**AI ASSISTED CODING LAB**

**ASSIGNMENT 3.2**

**ENROLLMENT NO :**2503A51L15

**BATCH NO:** 19

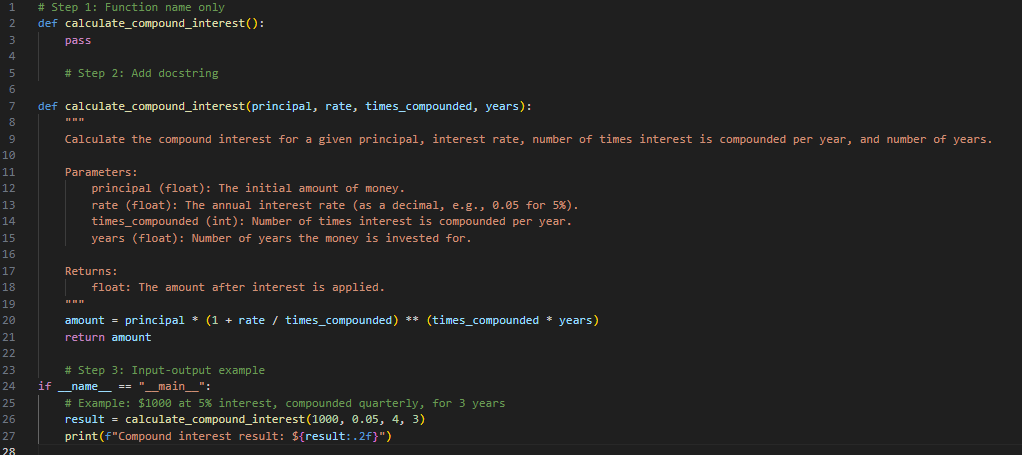
**NAME:** MOHAMMAD KHAJA AFZALUDDIN

**TASK1**

**TASK1 DESCRIPTION:-** Ask AI to write a function to calculate compound interest, starting with only the function name. Then add a docstring, then input-output example

**PROMPT:-** **"**Write a Python function to calculate compound interest. Start by providing only the function name. Next, add a docstring that explains what the function does, its parameters, and its return value. Then, include an input-output example that demonstrates how to use the function and what result it produces." implement with example

**CODE:-**

****

**OUTPUT:-**

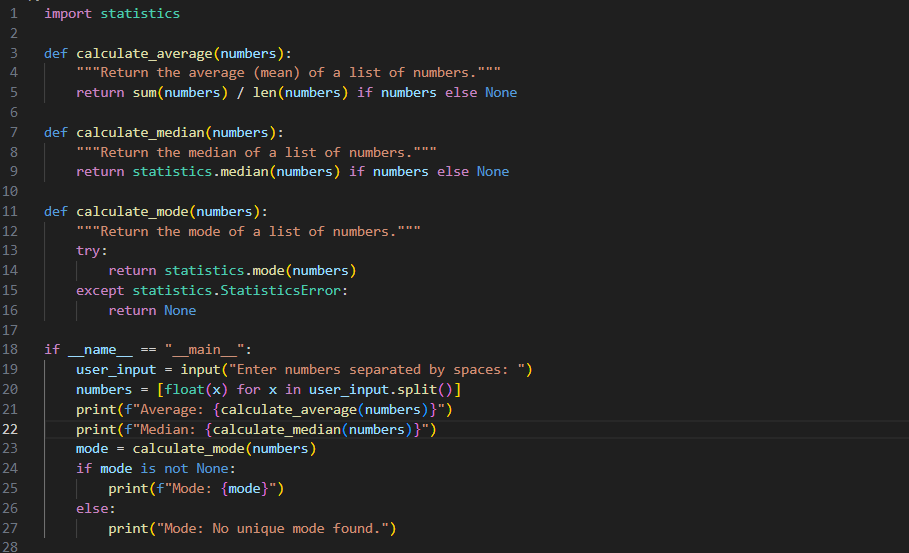
****

**TASK2**

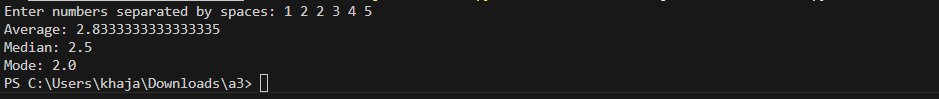
**TASK2 DESCRIPTION:-** Do math stuff, then refine it to: # Write a function to calculate average, median, and mode of a list of numbers

