

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.



File Edit Selection View Go Run ... ⏪ ⏩ Q AI ASSISTED CODING

EXPLORER

AI ASSISTED CODING

ASSIGN-2.py > ...

ASSIGN-2.py > ...

```
1 """Create a Python function that removes all negative values from a list of sensor readings."""
2 def filter_negative_numbers(sensor_data):
3     ...
4     filtered_data = [x for x in sensor_data if x >= 0]
5     return filtered_data
6
7 sensor_readings = [12, -5, 8, -3, 15, 0, -1]
8
9 print("Before filtering:", sensor_readings)
10
11 cleaned_readings = filter_negative_numbers(sensor_readings)
12 print("After filtering:", cleaned_readings)
```

BLACKBOX

Build with Agent

All responses may be inaccurate.

Generate Agent

## OUTPUT:

The screenshot shows the VS Code interface with the Python extension installed. The top navigation bar includes 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', 'PORTS', 'ALIGNMENT NEXT EDIT', and a 'Python Debug Console' tab which is currently active. The main area displays the following terminal output:

```
PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING> & 'c:\Users\sarik\AppData\Local\Python\pythoncore\3.14-64\python.exe' 'c:\Users\sarik\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '53462' '--' 'C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING\ASSIGN-2-2.py'
Before filtering: [12, -5, 8, -3, 15, 0, -1]
After filtering: [12, 8, 15, 0]
PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING>
```

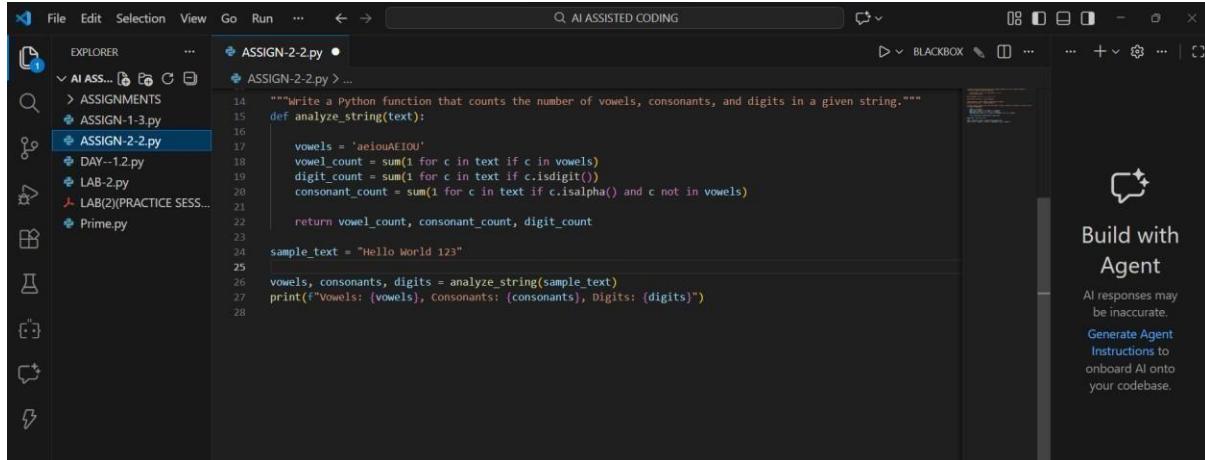
On the left, the 'BREAKPOINTS' sidebar is open, showing three breakpoint categories: 'Raised Except...', 'Uncought Except...', and 'User Uncaught...', with 'Uncought Except...' checked.

