

ASSIGNMENT -10.2

NAME: HABEEBA KHANAM

HT.NO:2303A51474

BT.NO:29

Lab 10 – Code Review and Quality: Using AI to Improve Code Quality and Readability

TASK DESCRIPTION -1 (ERROR DETECTION AND CORRECTION)

Task:

Use AI to analyze a Python script and correct all syntax and logical errors.

Sample Input Code:

```
def calculate_total(nums)
    sum = 0
    for n in nums
        sum += n
    return total
```

Expected Output-1:

Corrected and executable Python code with brief explanations of the identified syntax and logic errors.

CODE:

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. The editor tab bar at the top includes File, Edit, Selection, View, Go, Run, and other icons. The main area displays a Python script named '9876543.py' with the following code:

```
def calculate_total(nums):
    sum = 0
    for n in nums:
        sum += n
    return sum

# Example usage
numbers = [1, 2, 3, 4, 5]
total = calculate_total(numbers)
print("The total is:", total)
```

The terminal tab shows the output of running the script:

```
PS C:\Users\Sameera Khan> python 9876543.py
The total is: 15
```

The status bar at the bottom right indicates Spaces: 4, UTF-8, Python 3.13.5, and Go Live.

Errors Identified and Explanation

- Missing Colon (:)
- Indentation Errors
- Missing Colon in for Loop
- Variable Name Mismatch (Logical Error)
- Avoid Using Reserved Name sum

TASK DESCRIPTION -2(CODE STYLED ANDARDIZATION)

Task:

Use AI to refactor Python code to comply with standard coding style guidelines.

Sample Input Code:

```
def findSum(a,b):return a+b
print(findSum(5,10))
```

Expected Output-2:

Well-structured, consistently formatted Python code following standard style conventions.

CODE:

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

- File Explorer:** Shows multiple files including task2.py, Untitled-2, task 2344.py, ujhtt.py, bygvhikthy.py, lab - 9.5 Al.py, and 9876543.py.
- Editor:** The 9876543.py file is open, displaying the following code:

```
def find_sum(num1, num2):
    """
    Returns the sum of two numbers.
    """
    return num1 + num2

def main():
    result = find_sum(5, 10)
    print("Sum:", result)

if __name__ == "__main__":
    main()
```
- Terminal:** Shows command-line output from Python executing the script:

```
PS C:\Users\Sameera Khan> python 9876543.py
The total is: 15
PS C:\Users\Sameera Khan> python 9876543.py
Sum: 15
PS C:\Users\Sameera Khan>
```
- Bottom Status Bar:** Shows file path (C:\Users\Sameera Khan\OneDrive\Desktop\9876543.py), line 14, column 11, spaces 4, encoding UTF-8, CRLF, Python 3.13.5, and Go Live button.

Improvements Made According to Coding Standards

- Function Naming (PEP 8)
- Proper Spacing
- Meaningful Variable Names
- Structured Program Flow
- Added Docstring

TASK DESCRIPTION -3(CODE CLARITY IMPROVEMENT)

Task:

Use AI to improve code readability without changing its functionality.