**PROMPT:-** generate a python function to calculate average, median, and mode of a list of numbers that should be enter at runtime

**CODE:-**

****

**OUTPUT:-**

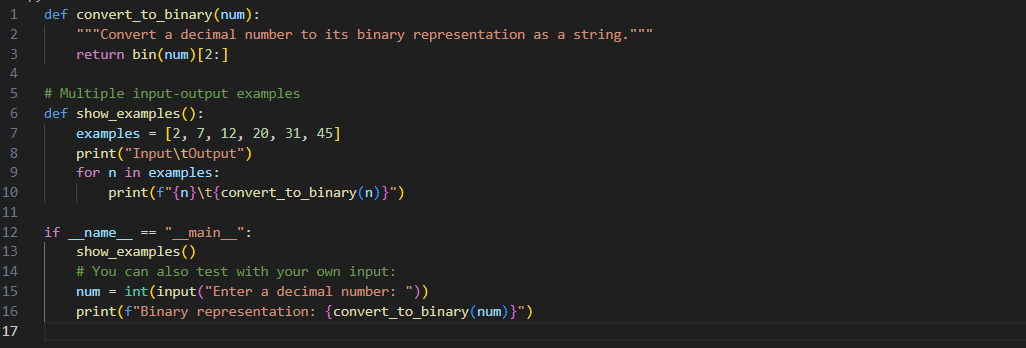
****

**TASK3**

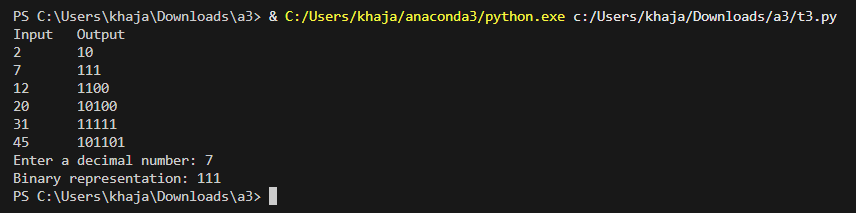
**TASK3 DESCRIPTION:-** Provide multiple examples of input-output to the AI for convert\_to\_binary(num) function. Observe how AI uses few-shot prompting to generalize.

**PROMPT:-** generate a python program that provides multiple input–output examples for a Python function convert\_to\_binary(num) that converts a decimal number into its binary representation