## **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.

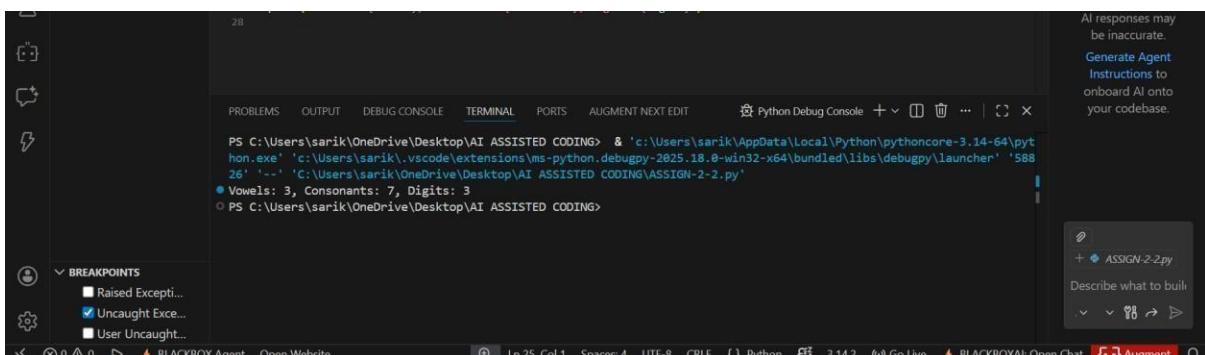


The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows files like `ASSIGN-1-3.py`, `ASSIGN-2-2.py` (selected), `DAY-1-2.py`, `LAB-2.py`, and `Prime.py`.
- Code Editor:** Displays the Python script `ASSIGN-2-2.py` containing the following 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
24     sample_text = "Hello World 123"
25
26     vowels, consonants, digits = analyze_string(sample_text)
27     print(f"Vowels: {vowels}, Consonants: {consonants}, Digits: {digits}")
```
- Right Panel:** A sidebar titled "Build with Agent" with the sub-section "Agent Instructions". It includes a note: "AI responses may be inaccurate.", a button "Generate Agent Instructions", and a note: "Instructions to onboard AI onto your codebase."

### OUTPUT:



The screenshot shows the terminal window in VS Code displaying the following output:

```
PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING> & 'c:\Users\sarik\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\sarik\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '58826' '--' 'C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING\ASSIGN-2-2.py'
● Vowels: 3, Consonants: 7, Digits: 3
○ PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING>
```

### EXPLANATION:

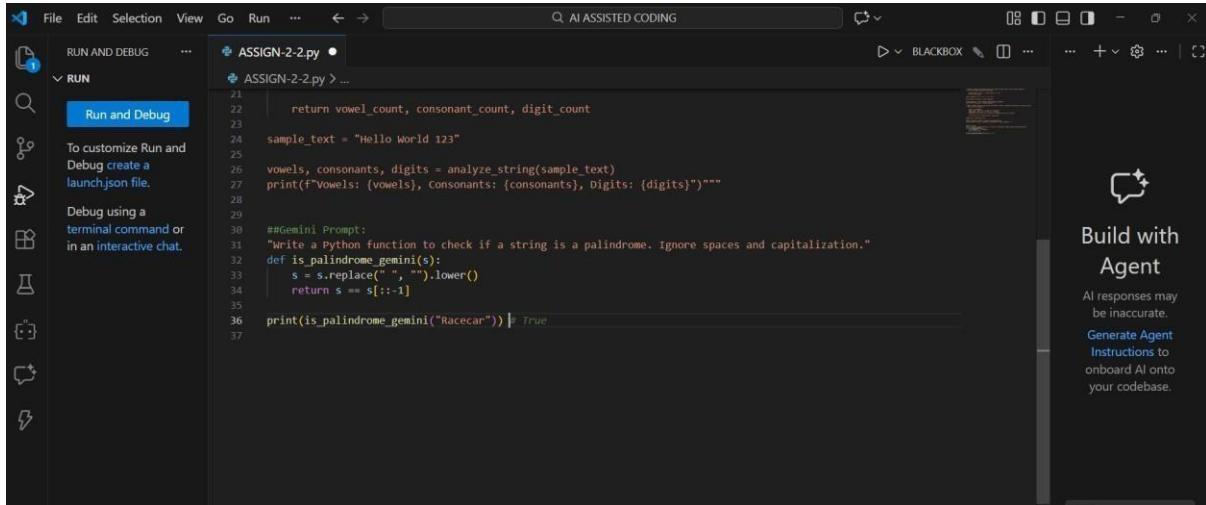
The function iterates through each character and classifies it as a vowel, consonant, or digit.

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.



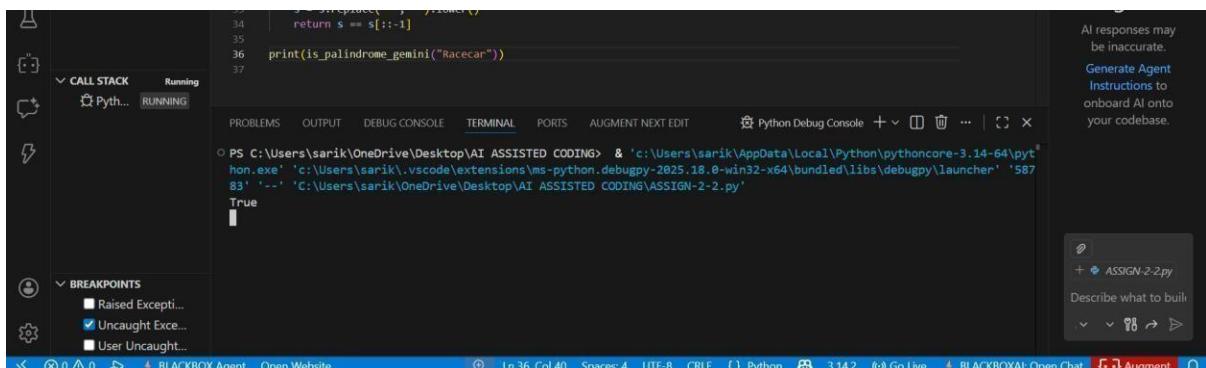
A screenshot of the Visual Studio Code interface. The left sidebar shows the 'RUN AND DEBUG' section with 'Run and Debug' selected. The main editor area contains Python code for checking if a string is a palindrome, ignoring spaces and capitalization. The code uses the `analyze\_string` function from a previous assignment to count vowels, consonants, and digits. It then defines a new function `is\_palindrome\_gemini` that removes spaces from the input string and compares it to its reverse. The output of the code is shown in the terminal below, indicating it returns 'True' for 'Racecar'.

