

# Amruth Sagar

2403A51L44

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.



The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows files like `ASSIGN-2.py`, `ASSIGN-2-2.py`, `ASSIGN-1-3.py`, `ASSIGN-2-2.py`, `DAY-12.py`, `LAB-2.py`, `LAB(2)PRACTICE SESSION...`, and `Prime.py`.
- Code Editor:** Displays the content of `ASSIGN-2-2.py`. The code defines a function `filter_negative_numbers` that takes a list of sensor data and returns a new list with all negative values removed. It also prints the original and filtered lists.
- Toolbar:** Includes standard file operations (File, Edit, Selection, View, Go, Run, etc.) and a search bar labeled "Q AI ASSISTED CODING".
- Right Panel:** Features a "Build with Agent" button and a message stating "All responses may be inaccurate." Below it is a "Generate Agent" button.

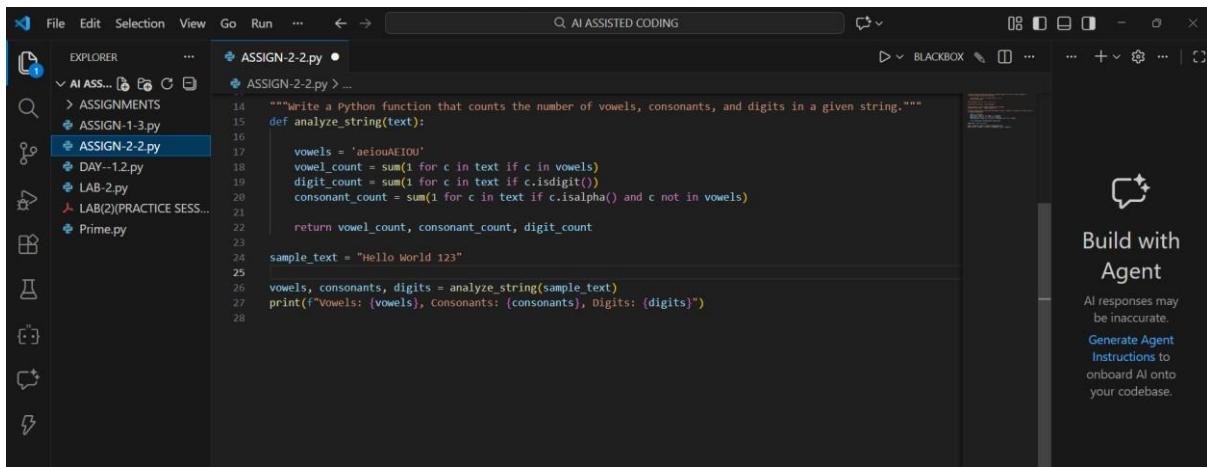
## **OUTPUT:**

## **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.

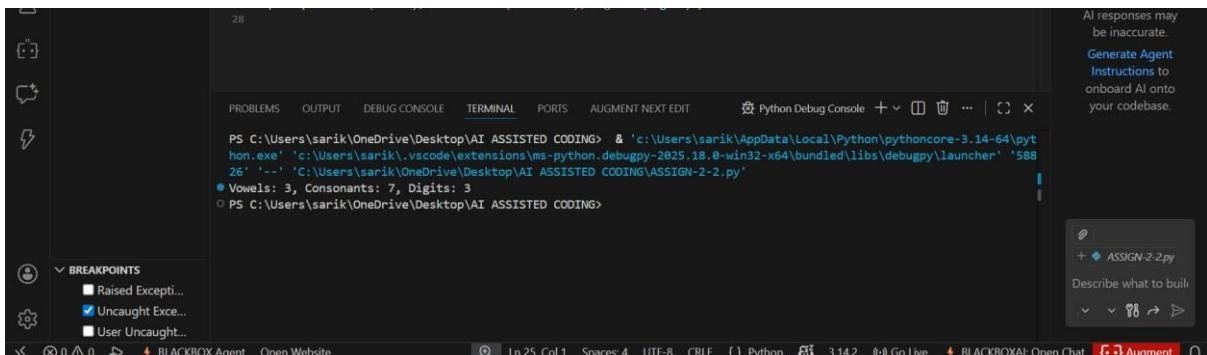


A screenshot of the Visual Studio Code interface. The left sidebar shows a file tree with several Python files: ASSIGN-1-3.py, DAY--1.2.py, LAB(2).PRACTICE SESSION..., and Prime.py. The main editor area contains the following Python code:

```
14 """Write a Python function that counts the number of vowels, consonants, and digits in a given string."""
15 def analyze_string(text):
16     vowels = 'aeiouAEIOU'
17     vowel_count = sum(1 for c in text if c in vowels)
18     digit_count = sum(1 for c in text if c.isdigit())
19     consonant_count = sum(1 for c in text if c.isalpha() and c not in vowels)
20
21     return vowel_count, consonant_count, digit_count
22
23 sample_text = "Hello World 123"
24
25
26 vowels, consonants, digits = analyze_string(sample_text)
27 print(f"Vowels: {vowels}, Consonants: {consonants}, Digits: {digits}")
```

The right sidebar features a "Build with Agent" panel with a message: "AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase."

