Laboratory Activity No. 11	
The Grid Manager	
Course Code: CPE103	**Program:** BSCPE
Course Title: Object-Oriented Programming	**Date Performed:** 05/04/2025
Section: 1-A	**Date Submitted:** 05/04/2025
Name: Asugas, Kenneth R.	**Instructor:** Engr. Maria Rizette Sayo

1. Objective(s):

This activity aims to familiarize students on how to implement geometry manager

2. Intended Learning Outcomes (ILOs):

The students should be able to:
2.1 Identify the main components in a GUI Application
2.2 Create a simple GUI Application using Grid manager

3. Discussion:

A Graphical User Interface (GUI) application is a program that the user can interact with through graphics (windows, buttons, text fields, checkboxes, images, icons, etc..) such as the Desktop GUI of Windows OS by using a mouse and keyboard unlike with a Command-line program or Terminal program that support keyboard inputs only.

Geometry managers are tools used to place widgets on the screen. There are three geometry managers available in tkinter—grid, pack, and place. The place manager provides complete control in the positioning of widgets, but is complicated to program

Grids

- A grid is an imaginary rectangle containing horizontal and vertical lines that subdivide it into rectangles called cells. The first row of cells is referred to as row 0, the second row is referred to as row1, and so on. Similarly, the first column of cells is referred to as column 0, the second column of cells is referred to as column 1, and so on. Each cell is identified by its row and column numbers.

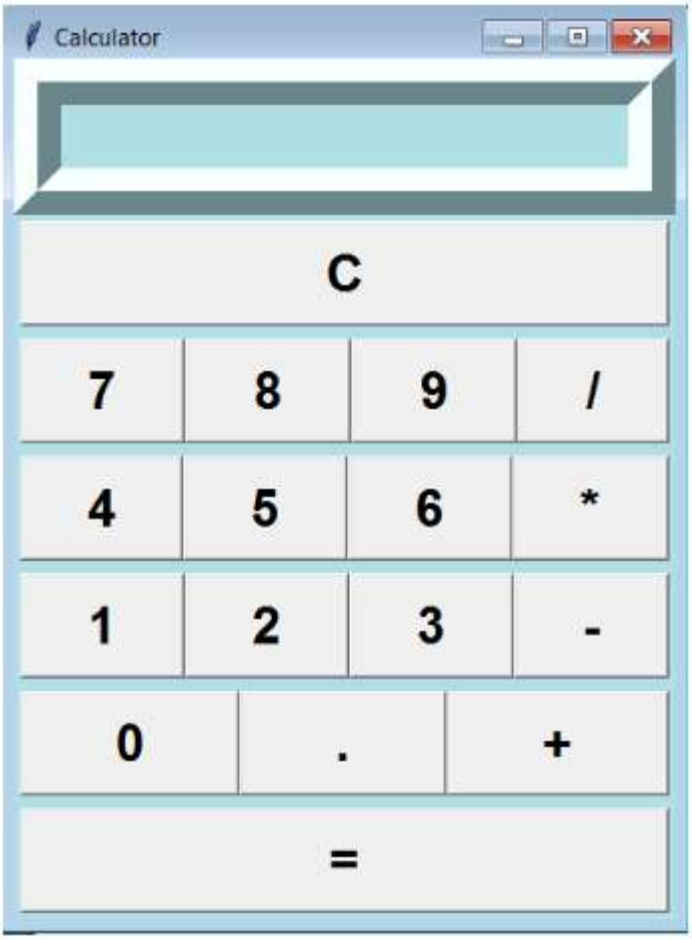
4. Materials and Equipment:

Desktop Computer with Pycharm
Windows Operating System

5. Procedure:

General Instruction:

1. Redesign the interface of the standard calculator using grid ( ) method:



2. Run the program and observe the output when the button is clicked.

For the program please refer to this link: https://github.com/Kenneth-Asugas/CPE-103-OOP-1-A/blob/main/Lab%2011%20Simple%20Calculator.py

6. Supplementary Activity:

1. Make a calculator program that can compute perform the Arithmetic operations as well as exponential operation, sin, cosine math functions as well clearing using the C button and/or clear from a menu bar.
2. Use Geometry manager grid()

3. Use bind () or command parameter in associating event to callback a function.

For the program please refer to this link: https://github.com/Kenneth-Asugas/CPE-103-OOP-1-A/blob/main/Lab%2011%20Advanced%20Calculator.py

Questions

1. How do you configure rows and columns in PyCharm when using Tkinter's grid() manager?

   In Tkinter grid is the manager for row & column configure and define using the row=() and column=() methods that defines how widgets expand and move around the window when being resized: this greatly facilitates creativity in layout - it 's easy to understand and apply - though there are many errors you will certainly encounter and it can be hard to put a button where you want it.

2. Why do widgets sometimes disappear when using grid() in PyCharm, and how can you fix it?

   Widgets might disappear when you use grid() if they are poorly placed or not configured, or if they do not call mainloop(). Make sure the widgets are assigned correctly, that the parent widgets are visible, and that mainloop("address") is called.

3. How can message boxes be used to provide a better User Experience or how can message boxes be used to make a GUI Application more user-friendly? How can you align widgets across multiple frames using grid() in PyCharm?

   Message boxes enhance the user experience by giving instant notification/alert/confirmation, letting users get feedback/warning from applications to enhance their interaction as well. The grid() manager also makes sure all widgets are aligned properly across multiple frames (using parameters like sticky & rowspan) to help with structured and organized layouts which create responsiveness and effective interaction among users.

7. Conclusion:

A GUI application requires a systematic way of arrange widgets. In Tkinter, grid() is one of the organized methods. The proper rows and columns configuration of the widgets makes them flexible and responsive. You should adjust the widget visibility to prevent it from popping up from unintended area. The use of message boxes also helps in building more intuitive user interface because user gets notifies and feedback from messages box when needed. By using grid(and other related functions), students develop well structured, user friendly interfaces which helps in overall output and user experience.

8. Assessment Rubric: