Assignment-10.2

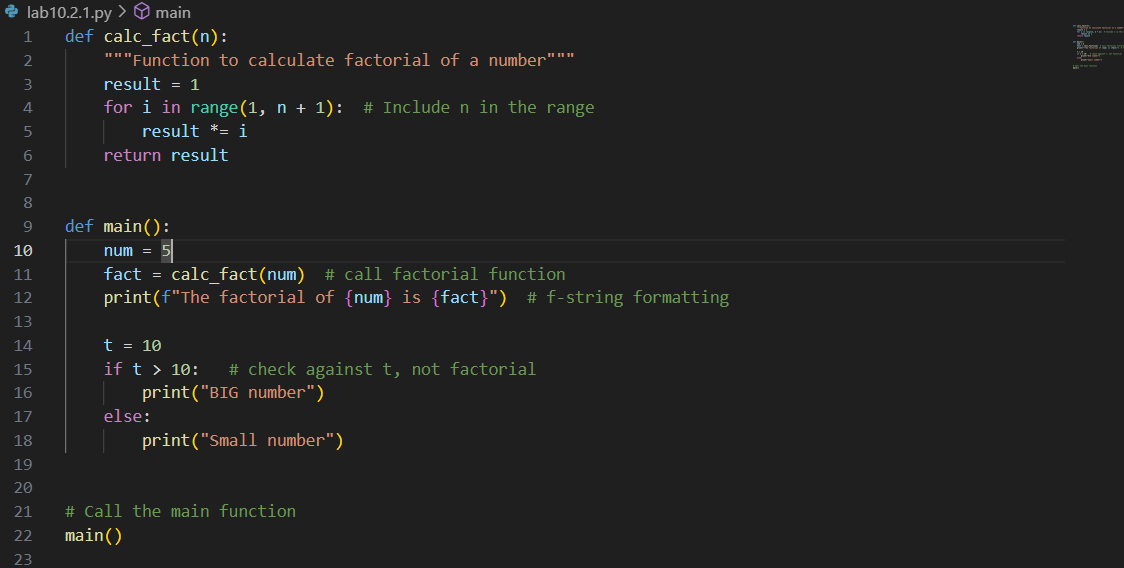
Lab 10: Code Review and Quality: Using AI to improve code quality and readability

10.2.1:

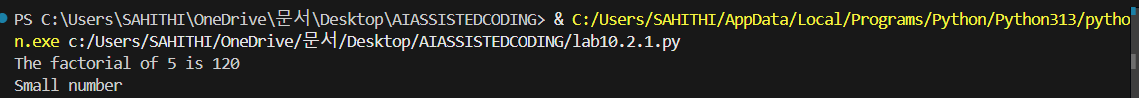
**Issues in the given code:**

1. **Factorial function loop**:
   * The loop for i in range(1, n): only goes up to n-1.
   * To correctly calculate factorial, it should be range(1, n+1).
2. **Unused variable x=0**:
   * This line is unnecessary and can be removed.
3. **Variable naming convention**:
   * In Python, variables are usually lowercase (fact instead of FACT).
   * Using all caps is reserved for constants.
4. **Logic in main function**:
   * The variable t = 10 is declared but not used.
   * The if condition is checking FACT > 10 instead of t > 10.
5. **Print formatting**:
   * Current print statement is a bit messy. Better to use f-strings

CODE:



OUTPUT:



**Key Fixes:**

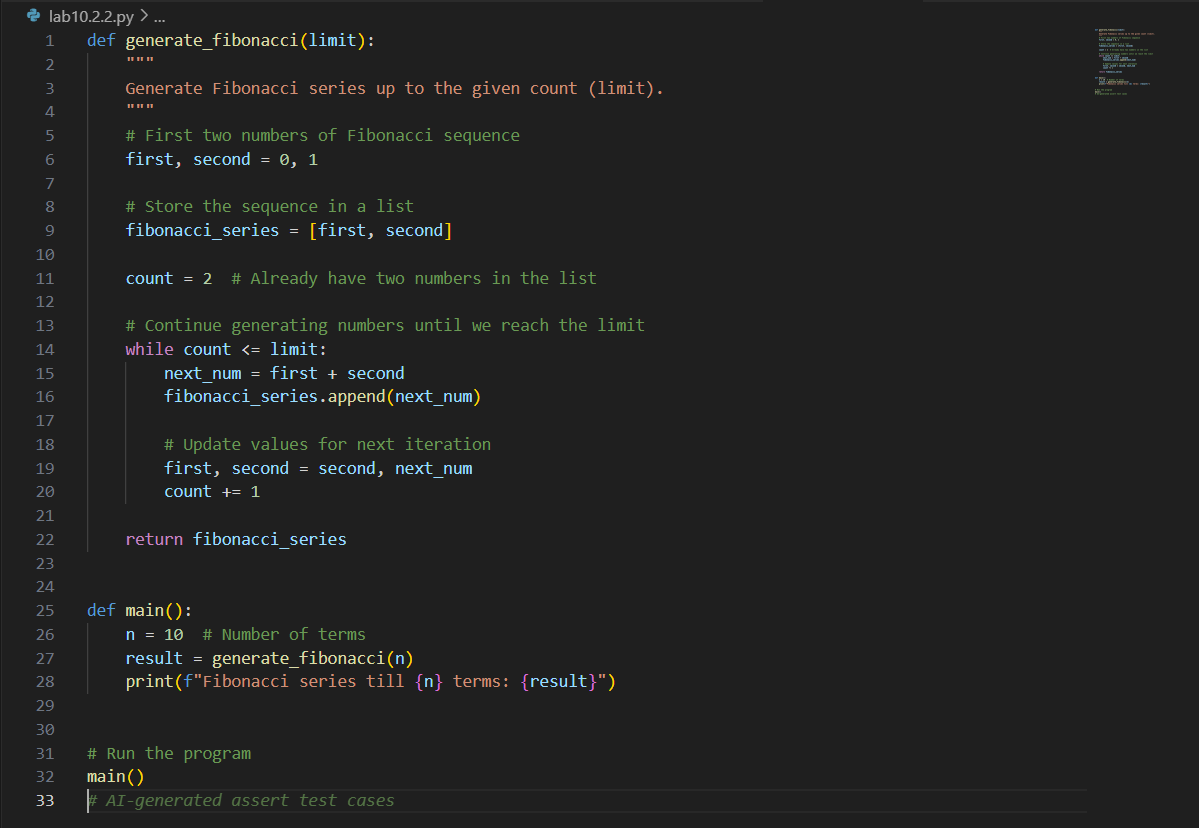
* Fixed loop range to calculate factorial correctly.
* Removed unused variable x.
* Changed variable names (FACT → fact) to follow Python conventions.
* Corrected conditional check (if t > 10: instead of if FACT > 10:).
* Improved print formatting using f-strings.

10.2.2:

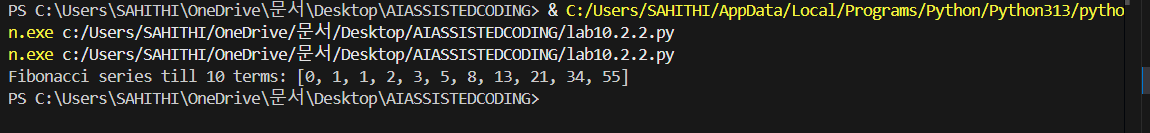
**Issues in the given code:**

1. Variable names (a, b, c, d, Zz, xX, NN) are not meaningful.
2. Inconsistent capitalization (xX, NN) → violates PEP8.
3. No comments → hard to understand the logic.
4. Formatting issues (extra spaces, unclear print).
5. Function names should be descriptive (f1 → generate\_fibonacci).

