ASSIGNMENT – 3.1

B.MANUDEEP REDDY
2303A52380
B-43

Question 1:
Zero-Shot Prompting (Palindrome Number Program)

Write a zero-shot prompt (without providing any examples) to generate

a Python function that checks whether a given number is a palindrome.

Task:

• Record the AI-generated code.

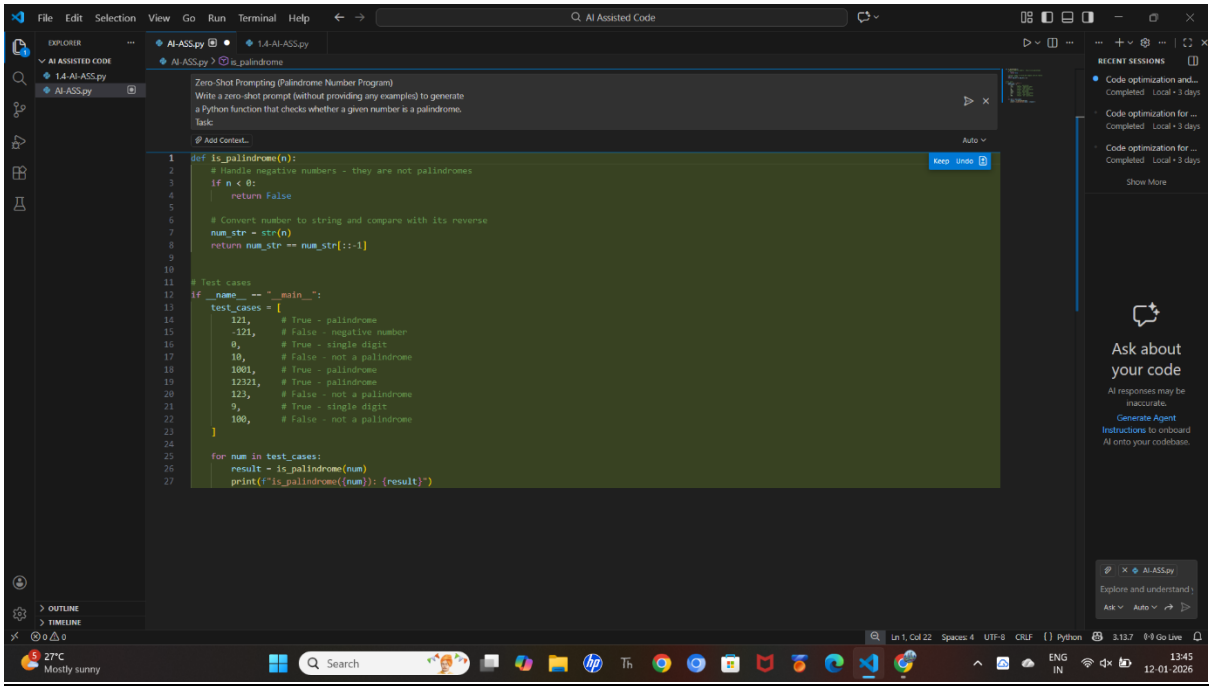
• Test the code with multiple inputs.

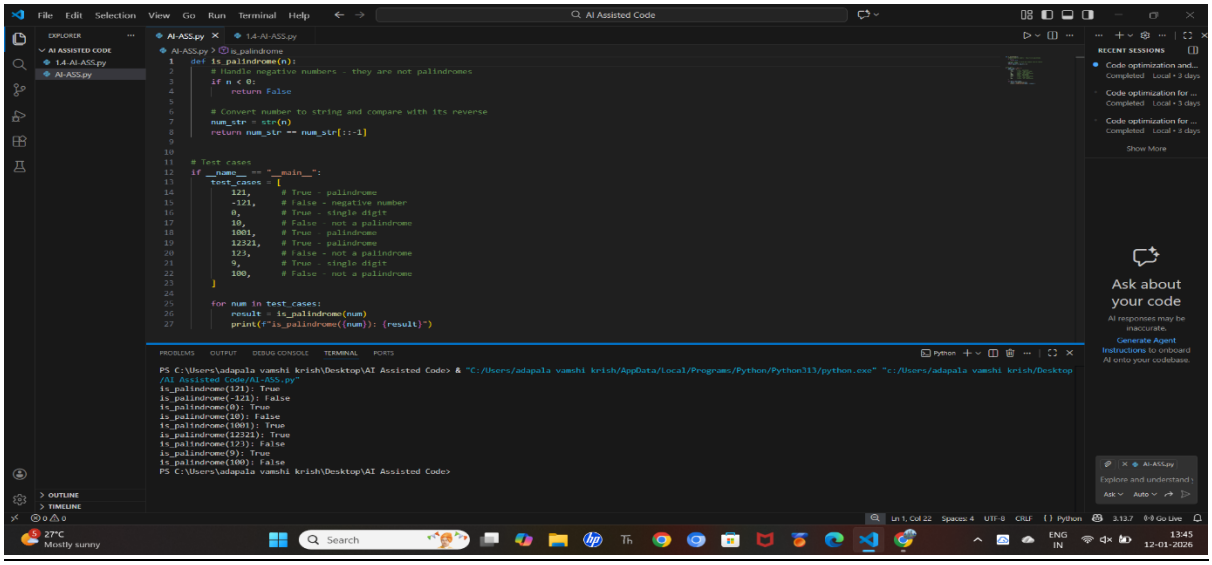
• Identify any logical errors or missing edge-case handling.

