ASSIGNMENT -3.1

BATCH-29

ROLL_NO:2303A51600

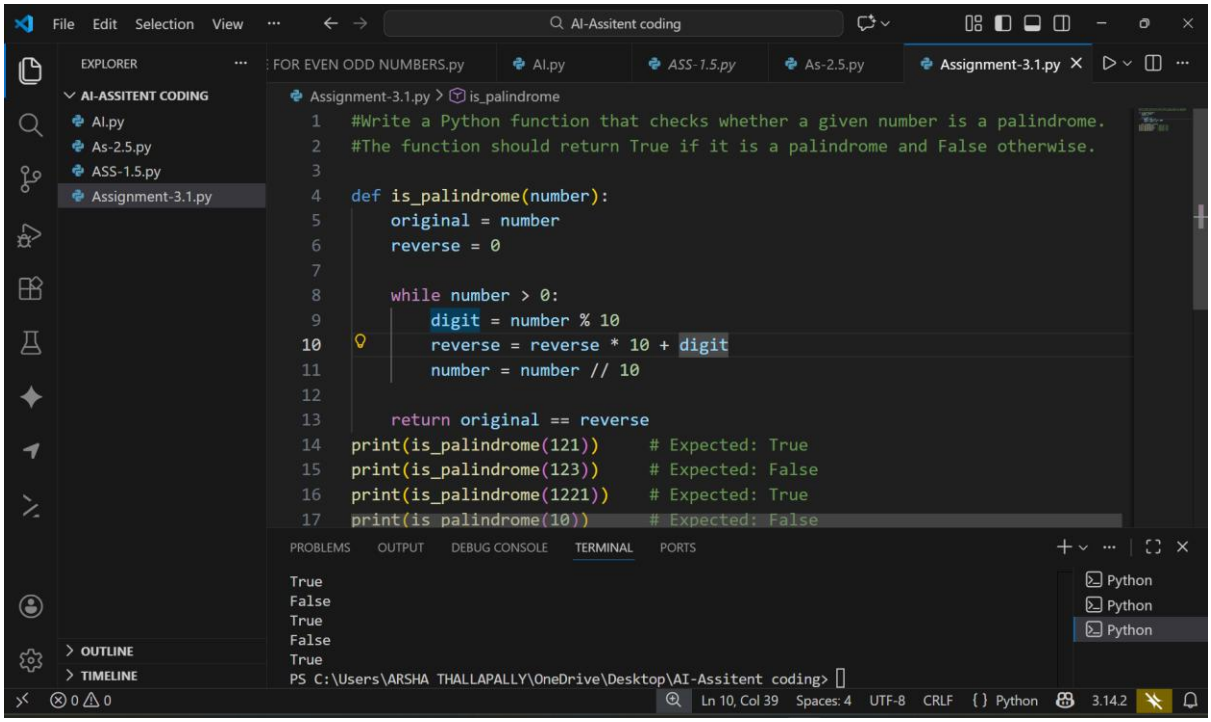
NAME-ARSHA VARDHINI

TASK-1: ZERO-SHOT PROMPTING (PALINDROME NUMBER PROGRAM)

PROMPT: Write a Python function that checks whether a given number is a palindrome.

The function should return True if it is a palindrome and False otherwise.

CODE:



OBSERVATION:

-The model is given only the explanation of the question

-Any example or detailed explanation is not given

-Answer is accurate but not specific with negative and non-integers values

TASK-2: ONE-SHOT PROMPTING (FACTORIAL CALCULATION)

