MINI PROJECT

TITLE :- “ Simple calender using tkinter”

**1} INTRODUCTION:-**

**1.1 Problem Statement:**

**Managing and checking calendar dates manually can be time-consuming and error-prone. A simple, user-friendly GUI calendar application is needed to assist users in viewing monthly calendars quickly within a desktop environment.**

**1.2 Objectives:**

* **To create a graphical calendar using Python’s Tkinter module.**
* **To allow users to select and display any month and year.**
* **To practice GUI design, event handling, and Python standard library usage.**

**1.3 Scope of the Project:**

* **Focuses on generating and displaying calendar data for any given month and year.**
* **Does not involve scheduling, reminders, or external data syncing.**
* **Usable on any desktop supporting Python and Tkinter.**

**2} TECHNOLOGY STACK USED:-**

**2.1 Programming Languages:**

* **Python 3**

**2.2 Libraries/Frameworks:**

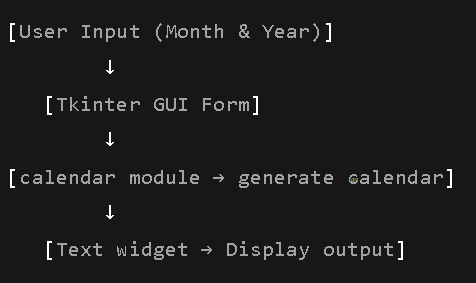
* **Tkinter – for GUI components**
* **calendar – to generate month views**
* **datetime – (optional) for default date handling**

**2.3 Tools and Platforms:**

* **Python IDLE / VS Code**
* **Windows or Linux OS**
* **Tkinter (bundled with Python)**

**3}SYSTEM ARCHITETCURE:-**

**3.1 Architecture Diagram:**

****

**3.2 Module Description:**

* **Main Window: Entry fields for month and year.**
* **Display Button: Triggers calendar generation.**
* **Output Area: Text box where calendar is shown.**

**4}DATASET DESCRIOTION:-**

**4.1 Source of Data:**

* **No external dataset is required. Calendar data is generated using Python’s built-in calendar module.**

**4.2 Data Preprocessing Steps:**

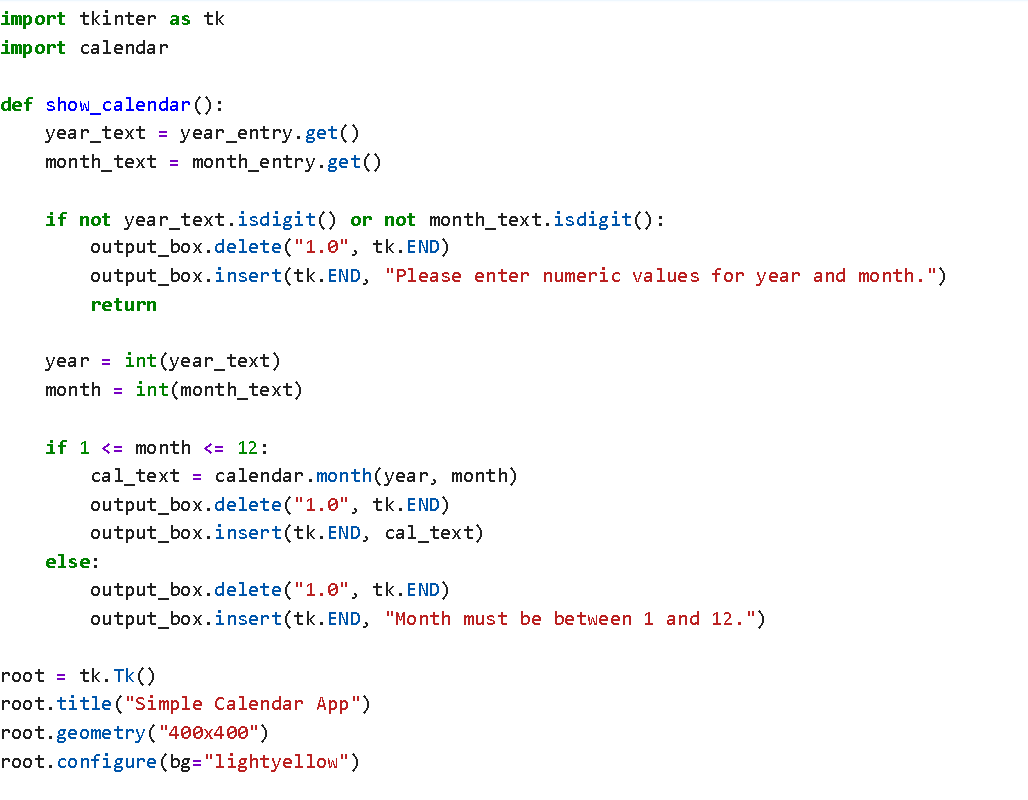
* **Not applicable.**

**5}IMPLEMENTATION:-**

**5.1 Code Flow Description:**

* **GUI is created using Tkinter.**
* **User enters a month and year.**
* **On clicking “Show Calendar,” calendar.month() is called.**
* **Output is shown in a multi-line Text widget.**

