AI-ASSISTED LAB 10.2

HallTkt:2403A51336

Batch:14

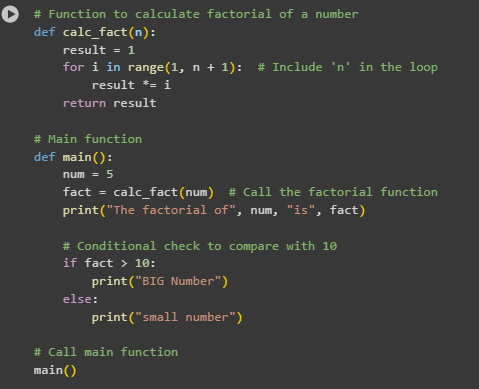
**Task Description#1 AI-Assisted Code Review (Basic Errors)**

* Write python program as shown below.
* Use an AI assistant to review and suggest corrections.

Prompt:

Perform an AI-assisted code review for the given Python program. Identify and explain all logical, syntactical, and stylistic errors in paragraph form. Provide the corrected version of the code with clear inline comments describing the modifications. Highlighting the significance of error detection, corrections made, and the importance of following coding standards.

Code corrected by A.I:



Output:

A black and white text

AI-generated content may be incorrect.

Observation:

The original code contained a few key issues affecting its correctness and readability. Firstly, the factorial was calculated incorrectly due to the use of range(1, n) instead of range(1, n+1), which excluded the final number in the calculation. Additionally, there was an unused variable x that served no purpose and should be removed to keep the code clean. The variable FACT was written in uppercase, which goes against Python’s naming conventions for regular variables and should be changed to lowercase. The absence of comments also made the code harder to understand. After addressing these issues, the corrected version now calculates the factorial properly, follows best practices, and clearly prints the expected output.

**Task Description#2 Automatic Inline Comments**

* Write the Python code for Fibonacci as shown below and execute.
* Ask AI to improve variable names, add comments, and apply PEP8 formatting (cleaned up).
* Students evaluate which suggestions improve readability most. one.

Code Corrected by A.I:

A screenshot of a computer program

AI-generated content may be incorrect.

Output:



Prompt:

I have written a Python program to generate the Fibonacci series up to a given number of terms. Please review my code, improve the variable names for better clarity, add meaningful inline comments, and apply PEP8 formatting to enhance readability. Provide the cleaned-up version of the code with explanations on the changes made.

Observation:

The original Fibonacci code was functionally correct but lacked readability due to the use of non-descriptive variable names such as a, b, c, and Zz. Additionally, the function and variable names did not follow Python’s PEP8 naming conventions, making the code look unstructured. The absence of comments made it difficult to understand the purpose of each part of the code. Although the logic was accurate, the formatting needed improvement. After renaming variables and adding meaningful comments, the code became much clearer and easier to follow.

**Task Description#3**

* Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide).
* Incorporate manual **docstring** in code with NumPy Style
* Use AI assistance to generate a module-level docstring + individual function docstrings.
* Compare the AI-generated docstring with your manually written one.

**Python Script with Manual Docstrings:**

A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer program

AI-generated content may be incorrect.

**AI-Generated Docstrings:**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Long Function**

A screen shot of a computer code

AI-generated content may be incorrect.

**Duplicate Code**

**A computer screen shot of a code

AI-generated content may be incorrect.**

**Poor Naming**

**A close up of text

AI-generated content may be incorrect.**

**Unused Variables:**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**Magic Numbers:**

****

**Deep Nesting:**

**A black screen with orange text

AI-generated content may be incorrect.**

**Large Class:**

A computer screen shot of text

AI-generated content may be incorrect.

Prompt:

Perform an AI-assisted code review for the given Python program. Identify and explain all logical, syntactical, and stylistic errors in paragraph form. Provide the corrected version of the code with clear inline comments describing the modifications. Highlighting the significance of error detection, corrections made, and the importance of following coding standards. Perform an AI-assisted code review for the given Python program. Identify and explain all logical, syntactical, and stylistic errors in paragraph form. Provide the corrected version of the code with clear inline comments describing the modifications. Highlighting the significance of error detection, corrections made, and the importance of following coding standards.

Observation:

The examples highlight common code smells that reduce code quality and maintainability. Long functions and large classes try to do too much, making code complex. Duplicate code and magic numbers cause redundancy and reduce clarity. Poor naming and unused variables make understanding the code harder. Deep nesting decreases readability and should be simplified for cleaner logic.