UNIVERSITY INSTITUTE of COMPUTING
Asia's Fastest Growing University

NAAC GRADE A+
Accredited University

Café Management System

Project Report
Python programming (24CAH-606)

Submitted by
Simranjeet kaur (24MCA20075)

in partial fulfilment for the award of the degree of

Master of Computer Application

CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

Chandigarh University

Aug 2024 – Nov 2024

UNIVERSITY INSTITUTE of COMPUTING
Asia's Fastest Growing University

NAAC GRADE A+
Accredited University

Café Management System using Tkinter

Project Report

Introduction

This project aims to develop a **simple graphical user interface (GUI)** application for managing a café using Python's **Tkinter** library. It allows users to input the quantity of items such as coffee, tea, and sandwiches, calculate the total bill, and display a receipt. This project demonstrates basic GUI programming and helps beginners understand Tkinter components such as labels, entry fields, buttons, and text boxes.

Objective

The primary objective of this project is to:

- Provide an **interactive interface** for users to place orders.
- **Calculate the total amount** based on ordered quantities.
- **Generate a receipt** summarizing the order details.

Features of the System

1. **User Interface (UI):**
    - Displays the menu of the café (Coffee, Tea, Sandwich).
    - Accepts user input for the quantities of each item.
    - Provides buttons to calculate the total and generate a receipt.
2. **Receipt Generation:**
    - Displays the item-wise cost and total amount.
    - Automatically clears old receipts for new orders.
3. **Dynamic Calculation:**
    - Calculates the total price based on user inputs.
    - Updates the total in real-time when the order is placed.

Technologies Used

- **Python**: The main programming language.
- **Tkinter**: A built-in Python library used for creating GUIs.

System Requirements

- **Software**:
  - Python 3.x
  - Tkinter (comes pre-installed with Python)
- **Hardware**:
  - Any computer with basic specifications to run Python code.

Code Explanation

- **Importing Tkinter Library**: The project begins by importing the Tkinter library, which is essential for creating the graphical user interface (GUI) of the application.

- **Function to Calculate Total and Generate Receipt**: This function is the core of the application. It performs several tasks:

  - **Calculating Item Costs**: It retrieves the quantities of coffee, tea, and sandwiches entered by the user and calculates the total cost based on predefined prices for each item.
  - **Updating the Total Amount**: After calculating the total cost, it updates a label in the GUI to display the total amount the user needs to pay.
  - **Generating a Receipt**: It formats a receipt string that includes the quantity and cost of each item ordered, as well as the total amount. This receipt string is then displayed in a text area within the application.

- **Creating the Main Window**: The application creates a main window where all elements will be displayed. This window is titled "Café Management" and has a specified size 500 pixels wide and 600 pixels tall. The background color of the window is set to a light cream shade (`#f2e3c6`), enhancing the visual appeal.

- **Variables to Store Quantities**: Three variables are defined to store the quantities of coffee, tea, and sandwiches. These variables are linked to the entry fields in the GUI, allowing the program to access the values entered by the user.

- **Adding Widgets**: Several interface elements (widgets) are added to the window:

  - **Labels**: These display static text, such as the café menu title and the prices of items.
  - **Entry Fields**: These allow users to input the quantities of each menu item.
  - **Button**: A button is provided that, when clicked, triggers the calculation of the total and generates the receipt.

- **Displaying Total and Receipt**: A label is created to show the total cost calculated from the quantities entered. Additionally, a text box is included to display the generated receipt, which gives a clear breakdown of the order and the total amount due.

- **Running the Application**: The application runs in a loop that keeps the window open, allowing users to interact with the interface. It will continue running until the user chooses to close it.

