N.Sai Akash

2403A51L57

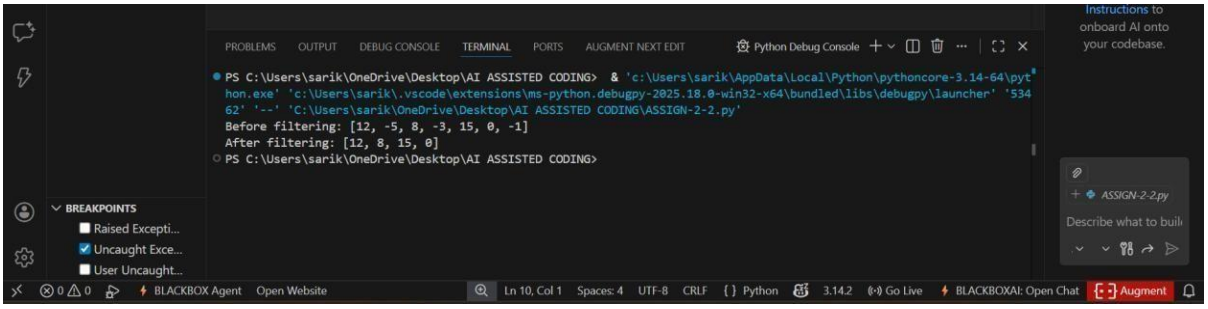
B-52

ASSIGNMENT -2.2

Task 1: Cleaning Sensor Data

PROMPT: Create a Python function that removes all negative values from a list of sensor readings.

OUTPUT:

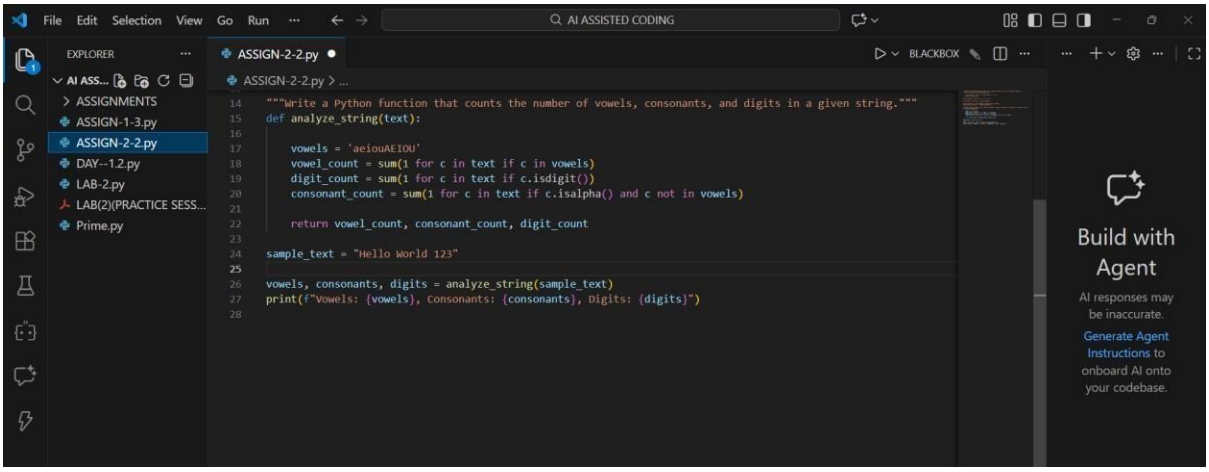


EXPLANATION:

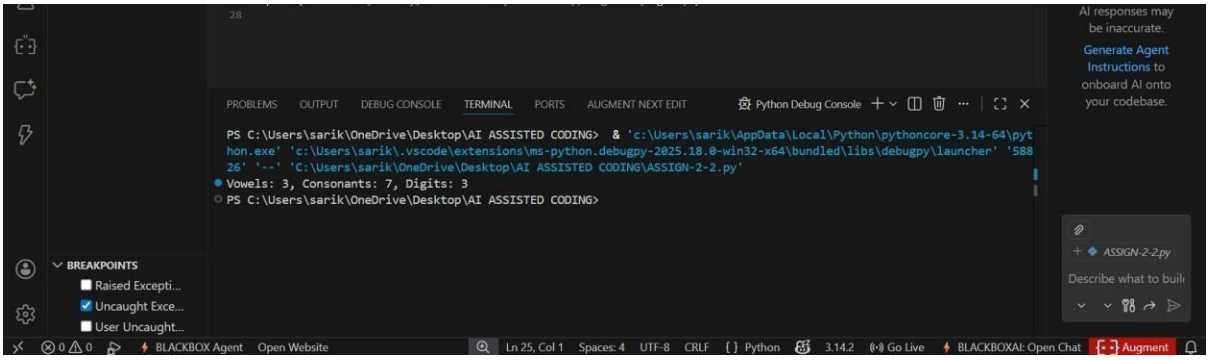
This function removes invalid negative sensor values using list comprehension. Only values greater than or equal to zero are retained, ensuring clean IoT sensor data.

Task 2: String Character Analysis

PROMPT: Write a Python function that counts the number of vowels, consonants, and digits in a given string.



OUTPUT:



EXPLANATION:

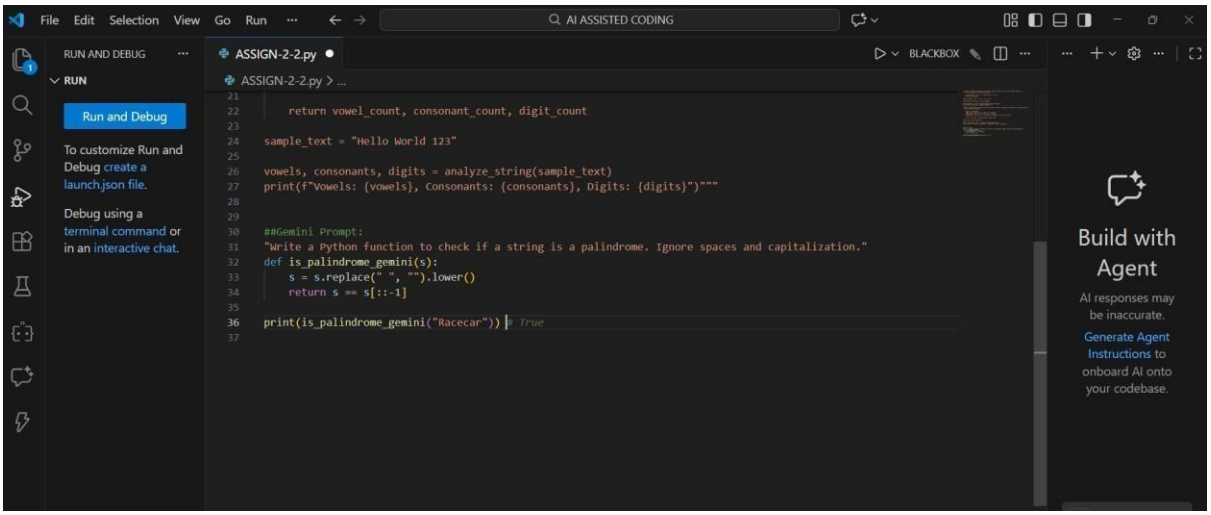
The function iterates through each character and classifies it as a vowel, consonant, or digit.

Python string methods like isalpha() and isdigit() improve accuracy and readability.

