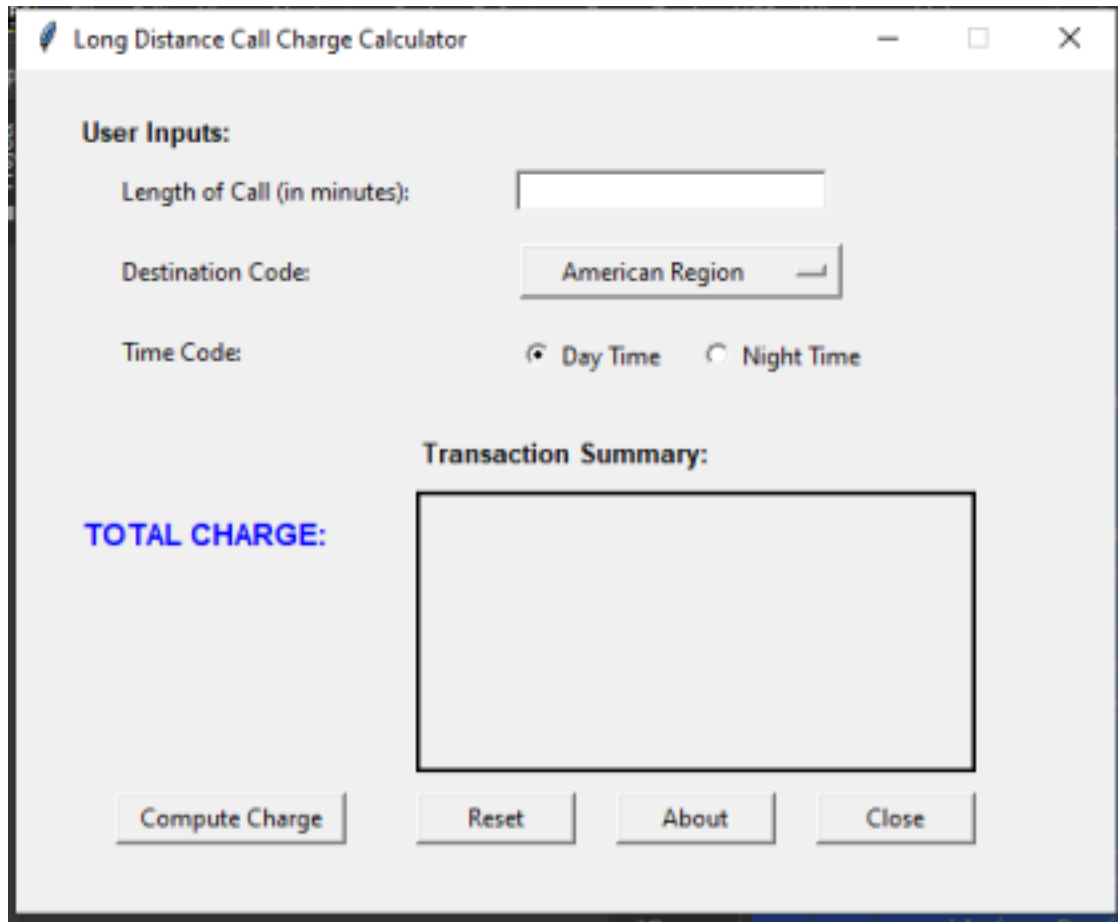


Coronel, Yshian M. 12/06/2025 C204 700P

Finals Lab Task 4. Python GUI using TKINTER

UI Menu



The screenshot shows a Python GUI titled "Long Distance Call Charge Calculator". The window has a standard title bar with minimize, maximize, and close buttons. The main content area is divided into two sections: "User Inputs:" and "Transaction Summary:". Under "User Inputs:", there are three fields: "Length of Call (in minutes):" with a text input box, "Destination Code:" with a dropdown menu showing "American Region", and "Time Code:" with two radio buttons, "Day Time" (selected) and "Night Time". Below these is the "Transaction Summary:" section, which contains a large empty rectangular box. To the left of this box, the text "TOTAL CHARGE:" is displayed in blue. At the bottom of the window, there are four buttons: "Compute Charge", "Reset", "About", and "Close".

Sample Output

Long Distance Call Charge Calculator

User Inputs:

Length of Call (in minutes):

Destination Code:

Time Code: ☐ Day Time ☒ Night Time

Transaction Summary:

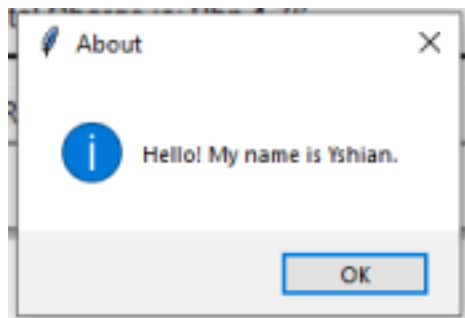
TOTAL CHARGE:

Duration of Call: 13.0 minute(s)