CODE:



OUTPUT:



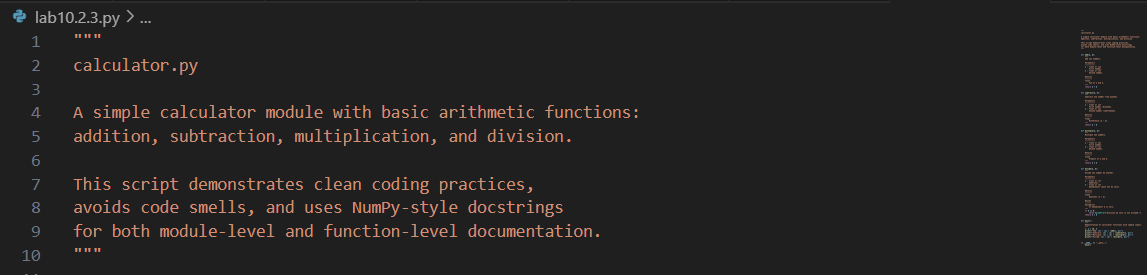
**Improvements:**

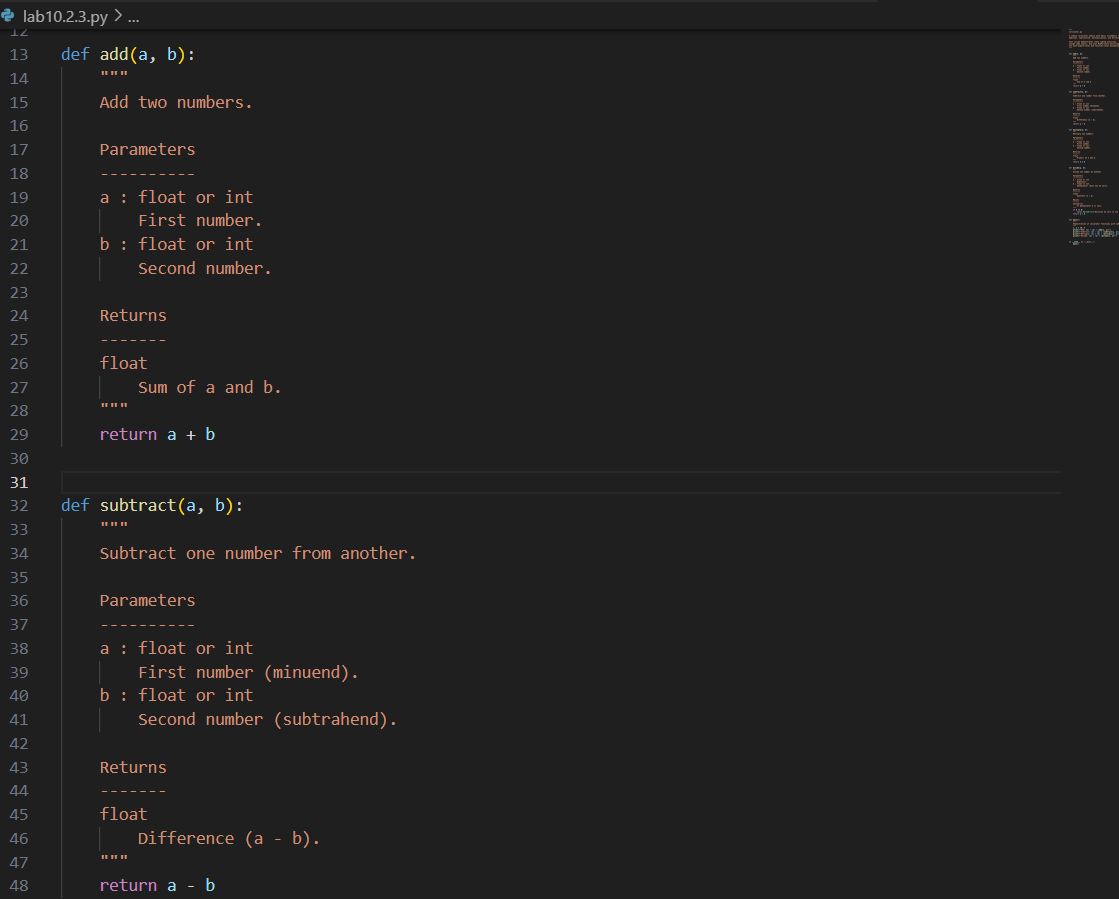
* Renamed functions and variables for clarity:
  + f1 → generate\_fibonacci
  + a, b, d → first, second, next\_num
  + Zz → fibonacci\_series
* Added **docstring** for the function.
* Added inline **comments** explaining each step.
* Followed **PEP8** naming conventions.
* Improved print formatting with f-strings.