**CODE:-**

****

**OUTPUT:-**

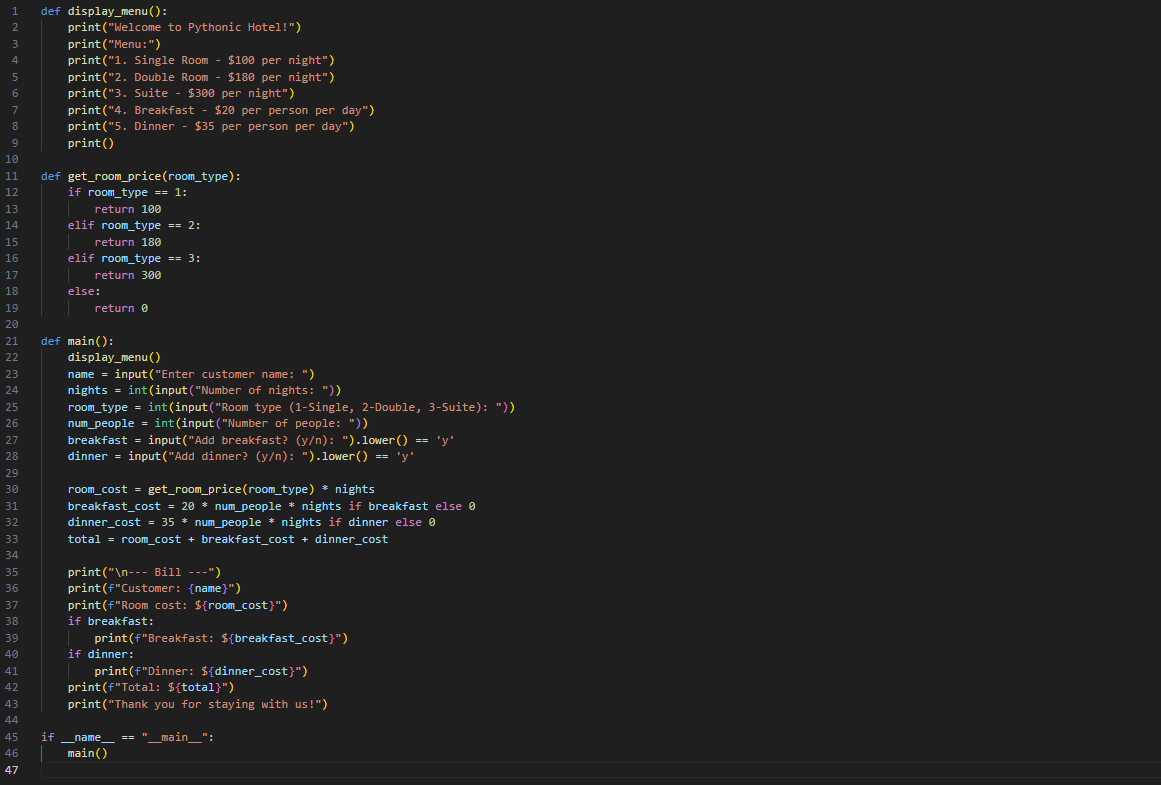
****

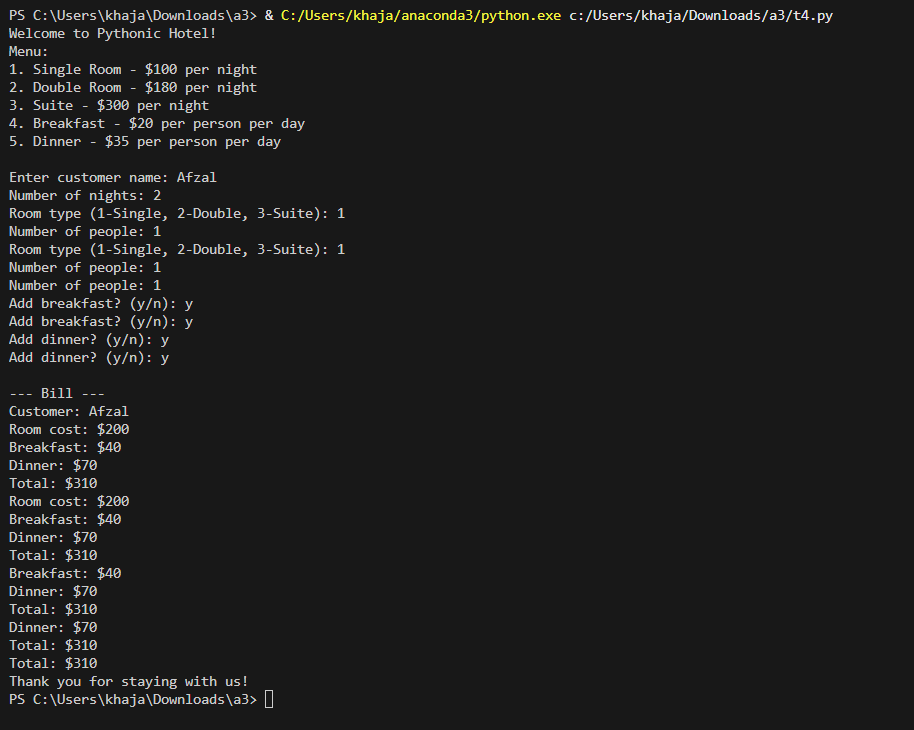
**TASK4**

**TASK4 DESCRIPTION:-**Create an user interface for an hotel to generate bill based on customer requirements

**PROMPT:-** Generate a python program to Create an user interface for an hotel to generate bill based on customer requirements