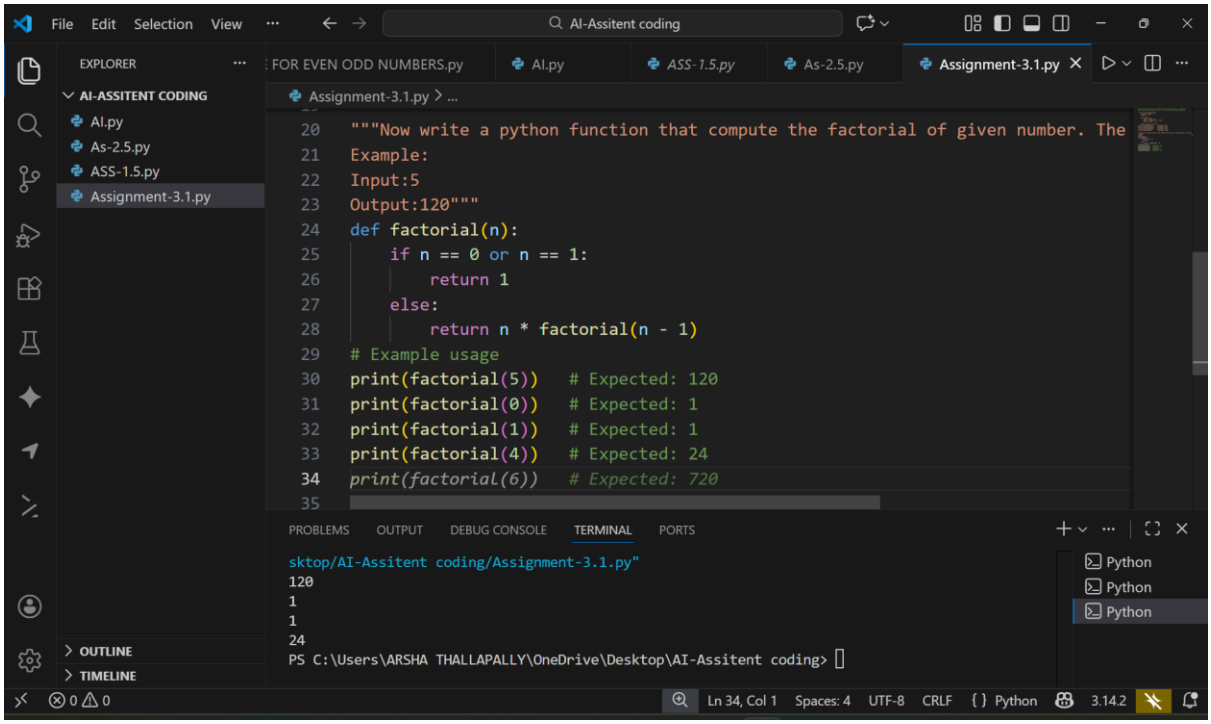
PROMPT: Now write a python function that compute the factorial of given number. The function should return the result.

Example:

Input:5

Output:120

CODE:



OBSERVATION:

Clear understanding of the output

Better choice of logic-stack overflow, recursion complexity

Correct handling of base case

Improve code simplicity

TASK-3: FEW-SHOT PROMPTING (ARMSTRONG NUMBER CHECK)

Prompt: Example 1:

Input: 153

Output: Armstrong Number

Example 2:

Input: 370

Output: Armstrong Number

Example 3:

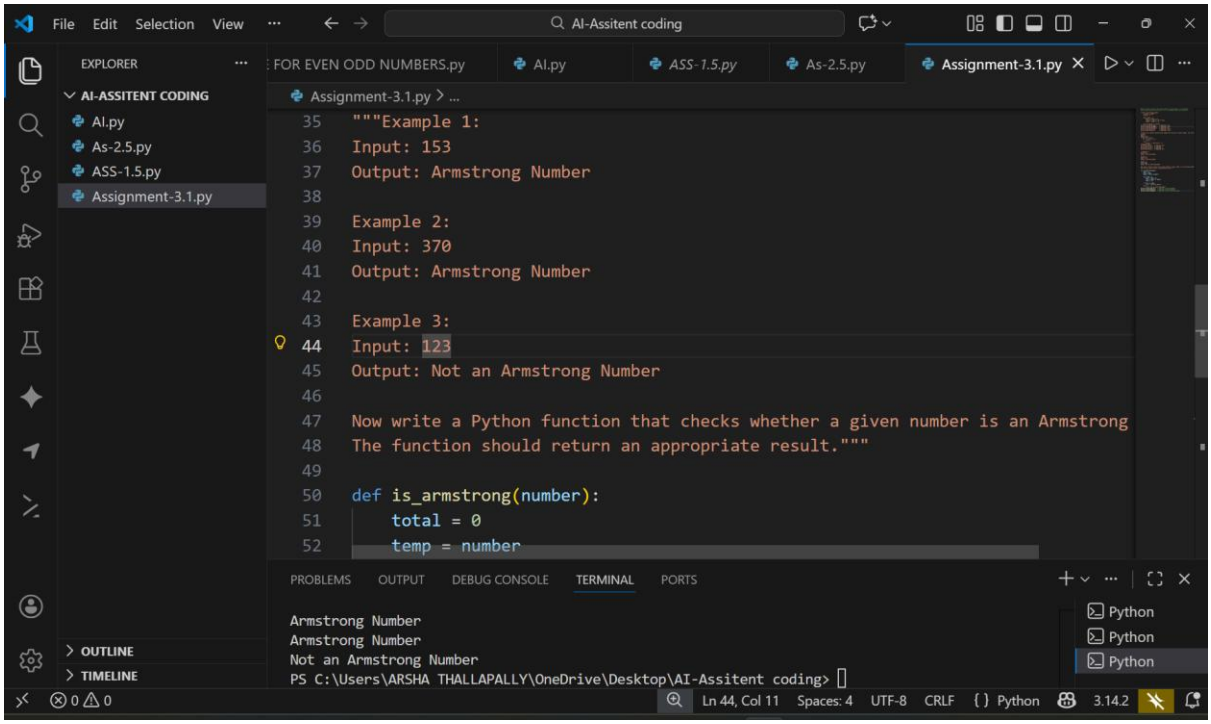
Input: 123

Output: Not an Armstrong Number

Now write a Python function that checks whether a given number is an Armstrong number.

The function should return an appropriate result.

CODE:



OBSERVATION:

Clear output formatting. structured way

Correct logic selection

Easy understanding of code

Exact Appropriate answer

Optimized and customized solution

TASK-4: CONTEXT-MANAGED PROMPTING (OPTIMIZED NUMBER CLASSIFICATION)

PROMPT:

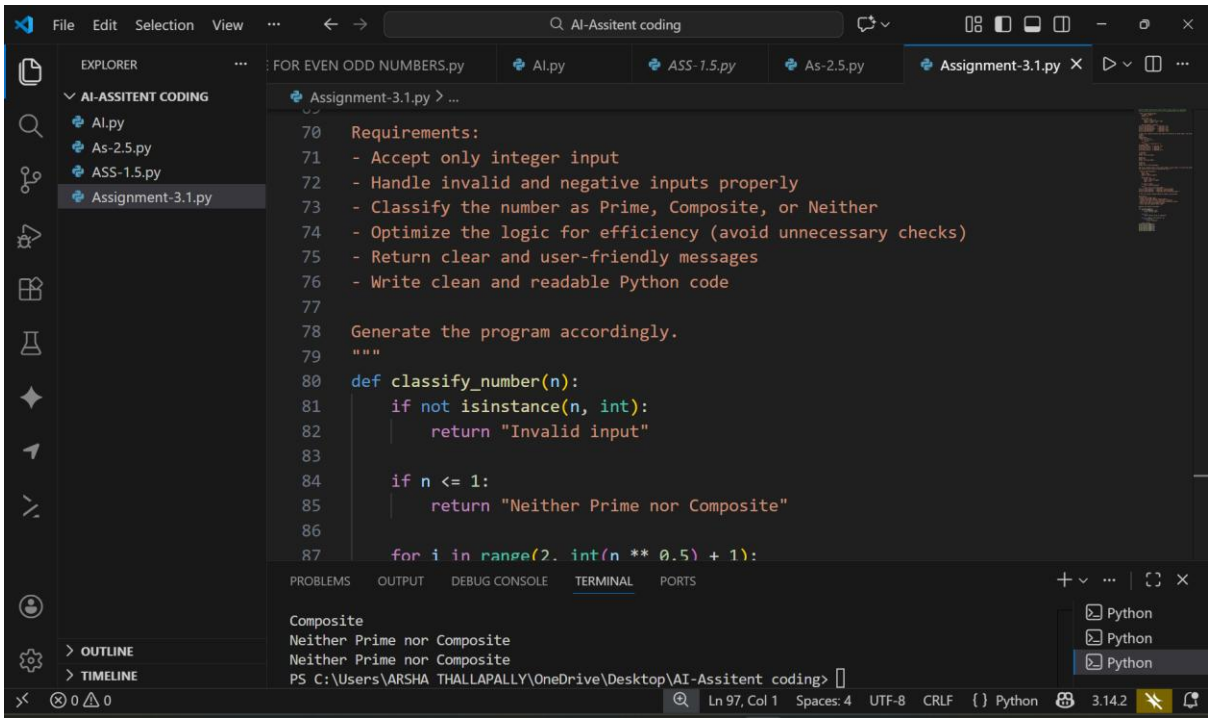
You are writing a Python program for number classification.