PROMPT:

checking whether given number it is palindrome or not

CODE:





Question 2:
One-Shot Prompting (Factorial Calculation)

Write a one-shot prompt by providing one input-output example and ask the AI to generate a Python function to compute the factorial of a given number.

Example:

Input: 5 → Output: 120

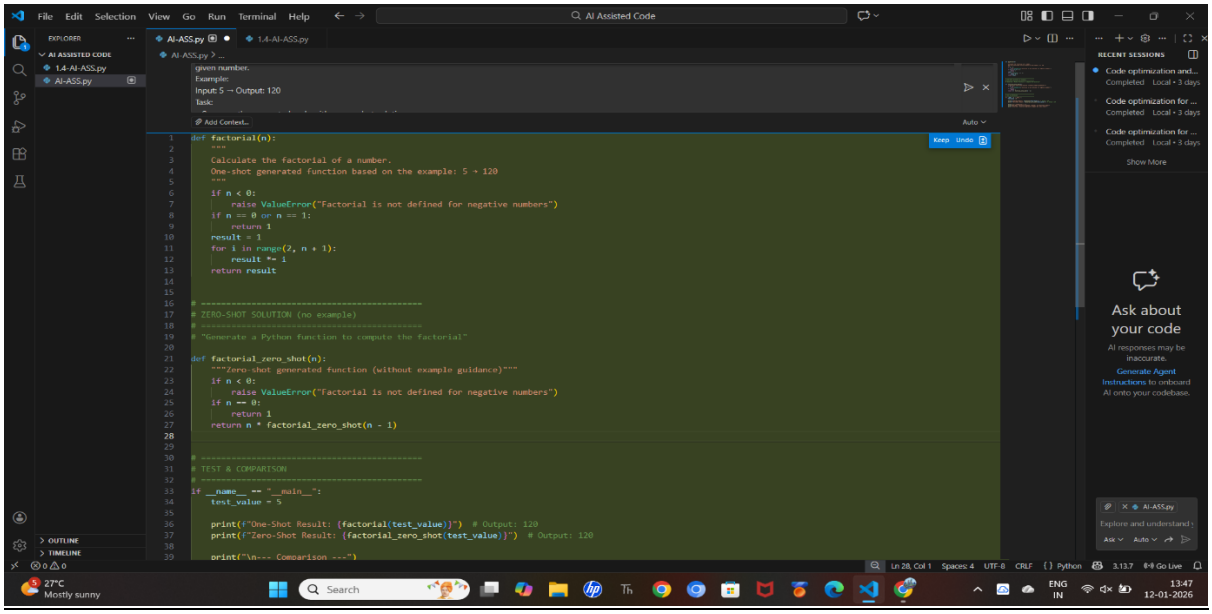
Task:

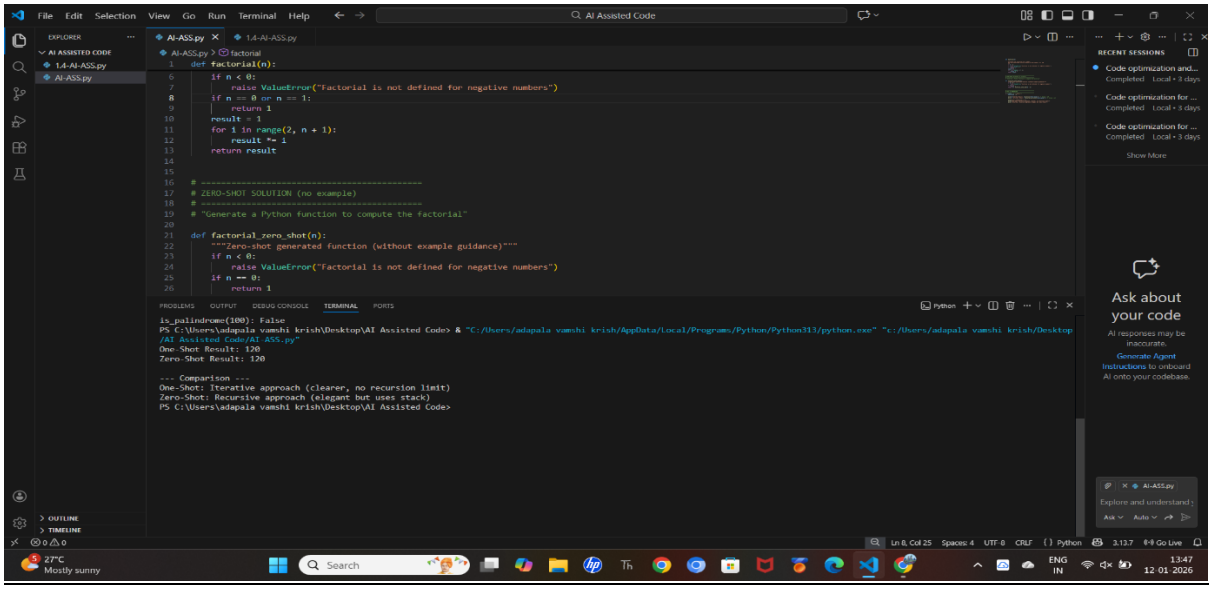
• Compare the generated code with a zero-shot solution.

• Examine improvements in clarity and correctness.

PROMPT: **Factorial Calculation giving some instructions input data**

CODE:





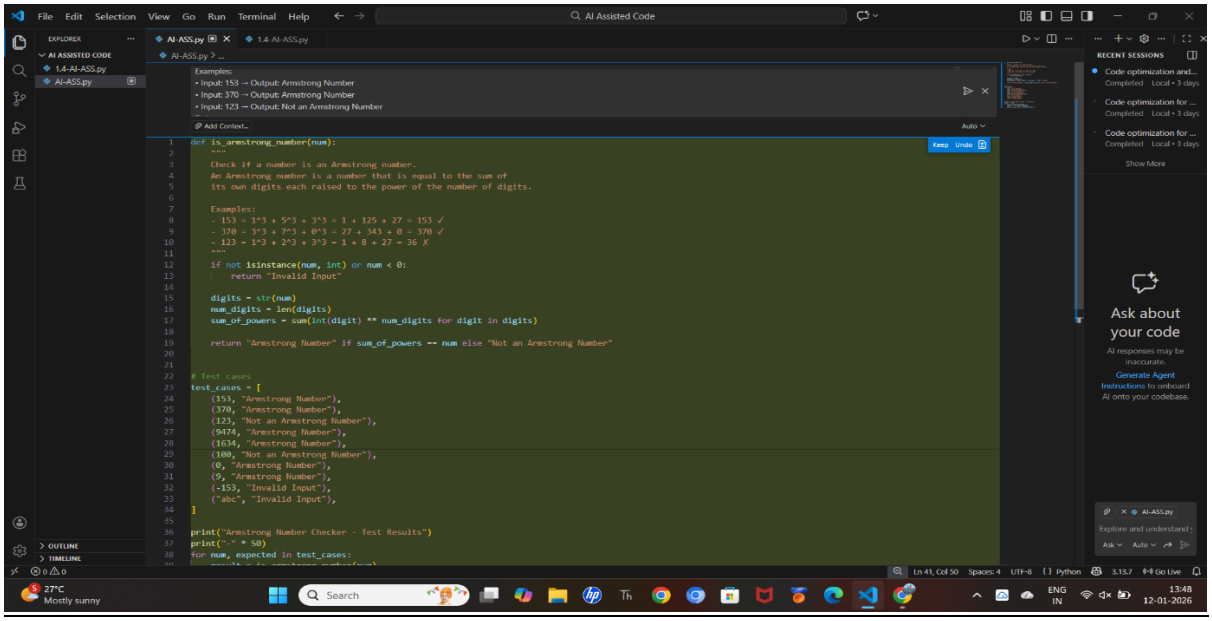
Question 3: Few-Shot Prompting (Armstrong Number Check)

