

AMRUTH SAGAR VEMUGANTI

2403A51L44

ASSIGNMENT-10.2

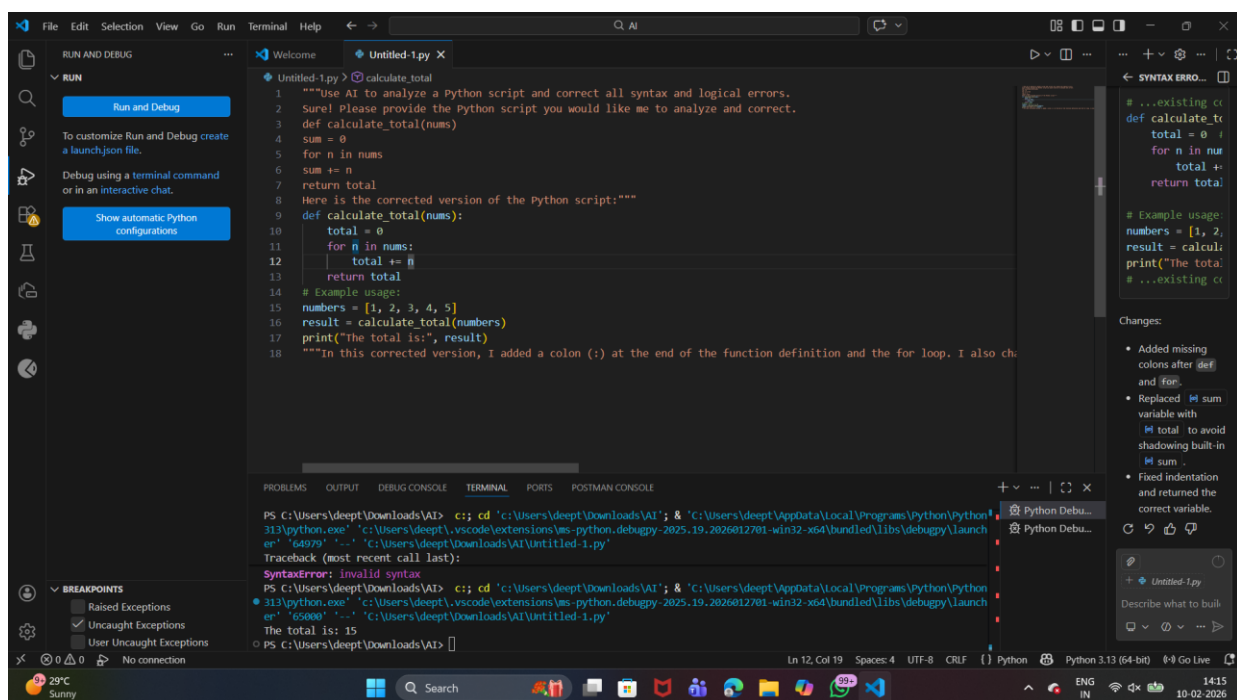
Code Review and Quality: Using AI to Improve Code

Quality and Readability

Task Description -1(Error Detection and Correction)

Prompt: Analyze the given Python code, identify all syntax and logical errors, and provide a corrected executable version with brief explanations

AI-Generated Code:



```
1 """Use AI to analyze a Python script and correct all syntax and logical errors.
2 Sure! Please provide the Python script you would like me to analyze and correct.
3 def calculate_total(nums)
4     sum = 0
5     for n in nums
6     sum += n
7     return total
8 Here is the corrected version of the Python script:'''
9 def calculate_total(nums):
10     total = 0
11     for n in nums:
12         total += n
13     return total
14 # Example usage:
15 numbers = [1, 2, 3, 4, 5]
16 result = calculate_total(numbers)
17 print("The total is:", result)
18 """In this corrected version, I added a colon (:) at the end of the function definition and the for loop. I also changed the variable name from 'sum' to 'total' to avoid shadowing the built-in 'sum' function.
```

```
PS C:\Users\deep\Downloads\AI> c:\python.exe -c: c:\Users\deep\Downloads\AI\ & 'c:\Users\deep\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\deep\vscode\extensions\ms-python.debugpy-2025.19.2026012701-win32-x64\bundle\libs\debugpy\launch' '65000' '-' 'c:\Users\deep\Downloads\AI\untitled-1.py'
Traceback (most recent call last):
SyntaxError: invalid syntax
PS C:\Users\deep\Downloads\AI> c:\python.exe -c: c:\Users\deep\Downloads\AI\ & 'c:\Users\deep\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\deep\vscode\extensions\ms-python.debugpy-2025.19.2026012701-win32-x64\bundle\libs\debugpy\launch' '65000' '-' 'c:\Users\deep\Downloads\AI\untitled-1.py'
The total is: 15
PS C:\Users\deep\Downloads\AI>
```