Requirements:

- Accept only integer input
- Handle invalid and negative inputs properly
- Classify the number as Prime, Composite, or Neither
- Optimize the logic for efficiency (avoid unnecessary checks)
- Return clear and user-friendly messages
- Write clean and readable Python code

Generate the program accordingly.

CODE:



OBSERVATION:

The role is defined

Constrains are clearly stated

Efficiency and validation of the code

But the inputs should be specified more clearly mentioned

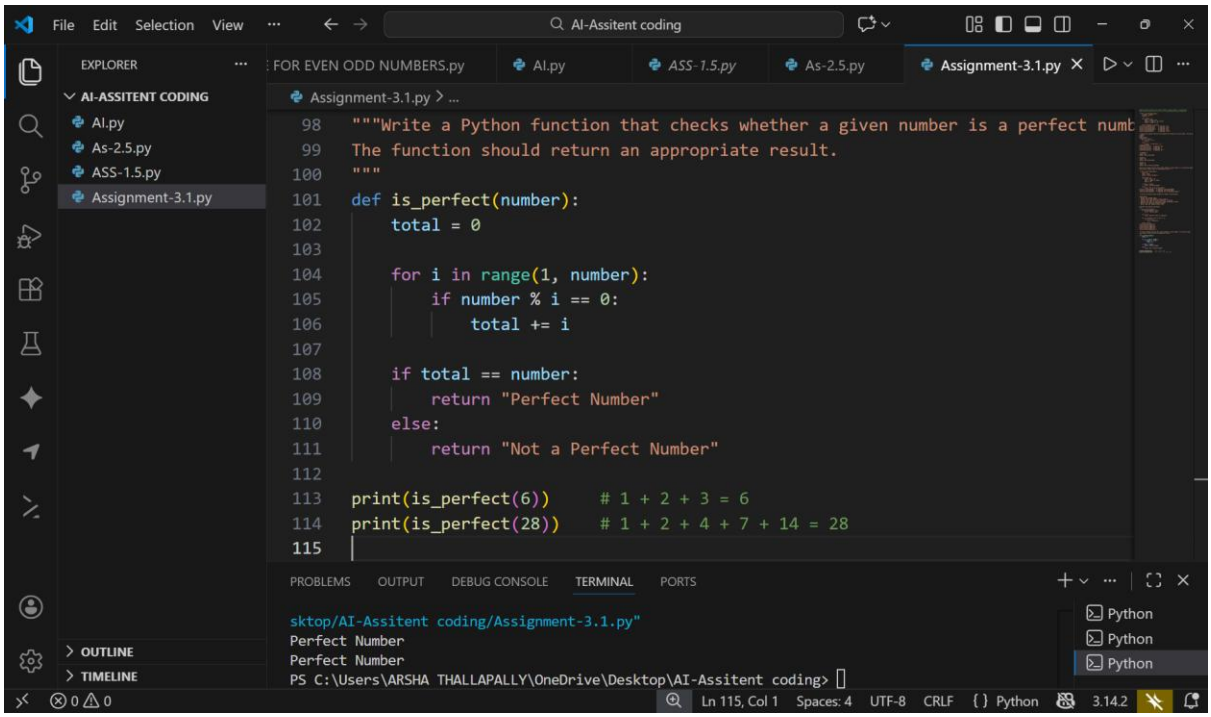
TASK-5: ZERO-SHOT PROMPTING (PERFECT NUMBER CHECK)

VALIDATION)

PROMPT: Write a Python function that checks whether a given number is a perfect number.