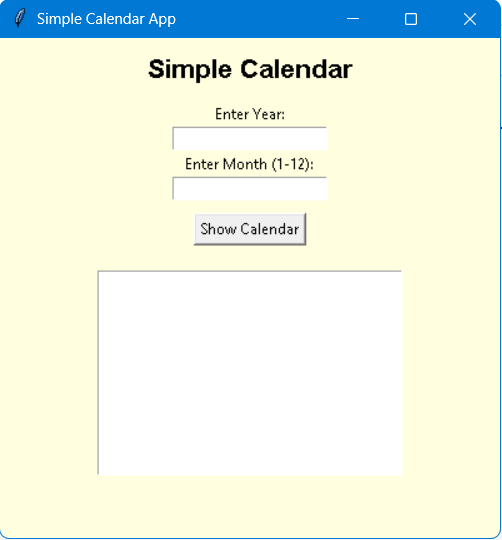
**5.2 Screenshots of Execution:**

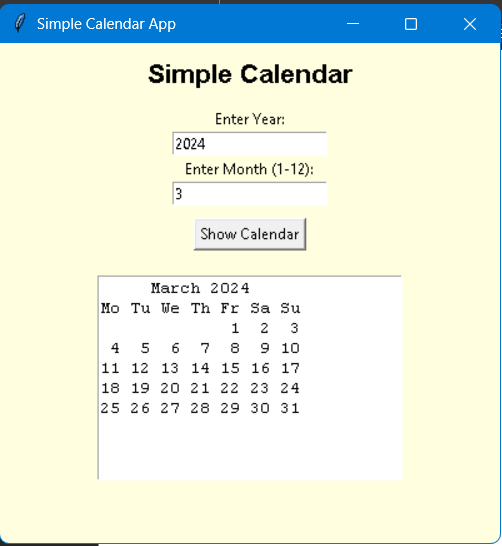
****

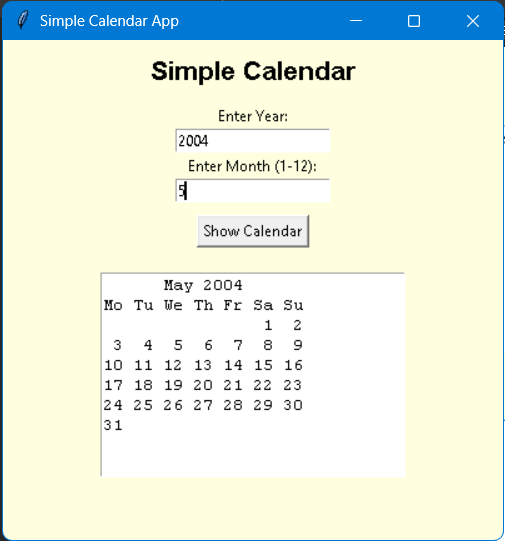
****

**6}RESULT AND ANALYSIS:-**

**6.1 Output Samples:**

****

****

****

**6.2 Performance Evaluation:**

* **Fast and responsive for user input**
* **Lightweight and portable**
* **No significant performance issues observed**

**7}CHALLENGES FACED AND SOLUTIONS:-**

| **Challenge:** | **Solution:** |
| --- | --- |
| **Understanding Tkinter grid system** | **Used documentation and examples** |
| **Handling invalid input (e.g., non-integer)** | **Can be improved with input validation** |
| **Formatting calendar output properly in Text widget** | **Used calendar.month() for pre-formatted strings** |

**8}CONCLUSIONS:-**

**This mini project successfully demonstrates how to build a basic calendar GUI using Python and Tkinter. It allowed the application of fundamental programming concepts and GUI design, resulting in a functional tool for calendar viewing.**

**9} References**

* **Python Tkinter Documentation: https://docs.python.org/3/library/tkinter.html**
* **Python calendar module: https://docs.python.org/3/library/calendar.html**
* **Stack Overflow discussions on GUI layout and event binding**

**10} SUBMITTED BY:-**

* **NAME:- ANSARI MOHAMMAD ARSHAN FIROZ**
* **CLASS:- TY-AIML**
* **ENROLLMENT NUMBER:- 24111590121**
* **COLLEGE:- ANJUMAN ISLAM ABDUL RAZZAQ KALSEKAR POLYTECHNIC PANVEL.**