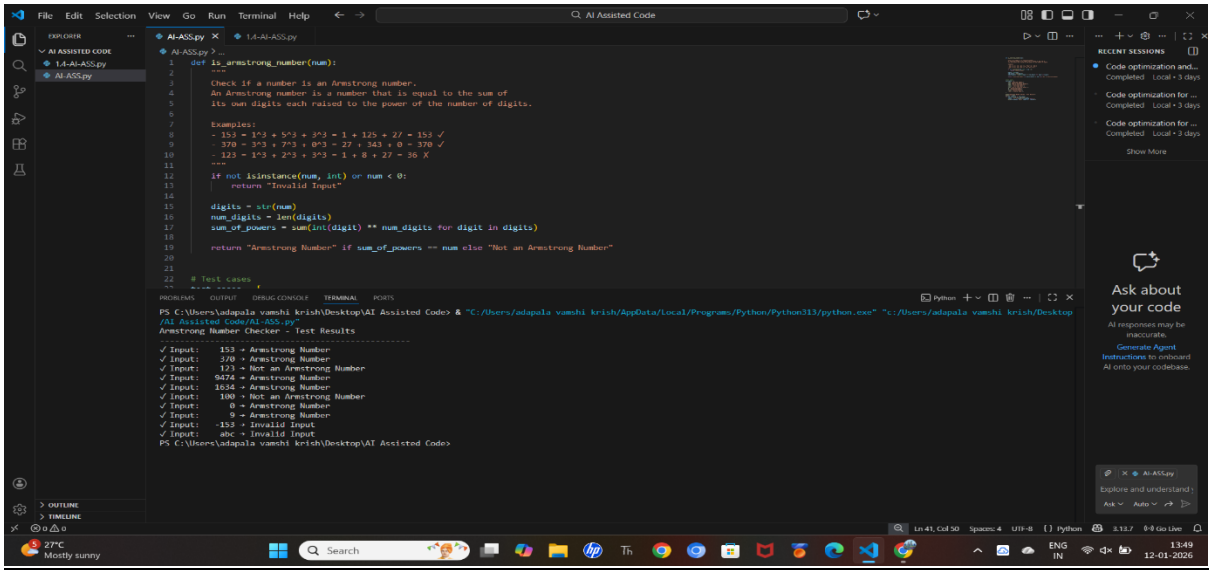
Write a few-shot prompt by providing multiple input-output examples to guide the AI in generating a Python function to check whether a given number is an Armstrong number.
Examples:
• Input: 153 → Output: Armstrong Number
• Input: 370 → Output: Armstrong Number
• Input: 123 → Output: Not an Armstrong Number

Task:
• Analyze how multiple examples influence code structure and accuracy.
• Test the function with boundary values and invalid inputs.

PROMPT: Armstrong Number Check
• Input: 153 → Output: Armstrong Number
• Input: 370 → Output: Armstrong Number
• Input: 123 → Output: Not an Armstrong Number

CODE:





Question 4: Context-Managed Prompting (Optimized Number Classification)

