I created a calculator program with three basic functions: add, subtract and multiply. The bad_calculator.py code violates the KISS, DRY Code, and Clean Code principles. Here is a breakdown of these violations:

- **KISS**: The print statements inside each function are unnecessary, adding complexity.
- **DRY Code**: Each function has redundant print statements that could be handled separately.
- **Clean Code**: The function names, add_func and sub_func, and variable name, c, are not meaningful enough. There are also hardcoded print statements in each function.

The good_calculator.py code fixes these issues in the following ways:

- **KISS**: The code is simple and modular as I got rid of unnecessary code.
- **DRY Code**: The print and input logic are consolidated outside the individual functions.
- **Clean Code**: Function and variable names are intuitive. Input, logic, and output are also organized separately. We no longer have hardcode.