The function should return an appropriate result.

CODE:



OBSERVATION:

No input validation – if negative or float any..

Inefficient for large input

Did not specify input constraints

No edge case handing seen

TASK-6: FEW-SHOT PROMPTING (EVEN OR ODD CLASSIFICATION WITH VALIDATION)

PROMPT:

Example 1:

Input: 8

Output: Even

Example 2:

Input: 15

Output: Odd

Example 3:

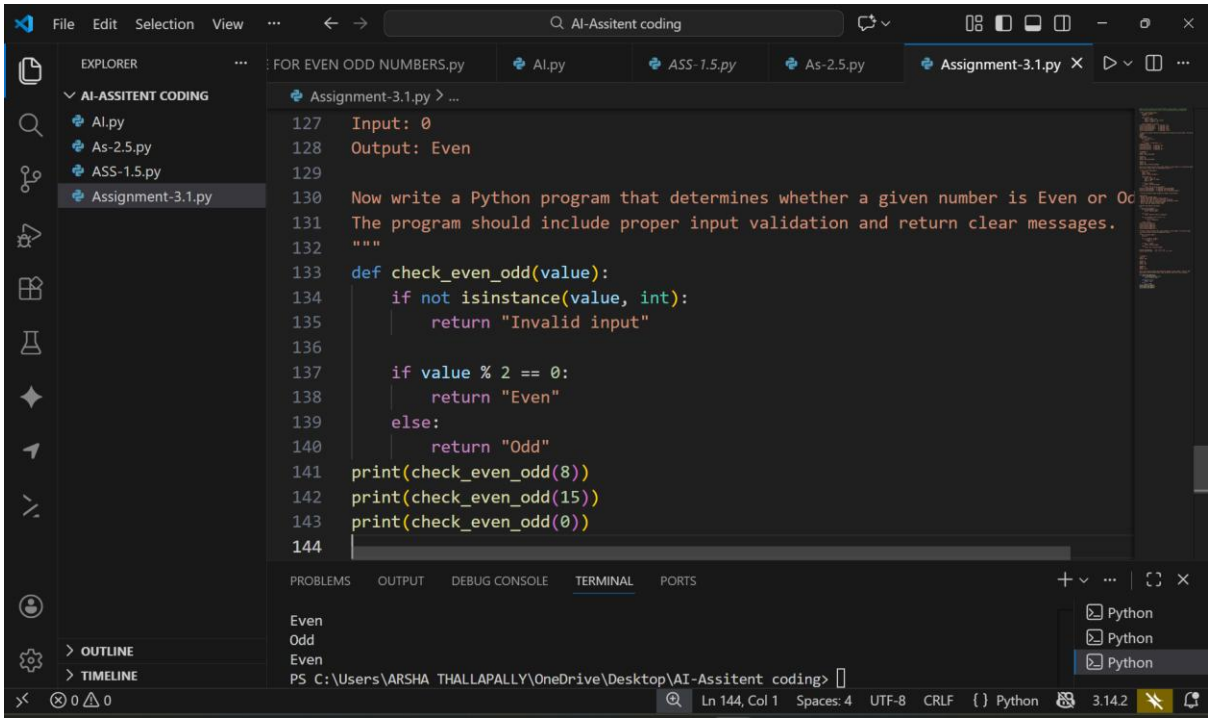
Input: 0

Output: Even

Now write a Python program that determines whether a given number is Even or Odd.

The program should include proper input validation and return clear messages.

CODE:



OBSERVATION:

Negative integer are handled correctly

Program safely rejected non integer inputs

Improve input handling

Clear and consistent output