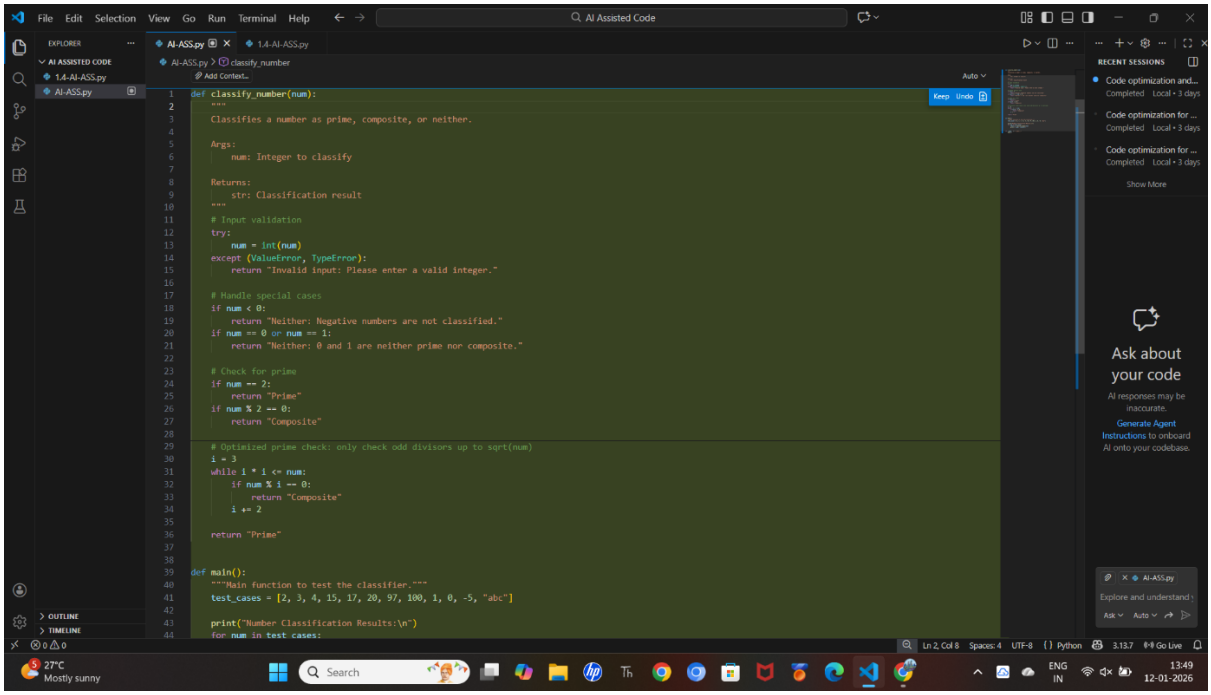
Design a context-managed prompt with clear instructions and
constraints to generate an optimized Python program that classifies a number as prime, composite, or neither.

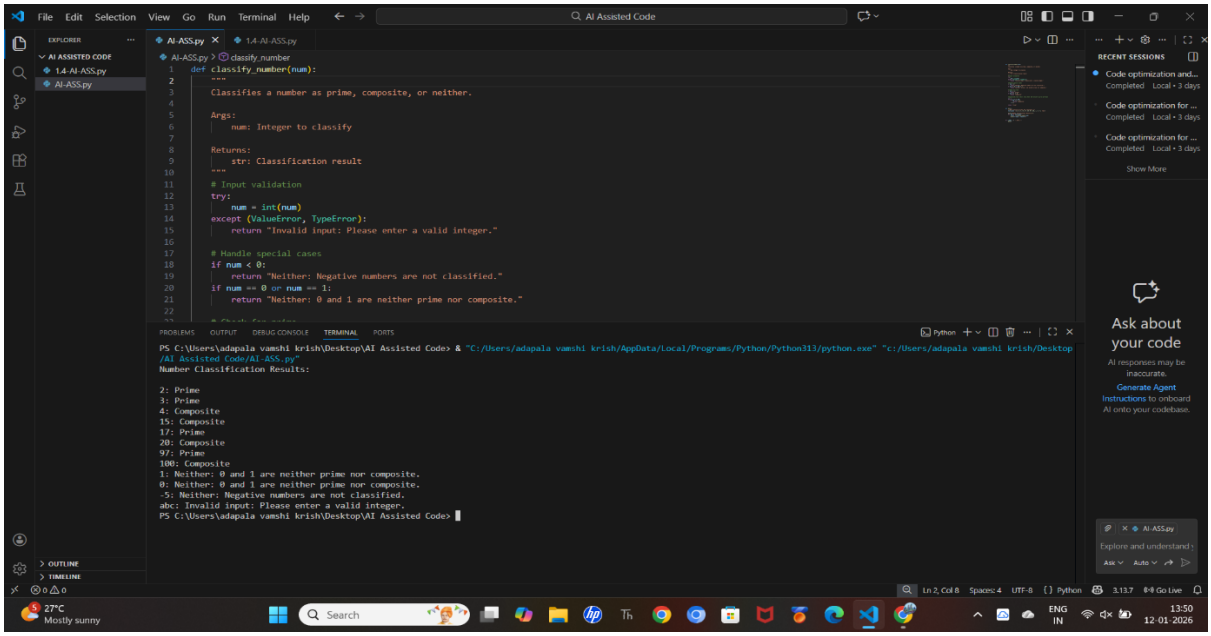
Task:

- Ensure proper input validation.
- Optimize the logic for efficiency.
- Compare the output with earlier prompting strategies.