```
sample_text = "Hello World 123"
vowels, consonants, digits = analyze_string(sample_text)
print(f"Vowels: {vowels}, Consonants: {consonants}, Digits: {digits}")


##Gemini Prompt:
#Write a Python function to check if a string is a palindrome. Ignore spaces and capitalization.
def is_palindrome_gemini(s):
    s = s.replace(" ", "").lower()
    return s == s[::-1]

print(is_palindrome_gemini("Racecar"))
```

## OUTPUT:

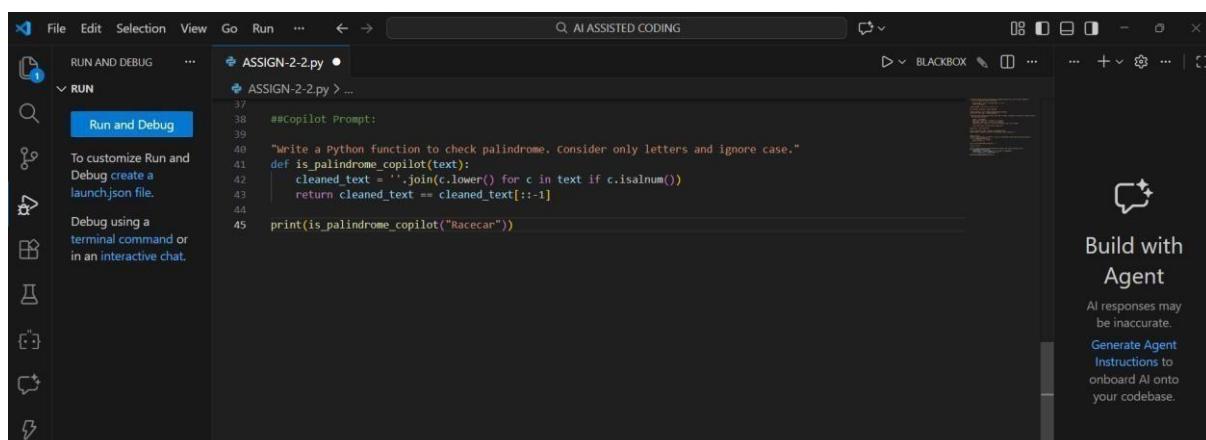


A screenshot of the Visual Studio Code interface, specifically the 'RUN AND DEBUG' view. It shows the 'CALL STACK' tab is active, with 'Python' listed as 'RUNNING'. The terminal output shows the result of running the code, which prints 'True' to the console. The right sidebar has a 'Build with Agent' panel.