Code

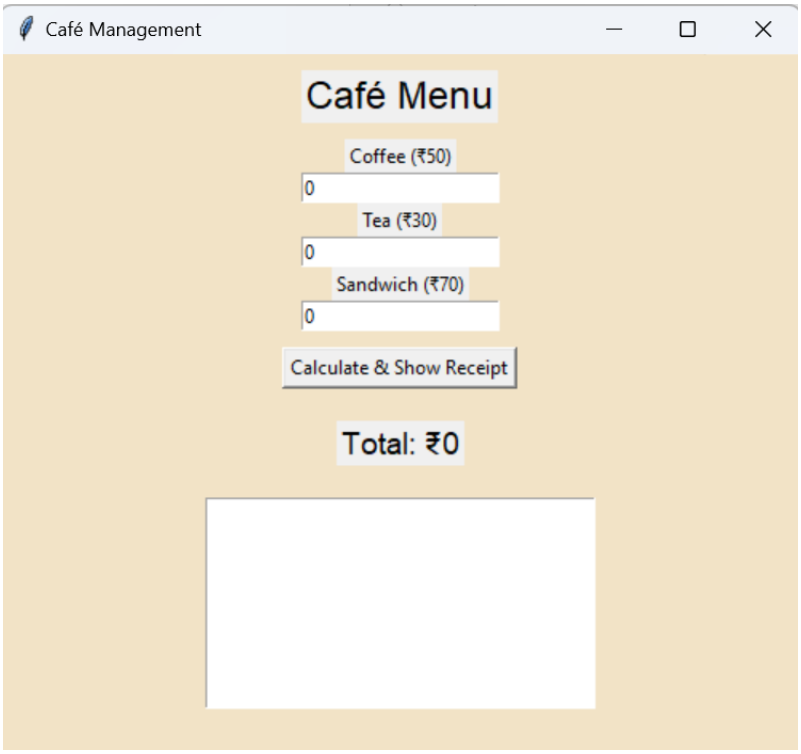
```
from tkinter import *

# Function to calculate total and generate receipt
def calculate_and_show():
    coffee = coffee_qty.get() * 50
    tea = tea_qty.get() * 30
    sandwich = sandwich_qty.get() * 70
    total = coffee + tea + sandwich

    # Update total label
    total_label.config(text=f"Total: ₹{total}")

    # Generate receipt
    receipt = f"Coffee x {coffee_qty.get()} = ₹{coffee}\n" \
              f"Tea x {tea_qty.get()} = ₹{tea}\n" \
              f"Sandwich x {sandwich_qty.get()} = ₹{sandwich}\n" \
              f"----------\nTotal: ₹{total}"
    receipt_text.delete(1.0, END)  # Clear previous receipt
    receipt_text.insert(END, receipt)

# Create the main window
root = Tk()
root.title("Café Management")
root.geometry("500x600")
root.config(bg="#f2e3c6")  # Set background color
```

```
# Variables for item quantities
coffee_qty = IntVar()
tea_qty = IntVar()
sandwich_qty = IntVar()

# Add widgets (labels, entries, and buttons)
Label(root, text="Café Menu", font=("Arial", 18)).pack(pady=10)

Label(root, text="Coffee (₹50)").pack()
Entry(root, textvariable=coffee_qty).pack()

Label(root, text="Tea (₹30)").pack()
Entry(root, textvariable=tea_qty).pack()

Label(root, text="Sandwich (₹70)").pack()
Entry(root, textvariable=sandwich_qty).pack()

Button(root, text="Calculate & Show Receipt", command=calculate_and_show).pack(pady=10)

# Display total and receipt
total_label = Label(root, text="Total: ₹0", font=("Arial", 14))
total_label.pack(pady=10)

receipt_text = Text(root, height=8, width=30)
receipt_text.pack(pady=10)

# Run the app
root.mainloop()
```

User Flow

1. The user opens the **Café Management System**.
2. They **enter the quantities** of coffee, tea, or sandwiches in the input fields.
3. They click the **"Calculate & Show Receipt"** button.
4. The system **calculates the total** and **displays the receipt**.
5. The user can repeat the process by entering new quantities.

Screenshots of the Application



Conclusion

The **Café Management System** built using Tkinter is a simple yet effective project for beginners learning Python GUI development. It demonstrates how to use widgets such as labels, entry fields, and buttons to create an interactive application. With additional features, this project can be expanded into a full-fledged café billing system.

Learning Outcomes

- Gained hands-on experience in designing graphical user interfaces using Tkinter, enhancing skills in creating user-friendly applications.
- Acquired skills in formatting and displaying data in a clear and organized manner, particularly in generating receipts that provide a summary of transactions.
- Enhanced overall Python programming skills, including the use of functions, data types, and control flow, while applying these concepts in a practical context.
- Developed project management skills by planning the project structure, breaking down tasks, and integrating various components into a cohesive application.