PROMPT: Optimized Number Classification

CODE:





Question 5: Zero-Shot Prompting (Perfect Number Check)

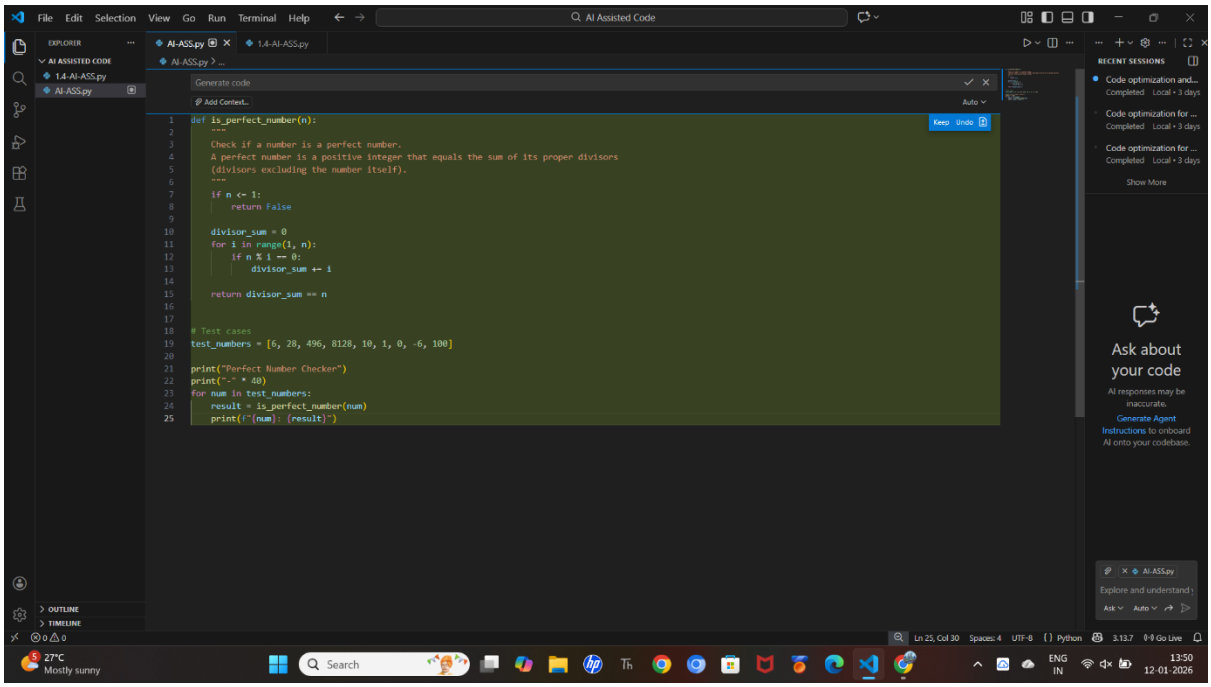
Write a zero-shot prompt (without providing any examples) to generate a Python function that checks whether a given number is a perfect number.

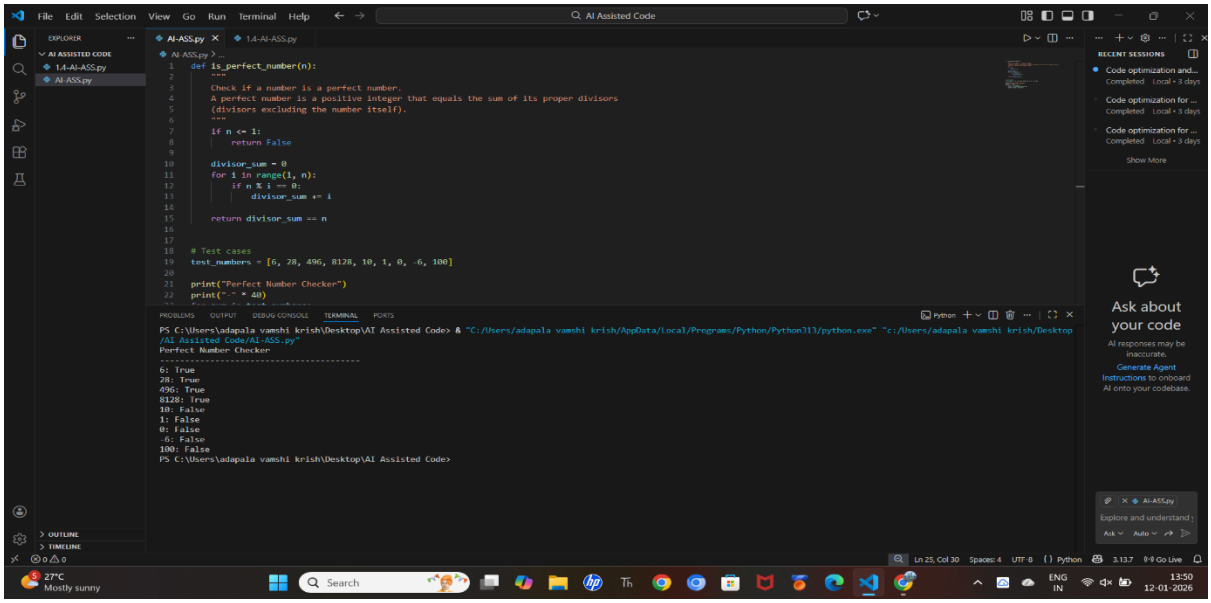
Task:

- Record the AI-generated code.
- Test the program with multiple inputs.
- Identify any missing conditions or inefficiencies in the logic.

PROMPT: **Perfect Number Check without providing any examples**

CODE:





Question 6: Few-Shot Prompting (Even or Odd Classification with Validation)

Write a few-shot prompt by providing multiple input-output examples to guide the AI in generating a Python program that determines whether a given number is even or odd, including proper input validation.

Examples:

- Input: 8 → Output: Even
- Input: 15 → Output: Odd
- Input: 0 → Output: Even

Task:

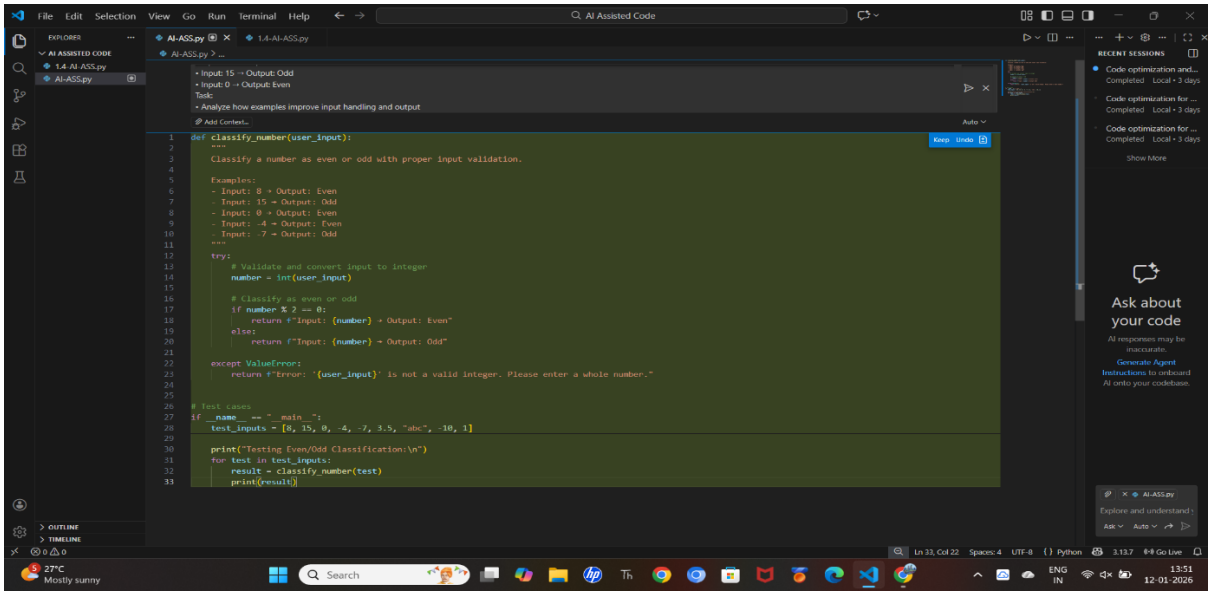
- Analyze how examples improve input handling and output clarity.
- Test the program with negative numbers and non-integer inputs.