Output:

```
The total is: 15
PS C:\Users\deep\Downloads\AI>
```

Justification:

Missing colon after function definition → Added :

Missing colon in for loop → Added :

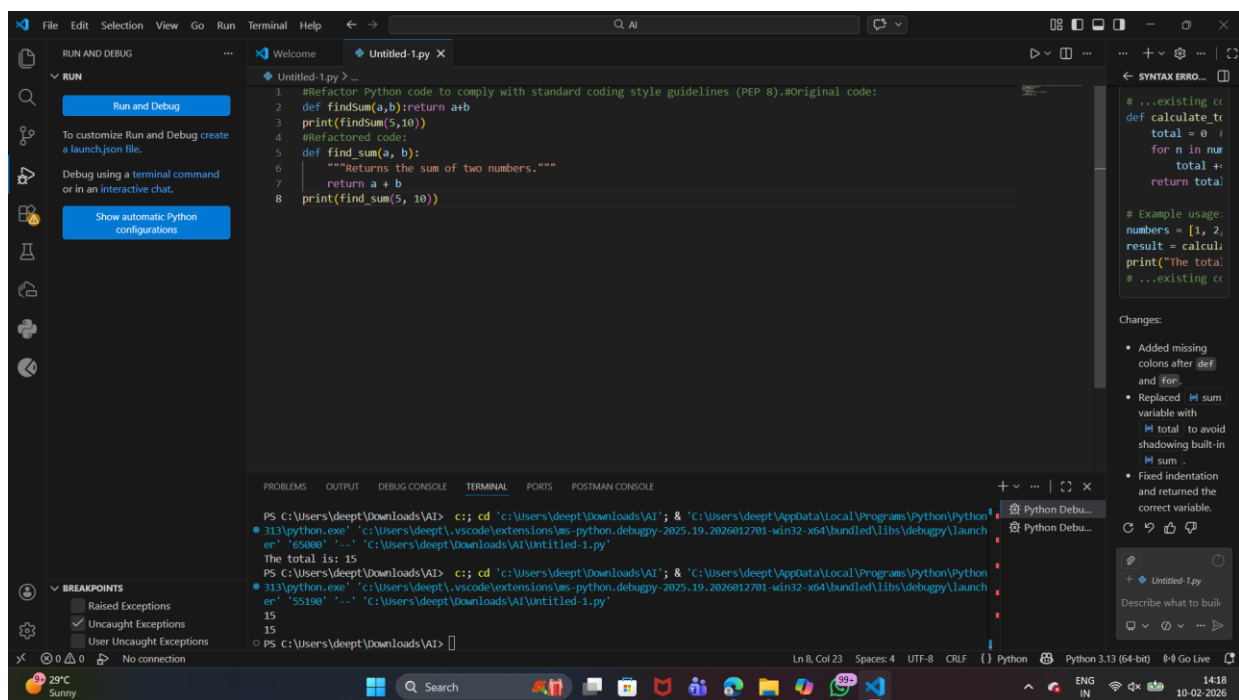
Used sum as variable name (overwrites built-in) → Renamed to total

Returned total though variable was named sum → Corrected consistency

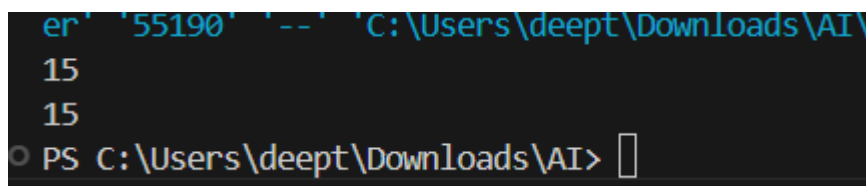
Task Description -2(Code Style Standardization)

Prompt : Refactor the Python code to follow PEP 8 coding standards with proper formatting and readability

AI-Generated Code:



OUTPUT:



Justification:

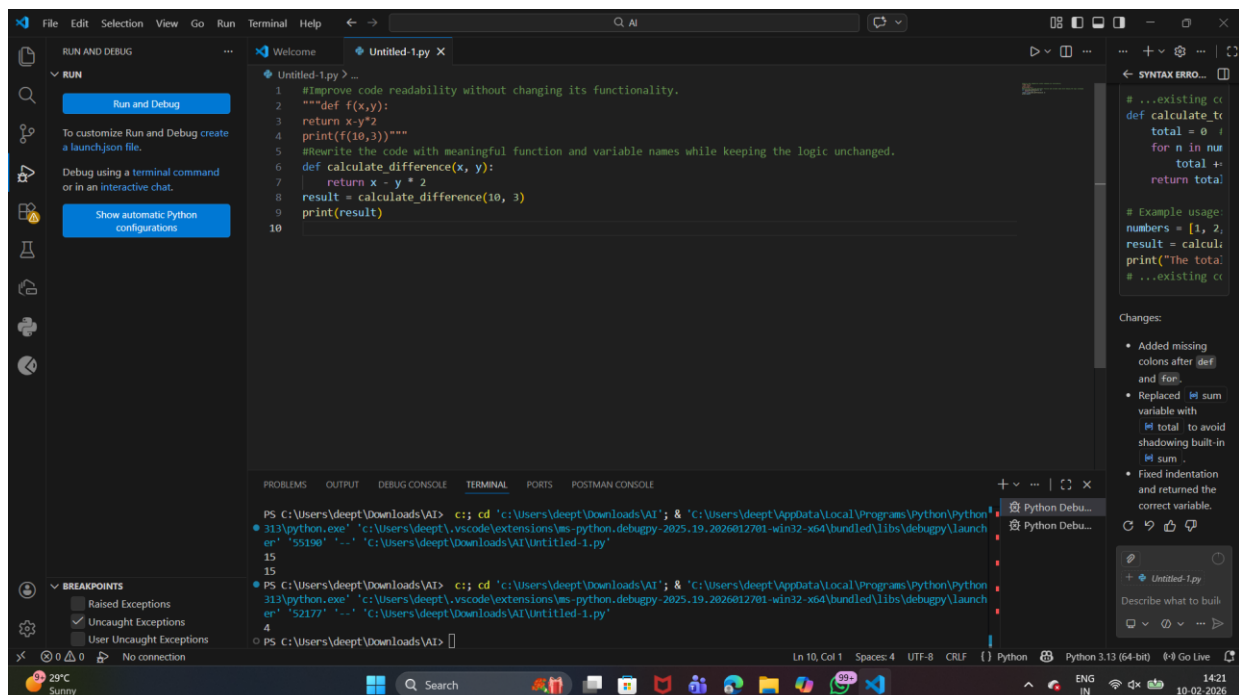
Added proper indentation and spacing

Used snake_case naming (PEP-8 standard)

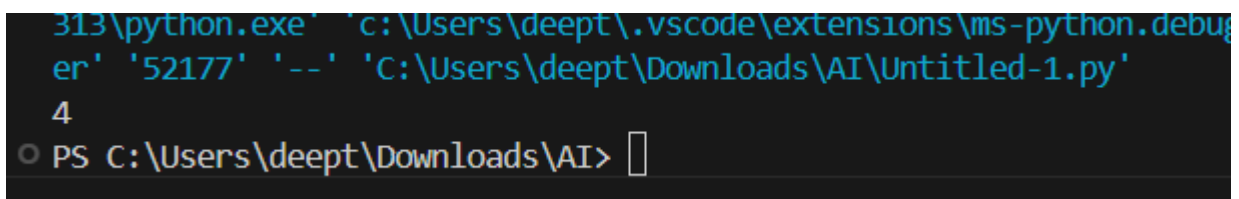
Split single-line function into readable multi-line format

Task Description -3(Code Clarity Improvement)

Prompt: Rewrite the code with meaningful function and variable names while keeping the logic unchanged.



