

## 7OOP FINALS LAB TASK 4

### I. PROBLEM

**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		NIGHTTIME 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:

**Rubrics: Form Design and Layout : 10 points**

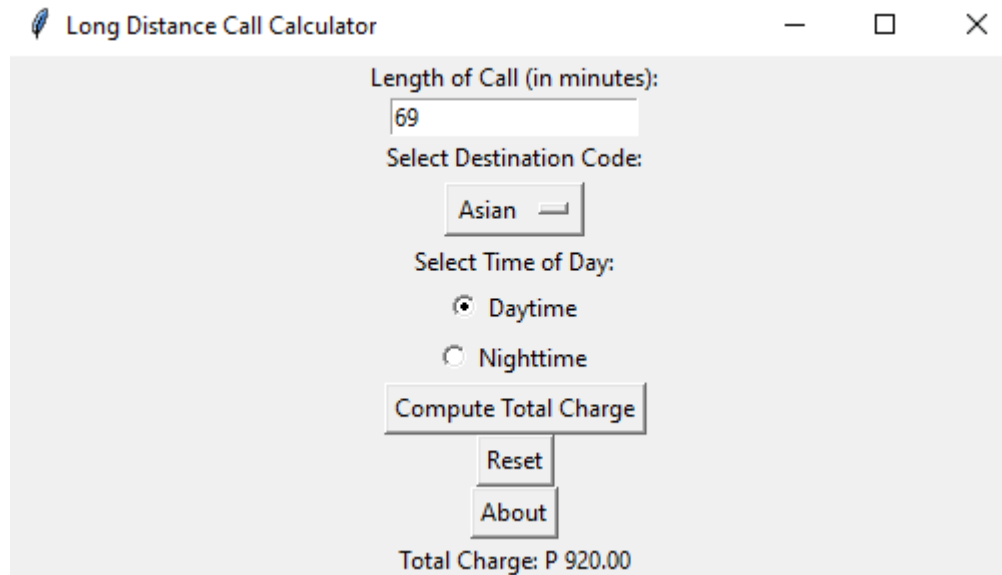
**Program Correctness : 40 points (Reset – 5 pts., About – 5 pts. , Compute – 30 pts.)**

## II. SOURCE CODE

```
1 import tkinter as tk
2 from tkinter import messagebox
3
4 usage
5 def compute_total_charge():
6     try:
7         length_of_call = int(length_entry.get())
8         destination = destination_var.get()
9         time_of_day = time_var.get()
10
11         if length_of_call <= 0:
12             raise ValueError("Length of call must be a positive number")
13
14         rates = {
15             'American': {'Day': 50, 'Night': 15},
16             'Asian': {'Day': 40, 'Night': 27},
17             'African': {'Day': 45, 'Night': 30},
18             'European': {'Day': 35, 'Night': 20}
19         }
20
21         rate = rates[destination][time_of_day]
22
23         total_charge = (length_of_call / 3) * rate
24
25         result_label.config(text=f"Total Charge: P {total_charge:.2f}")
26     except ValueError as e:
27         messagebox.showerror(title="Invalid Input", message=f"Error: {e}")
28
29 usage
30 def reset_fields():
31     length_entry.delete(0, tk.END)
32     result_label.config(text="Total Charge: P 0.00")
33     destination_var.set("American")
34     time_var.set("Day")
35
36 usage
37 def show_about():
38     messagebox.showinfo(title="About", message="Hello! I'm Your Assistance.")
```

```
37 root = tk.Tk()
38 root.title("Long Distance Call Calculator")
39
40 tk.Label(root, text="Length of Call (in minutes):").pack()
41 length_entry = tk.Entry(root)
42 length_entry.pack()
43
44 destination_var = tk.StringVar(value="American")
45 tk.Label(root, text="Select Destination Code:").pack()
46 destination_menu = tk.OptionMenu(root, destination_var, values="American", values="Asian", "African", "European")
47 destination_menu.pack()
48
49 time_var = tk.StringVar(value="Day")
50 tk.Label(root, text="Select Time of Day:").pack()
51 daytime_button = tk.Radiobutton(root, text="Daytime", variable=time_var, value="Day")
52 nighttime_button = tk.Radiobutton(root, text="Nighttime", variable=time_var, value="Night")
53 daytime_button.pack()
54 nighttime_button.pack()
55
56 compute_button = tk.Button(root, text="Compute Total Charge", command=compute_total_charge)
57 compute_button.pack()
58
59 reset_button = tk.Button(root, text="Reset", command=reset_fields)
60 reset_button.pack()
61
62 about_button = tk.Button(root, text="About", command=show_about)
63 about_button.pack()
64
65 result_label = tk.Label(root, text="Total Charge: P 0.00")
66 result_label.pack()
67
68 root.mainloop()
69
```

### III. SAMPLE OUTPUT



Long Distance Call Calculator

Length of Call (in minutes):  
69

Select Destination Code:  
Asian

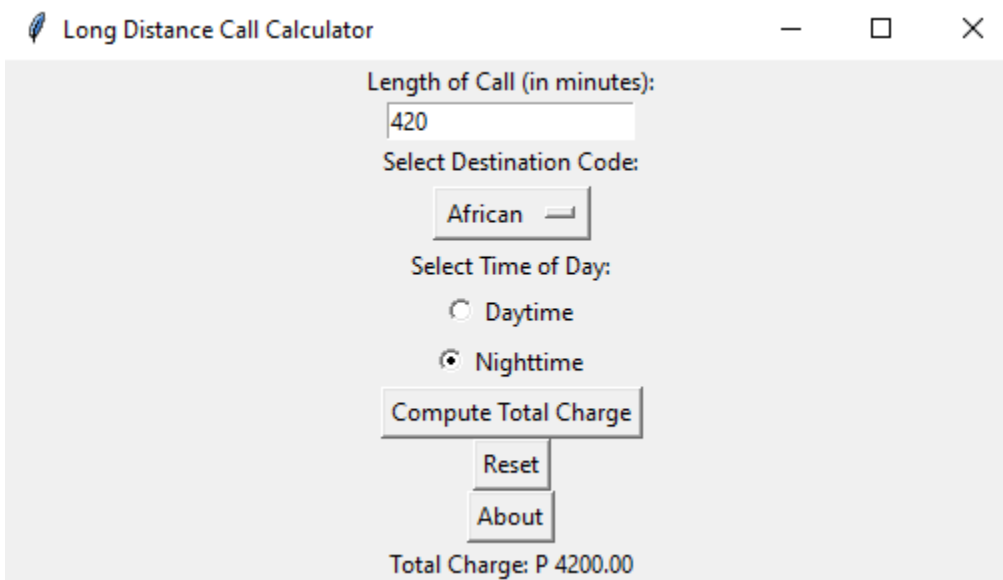
Select Time of Day:  
☒ Daytime  
☐ Nighttime

Compute Total Charge

Reset

About

Total Charge: P 920.00



Long Distance Call Calculator

Length of Call (in minutes):  
420

Select Destination Code:  
African

Select Time of Day:  
☐ Daytime  
☒ Nighttime

Compute Total Charge

Reset

About

Total Charge: P 4200.00

Long Distance Call Calculator

Length of Call (in minutes):  
80

Select Destination Code:  
African

Select Time of Day:  
☒ Daytime  
☐ Nighttime

Compute Total Charge

Reset

About

Total Charge: P 1200.00

Long Distance Call Calculator

Length of Call (in minutes):  
109

Select Destination Code:  
European

Select Time of Day:  
☐ Daytime  
☒ Nighttime

Compute Total Charge

Reset

About

Total Charge: P 726.67