**CODE:-**

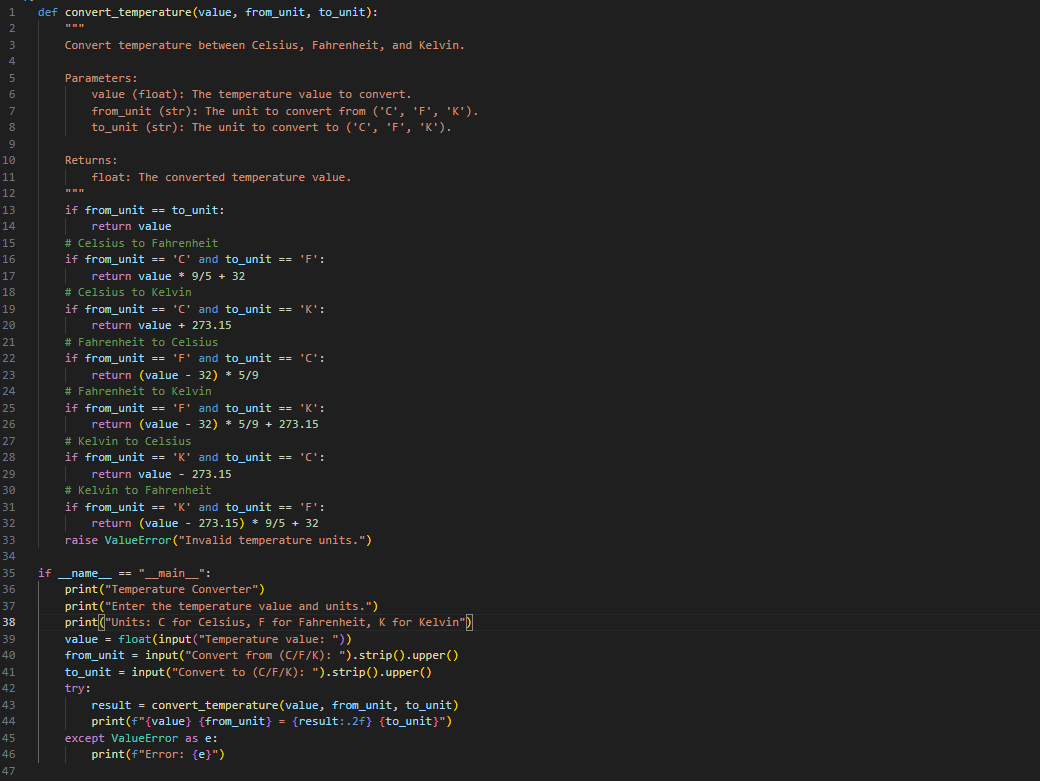
****

****

**TASK5**

**TASK5 DESCRIPTION:-** Analyzing Prompt Specificity: Improving Temperature Conversion Function with Clear Instructions

**PROMPT:-** Generate a python program to Improving Temperature Conversion Function with Clear Instructions

**CODE:-**

**OUTPUT:-**

****

**OBSERVATION:-** From this assignment, I observed the practical role of prompt engineering and AI-assisted coding in generating Python programs. By giving different levels of instructions, the AI was able to produce complete implementations, examples, and even user interfaces.

* In Task 1, starting with only a function name and gradually adding docstrings and examples demonstrated how AI understands step-by-step instructions and builds code systematically.
* In Task 2, I observed how runtime inputs can be used for statistical calculations (average, median, mode), showing AI’s capability to handle mathematical logic on user-provided data.
* In Task 3, by providing multiple input–output examples for the convert\_to\_binary(num) function, I noticed how AI applied few-shot prompting to generalize and generate correct binary conversions for any decimal input.
* In Task 4, the hotel billing program highlighted how AI can extend beyond simple functions to build user-oriented applications, combining logic with interface design.
* In Task 5, refining the temperature conversion function showed how prompt specificity directly affects the accuracy, clarity, and usability of AI-generated code.