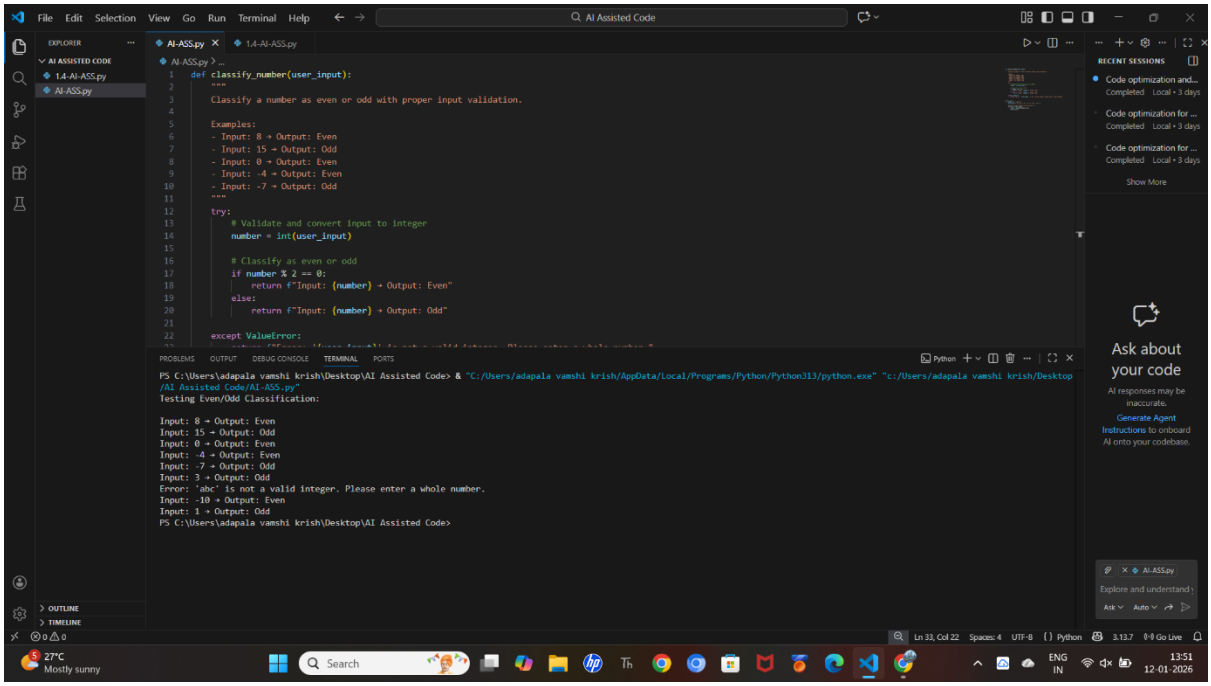
PROMPT:

Even or Odd Classification with Validation

- Input: 8 → Output: Even
- Input: 15 → Output: Odd
- Input: 0 → Output: Even

CODE:




File Edit Selection View Go Run Terminal Help
AI Assisted Code
EXPLORER
AI ASSISTED CODE
1.4-AI-ASS.py
AI-ASS.py
AI-ASS.py
1.4-AI-ASS.py
def classify_number(user_input):
"""
Classify a number as even or odd with proper input validation.
Examples:
- Input: 8 → Output: Even
- Input: 15 → Output: Odd
- Input: 0 → Output: Even
- Input: -4 → Output: Even
- Input: -7 → Output: Odd
"""
try:
# Validate and convert input to integer
number = int(user_input)
# Classify as even or odd
if number % 2 == 0:
return f"Input: {number} → Output: Even"
else:
return f"Input: {number} → Output: Odd"
except ValueError:
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python
PS C:\Users\adapala vamshi krish\Desktop\AI Assisted Code> & "C:/Users/adapala vamshi krish/AppData/Local/Programs/Python/Python313/python.exe" "c:/Users/adapala vamshi krish/Desktop/AI Assisted Code/AI-ASS.py"
Testing Even/Odd Classification:
Input: 8 → Output: Even
Input: 15 → Output: Odd
Input: 0 → Output: Even
Input: -4 → Output: Even
Input: -7 → Output: Odd
Input: 3 → Output: Odd
Error: 'abc' is not a valid integer. Please enter a whole number.
Input: -10 → Output: Even
Input: 1 → Output: Odd
PS C:\Users\adapala vamshi krish\Desktop\AI Assisted Code>
RECENT SESSIONS
Code optimization and...
Completed Local • 3 days
Code optimization for ...
Completed Local • 3 days
Code optimization for ...
Completed Local • 3 days
Show More
Ask about your code
AI responses may be inaccurate.
Generate Agent Instructions to onboard AI onto your codebase.
AI-ASS.py
Explore and understand
Ask
Auto
OUTLINE
TIMELINE
Ln 33, Col 22 Spaces: 4 UTF-8 Python 3.13.7 Go Live
27°C
Mostly sunny
Search
ENG
IN
13:51
12-01-2026