.0
```