**OUTPUT:**



A screenshot of the VS Code terminal window. The command `python ASSIGN-2-2.py` is run, and the output is:

```
PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING> python ASSIGN-2-2.py
● Vowels: 3, Consonants: 7, Digits: 3
○ PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING>
```

The terminal also displays the Python environment details: Python 3.14.2, Go Live, and BLACKBOX Agent.

**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.

The screenshot shows the VS Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** AI ASSISTED CODING
- Left Sidebar (RUN AND DEBUG):**
  - RUN
  - Run and Debug
  - To customize Run and Debug create a launch.json file.
  - Debug using a terminal command or in an interactive chat.
- Code Editor:** File: ASSIGN-2-2.py  
Content:

```
21     return vowel_count, consonant_count, digit_count
22
23     sample_text = "Hello World 123"
24
25     vowels, consonants, digits = analyze_string(sample_text)
26     print(f"Vowels: {vowels}, Consonants: {consonants}, Digits: {digits}"")
27
28
29     ##Gemini Prompt:
30     "#Write a Python function to check if a string is a palindrome. Ignore spaces and capitalization."
31     def is_palindrome_gemini(s):
32         s = s.replace(" ", "").lower()
33         return s == s[::-1]
34
35
36     print(is_palindrome_gemini("Racecar"))
```
- Output Panel:** Shows the output of the code execution.
- Right Sidebar:**
  - Build with Agent
  - AI responses may be inaccurate.
  - Generate Agent Instructions to onboard AI onto your codebase.

## OUTPUT:

The screenshot shows the VS Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** AI ASSISTED CODING
- Left Sidebar (RUN AND DEBUG):**
  - CALL STACK Running
  - Pyth... RUNNING
  - BREAKPOINTS
- Code Editor:** File: ASSIGN-2-2.py  
Content:

```
34     return s == s[::-1]
35
36     print(is_palindrome_gemini("Racecar"))
```
- Terminal:** Shows the command run and the output: True
- Output Panel:** Shows the output of the code execution.
- Right Sidebar:**
  - AI responses may be inaccurate.
  - Generate Agent Instructions to onboard AI onto your codebase.

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

The screenshot shows the VS Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** AI ASSISTED CODING
- Left Sidebar (RUN AND DEBUG):**
  - RUN
  - Run and Debug
  - To customize Run and Debug create a launch.json file.
  - Debug using a terminal command or in an interactive chat.
- Code Editor:** File: ASSIGN-2-2.py  
Content:

```
37
38     ##Copilot Prompt:
39
40     "#Write a Python function to check palindrome. Consider only letters and ignore case."
41     def is_palindrome_copilot(text):
42         cleaned_text = ''.join(c.lower() for c in text if c.isalnum())
43         return cleaned_text == cleaned_text[::-1]
44
45     print(is_palindrome_copilot("Racecar"))
```
- Output Panel:** Shows the output of the code execution.
- Right Sidebar:**
  - Build with Agent
  - AI responses may be inaccurate.
  - Generate Agent Instructions to onboard AI onto your codebase.

## OUTPUT:

The screenshot shows the Visual Studio Code interface with the following elements:

- Terminal:** Displays command-line output for Python 3.14.2. It shows the command `python.exe "c:/Users/sarik/OneDrive/Desktop/AI ASSISTED CODING/ASSIGN-2-2.py"`, followed by a True result and a note about AI assisted coding.
- Code Editor:** Shows a Python file named `ASSIGN-2-2.py`.
- AI Agent Sidebar:** Located on the right, it says "Agent" and "AI responses may be inaccurate." It includes a "Generate Agent Instructions" button and a note about onboard AI onto your codebase.
- Breakpoints:** A sidebar panel shows breakpoints: "Raised Excepti..." (unchecked), "Uncaught Exce..." (checked), and "User Uncaught..." (unchecked).
- Bottom Status Bar:** Shows file path, line number (Ln 45, Col 42), spaces, encoding (UTF-8), Python version (3.14.2), Go Live button, BLACKBOXAI Open Chat, and Augment button.

## 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:

The screenshot shows a code editor interface with a dark theme. On the left, there's a sidebar with icons for file operations like Open, Save, and Run, along with sections for RUN AND DEBUG and RUN. The main area displays a Python file named 'ASSIGN-2-2.py'. The code contains two lines of Python code:

```
49     ##Step 1 - Code Snippet(Code Explanation)
50     def is_palindrome(text):
51         text = text.replace(" ", "").lower() # Remove spaces and lowercase
52         return text == text[::-1]           # Compare string with its reverse
```

To the right of the code, there's a sidebar titled 'Build with Agent' which includes a note that AI responses may be inaccurate and a link to 'Generate Agent Instructions'.

## 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.