Destination Code: Asian Region

Time Code: Night Time

Total Charge is: Php 175.50



python_gui_using_tinker.py (Source Code)

```
from tkinter import *
from tkinter import messagebox
```

```
class CallChargeCalculator:
    def __init__(self, win):
        self.daytime_rates = {
            'American Region': (50, 3),
```

```
'Asian Region': (30, 2),  
'African Region': (40, 3),  
'European Region': (35, 2)  
}
```

```
self.nighttime_rates = {  
'American Region': (45, 3),  
'Asian Region': (27, 2),  
'African Region': (36, 3),  
'European Region': (30, 2)  
}
```

```
self.lbl_header = Label(win, text='User Inputs:', font=('Arial', 10, 'bold'))  
self.lbl_header.place(x=30, y=20)
```

```
self.lbl_length = Label(win, text='Length of Call (in minutes):')  
self.lbl_length.place(x=50, y=50)  
self.entry_length = Entry(win, bd=2, width=25)  
self.entry_length.place(x=250, y=50)
```

```
self.lbl_destination = Label(win, text='Destination Code:')  
self.lbl_destination.place(x=50, y=90)
```

```
self.destination_var = StringVar()  
self.destination_options = ['American Region', 'Asian Region', 'African Region',  
'European Region']  
self.destination_var.set(self.destination_options[0])
```

```
self.combo_destination = OptionMenu(win, self.destination_var,  
*self.destination_options)  
self.combo_destination.config(width=20)  
self.combo_destination.place(x=250, y=85)
```

```
self.lbl_timecode = Label(win, text='Time Code:')  
self.lbl_timecode.place(x=50, y=130)  
self.time_var = StringVar()  
self.time_var.set('Day Time')
```

```
self.radio_day = Radiobutton(win, text='Day Time', variable=self.time_var,  
value='Day Time')  
self.radio_day.place(x=250, y=130)
```

```
self.radio_night = Radiobutton(win, text='Night Time', variable=self.time_var,  
value='Night Time')  
self.radio_night.place(x=340, y=130)
```

```

self.lbl_summary = Label(win, text='Transaction Summary:', font=('Arial', 10,
'bold'))
self.lbl_summary.place(x=200, y=180)

self.lbl_total = Label(win, text='TOTAL CHARGE:', font=('Arial', 11, 'bold'),
fg='blue')
self.lbl_total.place(x=30, y=220)

# Summary Display Frame
self.frame_summary = Frame(win, bd=2, relief='solid', width=280, height=140)
self.frame_summary.place(x=200, y=210)
self.frame_summary.pack_propagate(False)

self.lbl_duration = Label(self.frame_summary, text='', anchor='w',
justify='left', font=('Arial', 9))
self.lbl_duration.place(x=10, y=5)

self.lbl_dest = Label(self.frame_summary, text='', anchor='w', justify='left',
font=('Arial', 9))
self.lbl_dest.place(x=10, y=40)

self.lbl_time = Label(self.frame_summary, text='', anchor='w', justify='left',
font=('Arial', 9))
self.lbl_time.place(x=10, y=75)

self.lbl_charge = Label(self.frame_summary, text='', anchor='w', justify='left',
font=('Arial', 9))
self.lbl_charge.place(x=10, y=110)

self.btn_compute = Button(win, text='Compute Charge',
command=self.compute_charge, width=15)
self.btn_compute.place(x=50, y=360)

self.btn_reset = Button(win, text='Reset', command=self.reset_fields, width=10)
self.btn_reset.place(x=200, y=360)
self.btn_about = Button(win, text='About', command=self.show_about, width=10)
self.btn_about.place(x=300, y=360)

self.btn_close = Button(win, text='Close', command=win.quit, width=10)
self.btn_close.place(x=400, y=360)

def compute_charge(self):
    try:
        length = float(self.entry_length.get())

```

```

if length <= 0:
    messagebox.showerror('Invalid Input', 'Length of call must be greater than 0')
    return

destination = self.destination_var.get()
time_code = self.time_var.get()

if time_code == 'Day Time':
    rate, interval = self.daytime_rates[destination]
else:
    rate, interval = self.nighttime_rates[destination]

total_charge = (length / interval) * rate

self.lbl_duration.config(text=f"Duration of Call: {length} minute(s)")
self.lbl_dest.config(text=f"Destination Code: {destination}")
self.lbl_time.config(text=f"Time Code: {time_code}")
self.lbl_charge.config(text=f"Total Charge is: Php {total_charge:.2f}")

except ValueError:
    messagebox.showerror('Invalid Input', 'Please enter a valid numeric value for length of call')
except Exception as e:
    messagebox.showerror('Error', f'An error occurred: {str(e)}')

def reset_fields(self):
    self.entry_length.delete(0, END)
    self.destination_var.set(self.destination_options[0])
    self.time_var.set('Day Time')
    self.lbl_duration.config(text='')
    self.lbl_dest.config(text='')
    self.lbl_time.config(text='')
    self.lbl_charge.config(text='')
def show_about(self):
    messagebox.showinfo('About', "Hello I'm your Name")

if __name__ == '__main__':
    window = Tk()
    calculator = CallChargeCalculator(window)
    window.title('Long Distance Call Charge Calculator')
    window.geometry("550x420+10+10")
    window.resizable(False, False)
    window.mainloop()

```