```
34     s = s.replace(" ", "").lower()
35
36     print(is_palindrome_gemini("Racecar"))

PS C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING> & 'c:\Users\sarik\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\sarik\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '58783' '--' 'C:\Users\sarik\OneDrive\Desktop\AI ASSISTED CODING\ASSIGN-2-2.py'
True
```

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

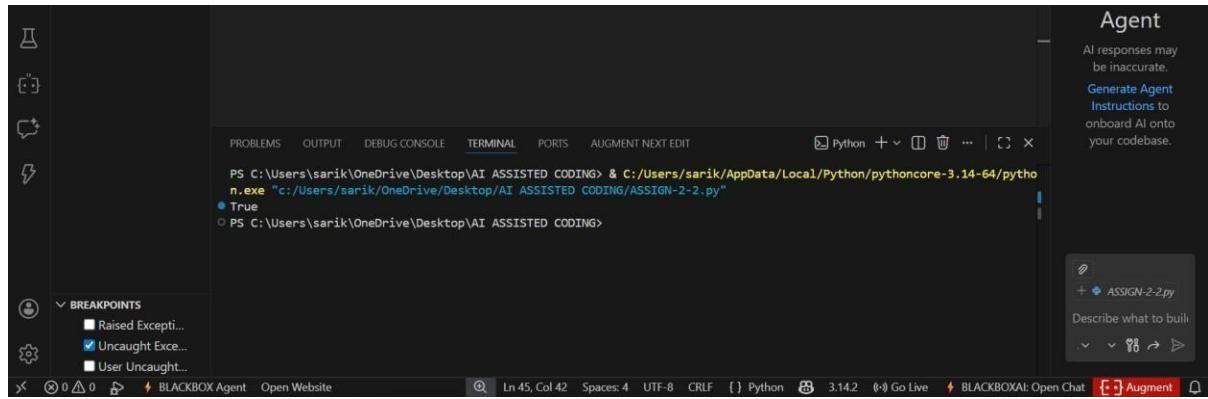


A screenshot of the Visual Studio Code interface. The left sidebar shows the 'RUN AND DEBUG' section with 'Run and Debug' selected. The main editor area contains Python code for checking if a string is a palindrome, considering only letters and ignoring case. The code uses the `isalnum` method to filter out non-alphanumeric characters before comparing the string to its reverse. The output of the code is shown in the terminal below, indicating it returns 'True' for 'Racecar'.

```
##Copilot Prompt:
#Write a Python function to check palindrome. Consider only letters and ignore case.
def is_palindrome_copilot(text):
    cleaned_text = ''.join(c.lower() for c in text if c.isalnum())
    return cleaned_text == cleaned_text[::-1]

print(is_palindrome_copilot("Racecar"))
```

## 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:

The screenshot shows a code editor interface with a dark theme. On the left, there's a sidebar with icons for file operations like Run, Debug, and Search. The main area displays a Python script named 'ASSIGN-2-2.py'. The code is as follows:

```
47
48
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
53
```

To the right of the code, there's an AI-assisted coding panel titled 'AI ASSISTED CODING' with a progress bar at 100%. It includes a 'Run and Debug' button and a message: 'To customize Run and Debug create a launch.json file.' Below this, there's a note: 'Debug using a terminal command or in an interactive chat.' On the far right, there's a 'Build with Agent' section with a message: 'AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase.'

## 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.