**Assignment 8 (working in pairs)**

In this assignment, you design and code a simple calculator for addition, subtraction, multiplication, or division. Multiple operations in a single expression such as 2 + 3\*4 is not considered, neither are negative numbers. However, the calculator does allow operations with multi-digit numbers such as 21 + 34. The calculator frame has three components: a numbers panel for 0, 1, 2, …, 9, an operator panel for +, -. \*, /, and =, and a panel for displaying the current calculation and logging the already performed calculations in the same session. You decide how to design these panels, and the layout. For simplicity, calculations should only work for integers. As an example, if one pushes number 2, and 2 again, then +, and then 3, and 3 again, the display panel would show the expression in progression as the keys were pressed, ending with 22+33. Once = is pushed, the display would show just 55 (just like how a simple calculator on your phone works).

**Requirements**

1. To implement the calculation, we typically use “if-else” structures to tell whether it is a +, -, \*, or / when an operation button is pushed. However, for this assignment, you avoid using “if-else” structures. Instead, you use polymorphism coupled with a factory method, thus allowing easier extension of the program when more operations would be added later.
2. Use the Builder pattern to construct the panels (i.e., product components) and the frame (i.e., the product).
3. Also, use Singleton pattern to ensure the uniqueness of the frame.