OUTPUT:



Justification:

Meaningful function and variable names

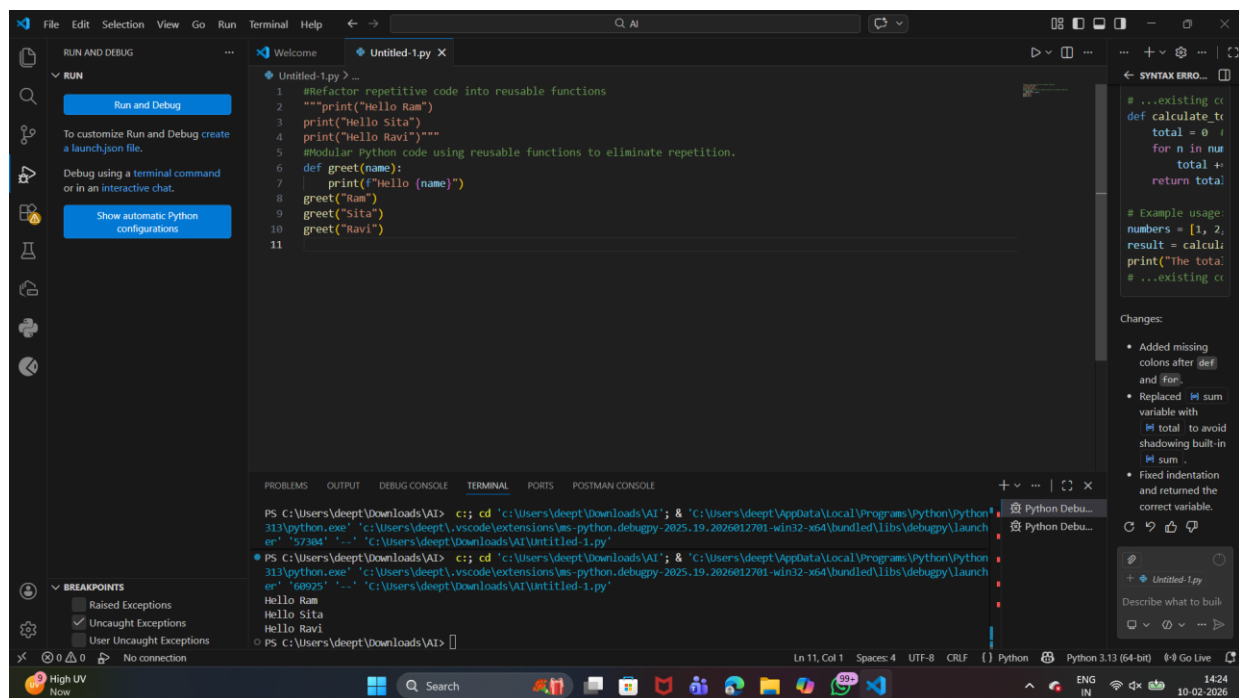
Docstring for clarity

Proper indentation and spacing

Task Description -4(Structural Refactoring)

Prompt: Refactor the repeated print statements into a reusable function.

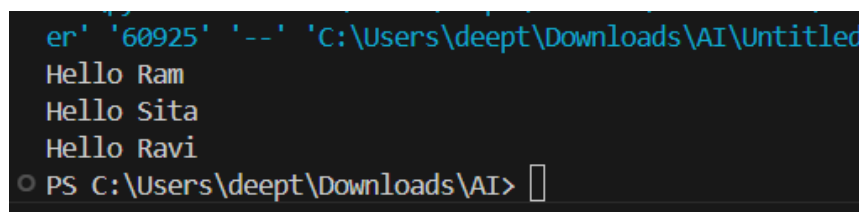
AI-Generated Code:



```
1 #Refactor repetitive code into reusable functions
2 """print("Hello Ram")
3 print("Hello Sita")
4 print("Hello Ravi)"""
5 #Modular Python code using reusable functions to eliminate repetition.
6 def greet(name):
7     print(f"Hello {name}")
8     greet("Ram")
9     greet("Sita")
10    greet("Ravi")
11
```

```
PS C:\Users\deept\Downloads\AI> cd 'c:\Users\deept\Downloads\AI'; & 'C:\Users\deept\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\deept\.vscode\extensions\ms-python.debugpy-2025.19.2026012701-win32-x64\bundle\libs\debugpy\launcher' '57384' '--' 'C:\Users\deept\Downloads\AI\Untitled-1.py'
Hello Ram
Hello Sita
Hello Ravi
PS C:\Users\deept\Downloads\AI>
```

OUTPUT:



```
er' '60925' '--' 'C:\Users\deept\Downloads\AI\Untitled-1.py'
Hello Ram
Hello Sita
Hello Ravi
PS C:\Users\deept\Downloads\AI>
```

Justification:

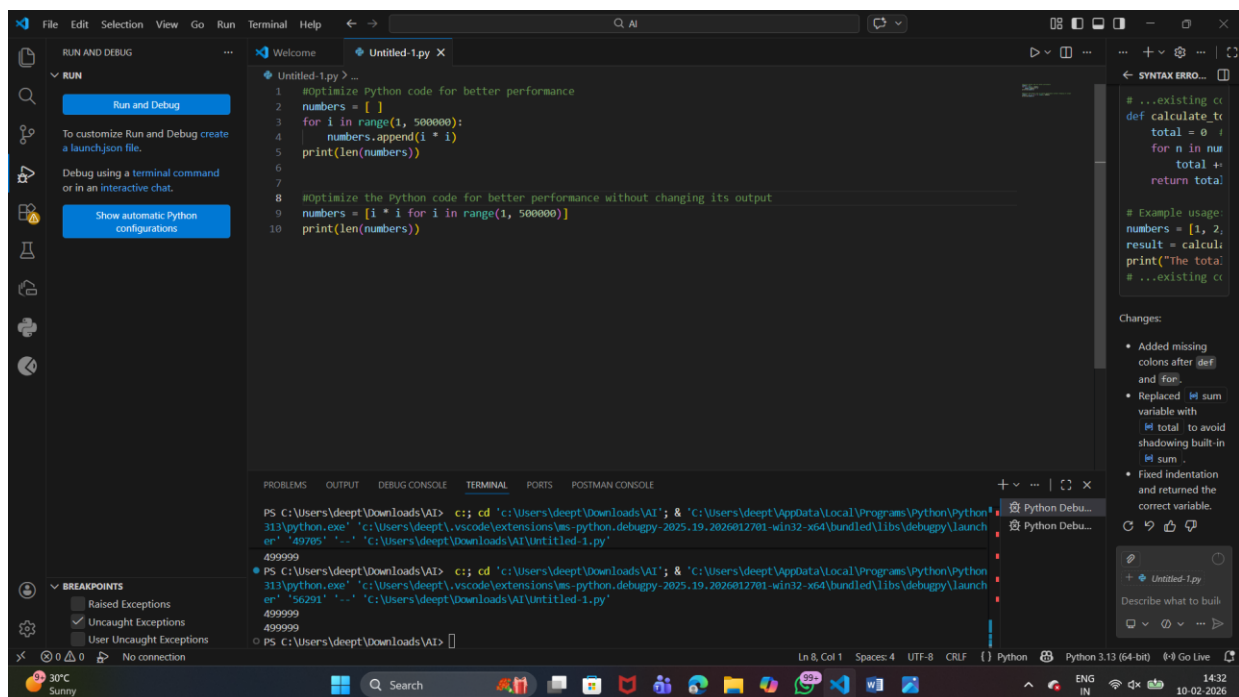
Eliminated repetition

Created reusable function greet()

Task Description -5(Efficiency Enhancement):

Prompt: Optimize the Python code for better performance without changing its output.

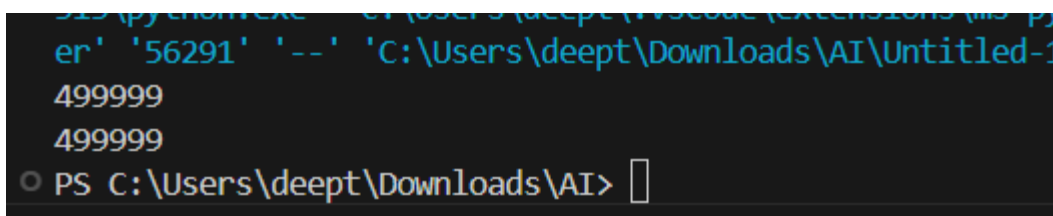
AI-Generated Code:



```
1 #Optimize Python code for better performance
2 numbers = []
3 for i in range(1, 500000):
4     numbers.append(i * i)
5 print(len(numbers))
6
7
8 #Optimize the Python code for better performance without changing its output
9 numbers = [i * i for i in range(1, 500000)]
10 print(len(numbers))
```

```
PS C:\Users\deept\Downloads\AI> cd 'C:\Users\deept\Downloads\AI'; & 'C:\Users\deept\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\deept\.vscode\extensions\ms-python.debugpy-2025.19.2026012701-win32-x64\bundle\libs\debugpy\launcher' '56291' '--' 'C:\Users\deept\Downloads\AI\Untitled-1.py'
499999
PS C:\Users\deept\Downloads\AI>
```

OUTPUT:



```
499999
499999
PS C:\Users\deept\Downloads\AI>
```

Justification:

- Performance Improvements
- Used list comprehension instead of loop
- Faster execution
- Cleaner and more Pythonic code