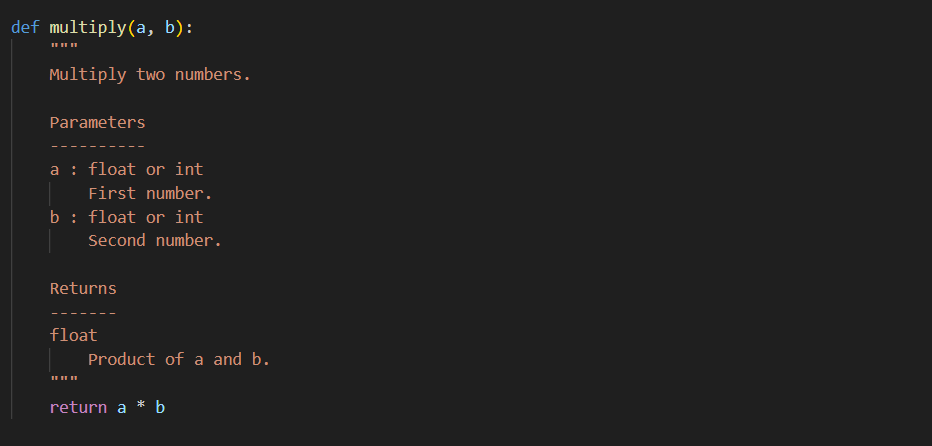
10.2.3

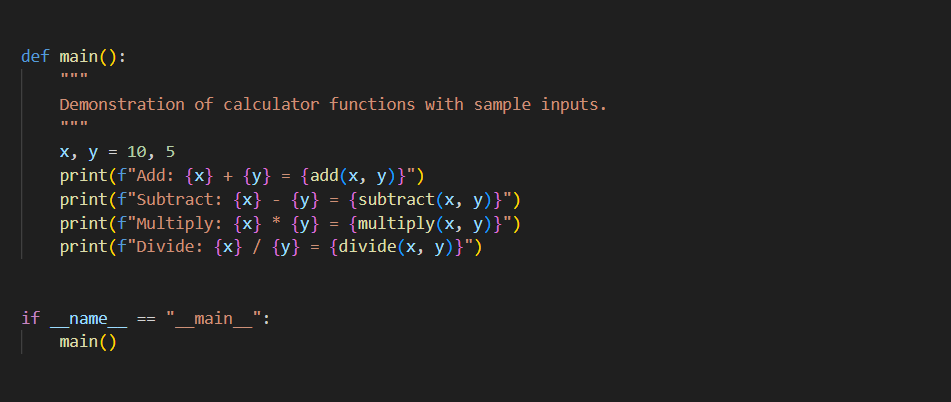
This task is about writing a **multi-function calculator program** with **NumPy-style docstrings**. Let’s build a clean script with structured documentation, avoiding **code smells** like poor naming, duplicate code, and unused variables.

CODE:

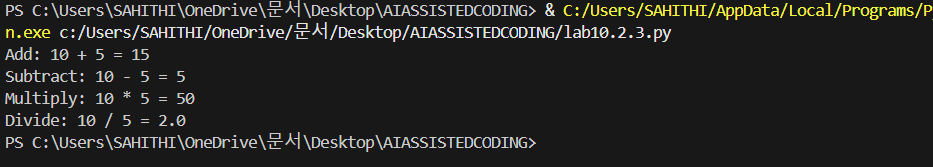








OUTPUT:



**Features of This Version:**

* **4 functions**: add, subtract, multiply, divide.
* **NumPy-style docstrings** at:
  + Module level (top of file).
  + Each function (with Parameters, Returns, and Raises).
* **No code smells**:
  + No duplicate code.
  + No unused variables.
  + No magic numbers (clear variable names x, y).
  + No deep nesting.
* **Error handling** for division by zero