

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:

The screenshot shows the Visual Studio Code interface with an AI-assisted code editor. The main editor window displays a Python script named `Untitled-1.py`. The script contains a function `calculate_total` that sums up a list of numbers. The AI has provided feedback and corrections, including adding colons at the end of function definitions and loops, and replacing the built-in `sum` variable with a local `total` to avoid shadowing. A sidebar titled "SYNTAX ERROR..." shows the original code with red highlights for errors. The "Changes" sidebar on the right details the modifications made by the AI. The terminal below shows the command to run the script and its output, which is "The total is: 15".

Output:

A terminal window from a Windows operating system shows the execution of the corrected Python script. The command `python Untitled-1.py` is run, and the output is "The total is: 15", indicating the script is working correctly.

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:

The screenshot shows the Visual Studio Code interface with the AI refactoring feature active. The left sidebar has a 'RUN AND DEBUG' section with a 'RUN' dropdown. The main editor window shows a file named 'Untitled-1.py' with the following code:

```
#Refactor Python code to comply with standard coding style guidelines (PEP 8).#Original code:  
1 def findsum(a,b):return a+b  
2 print(findsum(5,10))  
#Refactored code:  
3 def find_sum(a, b):  
4     """Returns the sum of two numbers."""  
5     return a + b  
6 print(find_sum(5, 10))
```

To the right of the editor is a 'SYNTAX ERROR...' panel showing a single error: '# ...existing co'. Below the editor is a 'TERMINAL' tab showing command-line output for running the script:

```
PS C:\Users\deept\Downloads\AI> c:; cd 'c:\Users\deept\Downloads\AI'; & 'c:\Users\deept\AppData\Local\Programs\Python\Python 3.11\python.exe' 'c:\Users\deept\vscode\extensions\ms-python.debugger-2025.19.2026012701-win32-x64\bundled\libs\debugpy\launch er' '65000' '--' 'c:\Users\deept\Downloads\AI\Untitled-1.py'  
The total is: 15  
PS C:\Users\deept\Downloads\AI> c:; cd 'c:\Users\deept\Downloads\AI'; & 'c:\Users\deept\AppData\Local\Programs\Python\Python 3.11\python.exe' 'c:\Users\deept\vscode\extensions\ms-python.debugger-2025.19.2026012701-win32-x64\bundled\libs\debugpy\launch er' '55190' '--' 'c:\Users\deept\Downloads\AI\Untitled-1.py'  
15  
15  
○ PS C:\Users\deept\Downloads\AI>
```

The status bar at the bottom indicates the terminal is using Python 3.13 (64-bit) and the date is 10-02-2026.

OUTPUT:

The terminal window shows the execution of the AI-generated Python code. The output is:

```
er' '55190' '--' 'C:\Users\deept\Downloads\AI\  
15  
15  
○ PS C:\Users\deept\Downloads\AI>
```

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.

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

- Left Sidebar:** RUN AND DEBUG section with RUN button highlighted. It also shows a note about creating a launch.json file and a link to show automatic Python configurations.
- Editor Area:** Untitled-1.py file open. The code is as follows:

```
#Improve code readability without changing its functionality.
def f(x,y):
    return x-y*2
print(f(10,3))

#Rewrite the code with meaningful function and variable names while keeping the logic unchanged.
def calculate_difference(x, y):
    return x - y * 2
result = calculate_difference(10, 3)
print(result)
```
- Right Panel:** SYNTAX ERROR... panel showing an error in the original code. Changes section lists:
 - Added missing colons after def and for.
 - Replaced sum variable with total to avoid shadowing built-in sum.
 - Fixed indentation and returned the correct variable.
- Bottom Panel:** TERMINAL tab showing command-line output. The output shows the command run and the resulting value 4.

OUTPUT:

The terminal output is as follows:

```
PS C:\Users\deept\Downloads\AI> c: cd 'c:\Users\deept\Downloads\AI'; & 'c:\Users\deept\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\deept\vscode\extensions\ms-python.debugger-2025.19.2026012701-win32-x64\bundledlibs\debugpy\launcher' '55190' '--' 'C:\Users\deept\Downloads\AI\Untitled-1.py'
15
15
PS C:\Users\deept\Downloads\AI> c: cd 'c:\Users\deept\Downloads\AI'; & 'c:\Users\deept\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\deept\vscode\extensions\ms-python.debugger-2025.19.2026012701-win32-x64\bundledlibs\debugpy\launcher' '52177' '--' 'C:\Users\deept\Downloads\AI\Untitled-1.py'
4
PS C:\Users\deept\Downloads\AI>
```

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:

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

- Left Sidebar:** RUN AND DEBUG panel with a RUN section containing "Run and Debug" and "Show automatic Python configurations".
- Central Area:** Untitled-1.py editor tab showing the following Python code:

```
1  #Refactor repetitive code into reusable functions
2  """
3      print("Hello Ram")
4      print("Hello Sita")
5      print("Hello Ravi")
6  """
7  def greet(name):
8      print(f"Hello {name}")
9
greet("Ram")
greet("Sita")
greet("Ravi")
```
- Right Panel:** SYNTAX ERRO... panel showing a diff between two versions of calculate_total(). The changes are:
 - Added missing colons after def and for.
 - Replaced sum variable with total to avoid shadowing built-in sum.
 - Fixed indentation and returned the correct variable.
- Bottom:** TERMINAL tab showing command-line output of the script being run, resulting in three "Hello" messages.

OUTPUT:

The terminal output shows the execution of the script and its output:

```
PS C:\Users\deepthi\Downloads\AI> python Untitled-1.py
Hello Ram
Hello Sita
Hello Ravi
PS C:\Users\deepthi\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:

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

- File Explorer:** Shows a folder structure including .gitignore, .vscode, and a file named 'Untitled-1.py'.
- Run and Debug sidebar:** Contains buttons for 'Run and Debug' (highlighted), 'Show automatic Python configurations', and links to 'Customize Run and Debug' and 'Create a launch.json file'.
- Code Editor:** Displays Python code for generating a list of squares and calculating their sum, with annotations for optimization and syntax errors.
- Terminal:** Shows command-line output for launching Python and running the script.
- Output:** Shows log messages from the Python process.
- Breakpoints:** Shows a list of breakpoints, including 'Raised Exceptions' and 'Unc caught Exceptions'.
- Syntactic Error:** A sidebar showing syntax errors in the code.
- Changes:** A sidebar showing code changes with annotations.

OUTPUT:

```
313 (pythontest) - C:\Users\deept\Downloads\AI> python -m pip install --upgrade pip  
deept@DEPT-56291 MINGW64 ~ % C:\Users\deept\Downloads\AI\Untitled-1  
499999  
499999  
○ PS C:\Users\deept\Downloads\AI> 
```

Justification:

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