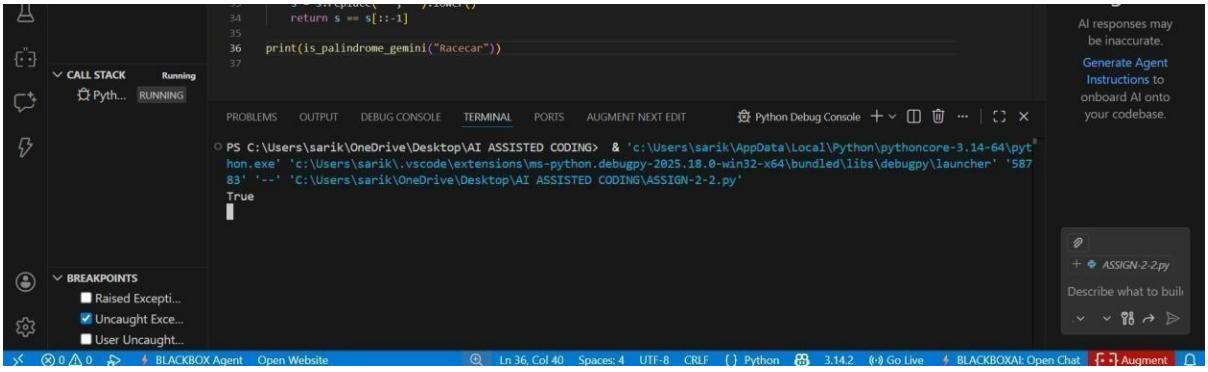
Task 3: Palindrome Check – Tool Comparison

Gemini Prompt: Write a Python function to check if a string is a palindrome.

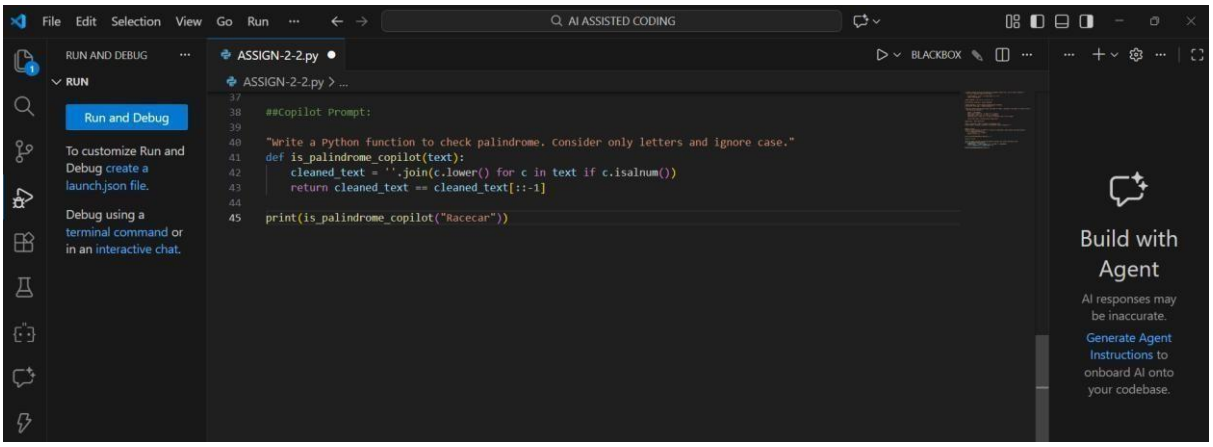
Ignore spaces and capitalization.



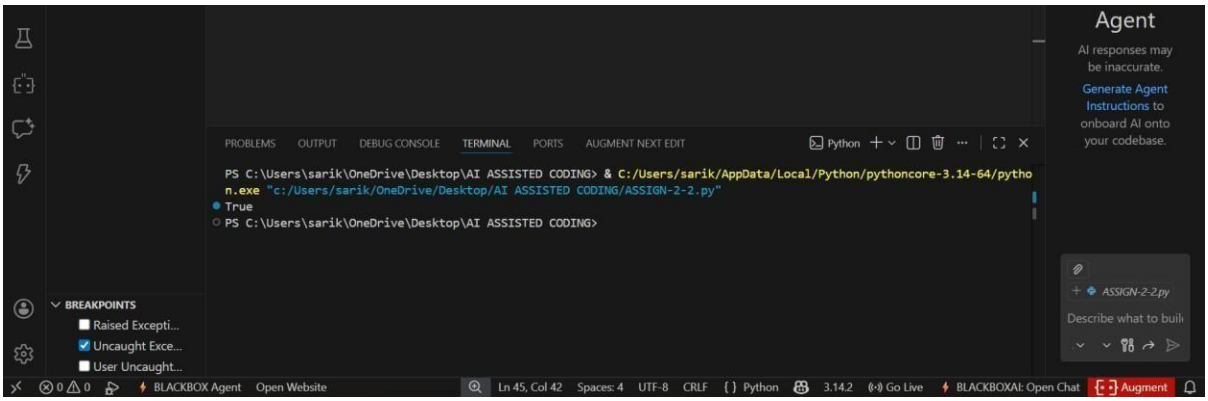
OUTPUT:



Copilot Prompt: Write a Python function to check palindrome. Consider only letters and ignore case.



OUTPUT:



Comparison Table:

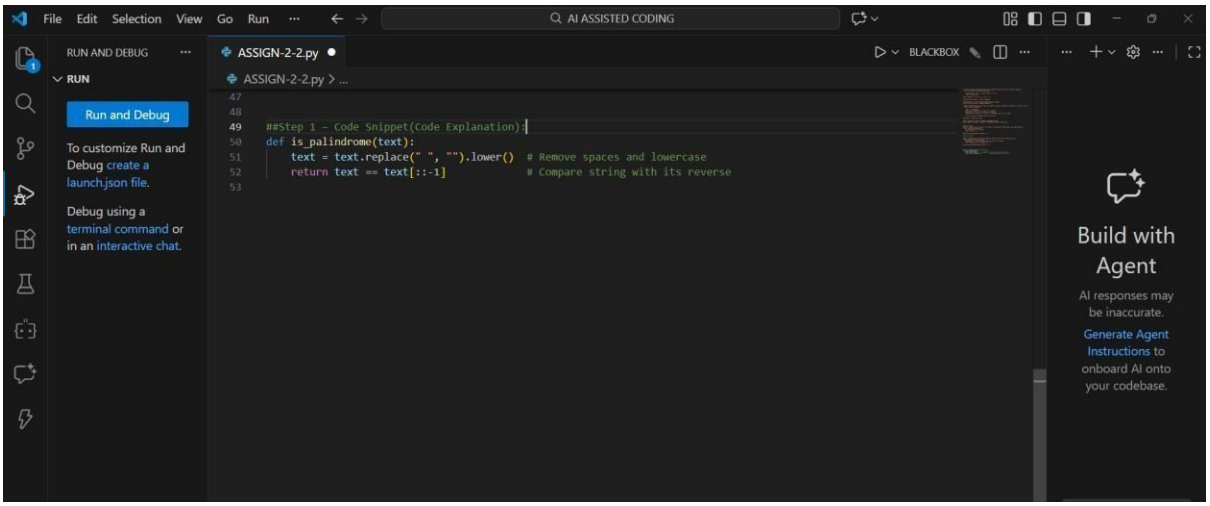
Feature	Gemini	Copilot
Clarity	Simple, minimal code	Slightly longer, more robust
Handling spaces/case	Ignores spaces, converts to lowercase	Ignores spaces and punctuation, lowercase
Readability	Very clear	Clear, slightly more detailed
Efficiency	Uses string slicing	Uses string comprehension

EXPLANATION:

Gemini provides concise and easy-to-read logic, making it beginnerfriendly. Copilot generates more robust code that handles punctuation and special characters.

Task 4: Code Explanation Using AI Step 1 –

Code Snippet:



Step 2 – AI Explanation:

1. text.replace(" ", "").lower() → Removes spaces and converts letters to lowercase.
2. text == text[::-1] → Checks if the string is equal to its reverse.

EXPLANATION:

The function normalizes the string to avoid case and space mismatches.
It then compares the string with its reverse to verify palindrome logic.