# Calculator

**Explanation of the Code:**

This is a GUI-based calculator application built using Python's **Tkinter** library. Here's a detailed breakdown of the code:

**Global Variables and Initialization**

1. **calculation**:
   * A string variable that holds the user's current input for the calculation.
2. **root**:
   * The main application window created using Tk.Tk().
   * The title method sets the window title as "Calculator."
   * The geometry method sets the size of the window to 300x375 pixels.

**Functions**

1. **add\_to\_calculation(Symbol)**:
   * Appends a symbol (number or operator) to the calculation string.
   * Updates the calculator's display (text\_result) with the current input.
2. **evaluate\_calculation()**:
   * Uses Python's eval() function to evaluate the mathematical expression stored in calculation.
   * If successful, displays the result and clears the calculation string.
   * If an error occurs (e.g., invalid syntax), clears the field and displays "Error."
3. **clear\_field()**:
   * Resets the calculation string and clears the display.
4. **backspace()**:
   * Removes the last character from the calculation string.
   * Updates the display to reflect the modified input.

**User Interface (GUI Elements)**

1. **text\_result**:
   * A Text widget to display user input and results.
   * Configured with a font size of 24 for readability and spans multiple columns using grid(columnspan=5).
2. **Buttons**:
   * Created using Tk.Button() with properties:
     + **text**: The label of the button (e.g., numbers, operators).
     + **command**: The action performed when the button is clicked (uses lambda for parameterized calls).
     + **width and font**: Define button size and appearance.
   * Positioned using the grid(row, column) method for proper layout.
3. **Specific Buttons**:
   * **Numbers (0-9):** Append the corresponding digit to calculation.
   * \**Operators (+, -, , /, %):* Add the operator to calculation.
   * **= Button:** Calls evaluate\_calculation() to compute the result.
   * **C Button:** Clears all input using clear\_field().
   * **← Button (Backspace):** Removes the last character via backspace().
   * **. Button:** Adds a decimal point to the input.

**Layout**

* The calculator layout follows a grid system:
  + Buttons are arranged row by row (e.g., numbers, operators, and functions).
  + Logical grouping ensures intuitive user interaction.

**Main Event Loop**

* **root.mainloop()**:
  + Starts the Tkinter event loop, allowing the GUI to remain responsive and operational.

This design creates a functional calculator capable of basic arithmetic operations with a simple graphical interface.