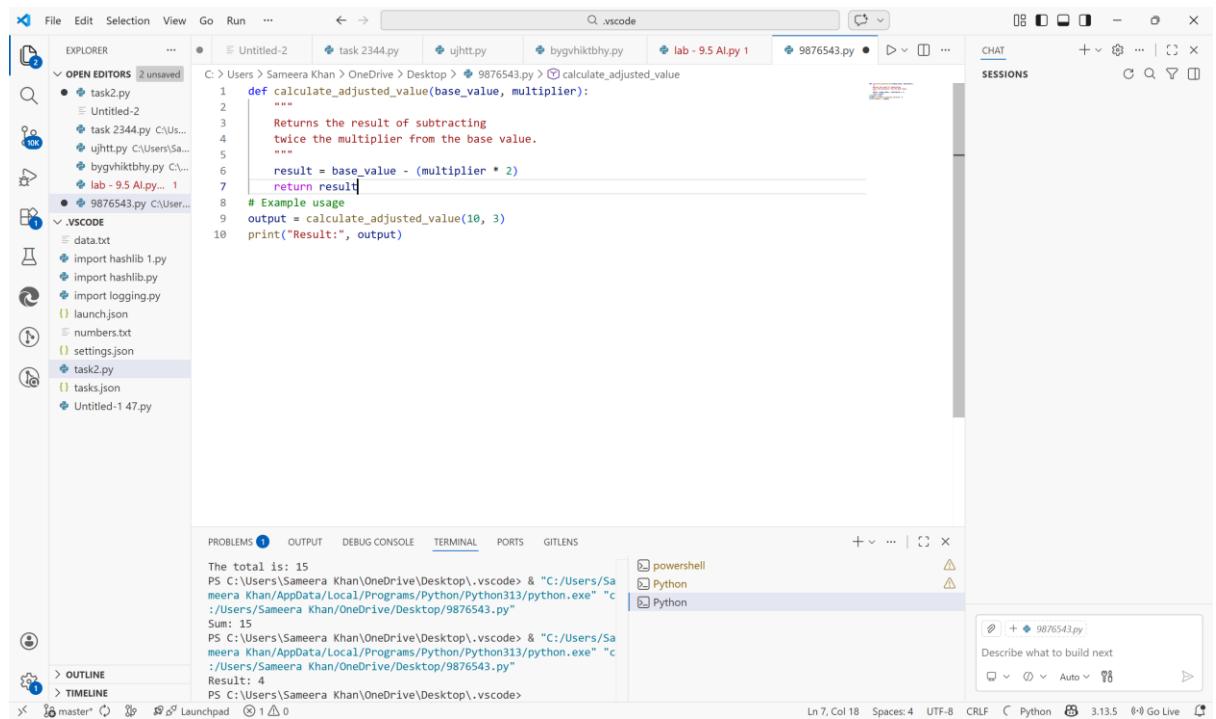
Sample Input Code:

```
def f(x,y):  
    return x-y**2  
print(f(10,3))
```

Expected Output-3:

Python code rewritten with meaningful function and variable names,
proper indentation, and improved clarity.

CODE:



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

- EXPLORER View:** Shows files like task2.py, task 2344.py, ujhtt.py, bygvhiktbhy.py, lab - 9.5 Al.py, and 9876543.py.
- OPEN EDITOR View:** The file 9876543.py is open, displaying the following Python code:

```
def calculate_adjusted_value(base_value, multiplier):  
    """  
    Returns the result of subtracting  
    twice the multiplier from the base value.  
    """  
    result = base_value - (multiplier * 2)  
    return result  
  
# Example usage  
output = calculate_adjusted_value(10, 3)  
print("Result:", output)
```
- TERMINAL View:** Shows the command line output:

```
The total is: 15  
PS C:\Users\Sameera Khan\OneDrive\Desktop> & "C:/Users/Sameera Khan/AppData/Local/Programs/Python/Python313/python.exe" "c:/Users/Sameera Khan/OneDrive/Desktop/9876543.py"  
Sum: 15  
PS C:\Users\Sameera Khan\OneDrive\Desktop> & "C:/Users/Sameera Khan/AppData/Local/Programs/Python/Python313/python.exe" "c:/Users/Sameera Khan/OneDrive/Desktop/9876543.py"  
Result: 4  
PS C:\Users\Sameera Khan\OneDrive\Desktop>
```
- STATUS BAR:** Shows the current file is 9876543.py, with 3.13.5 version information.

Improvements Made:

- Meaningful Function Name
- Descriptive Variable Names
- Proper Indentation & Formatting
- Added Parentheses for Clarity
- base_value - (multiplier * 2)
- Added Docstring

TASK DESCRIPTION -4(STRUCTURAL REFACTORING)

Task:

Use AI to refactor repetitive code into reusable functions.

Sample Input Code:

```
print("Hello Ram")
print("Hello Sita")
print("Hello Ravi")
```

Expected Output-4:

Modular Python code using reusable functions to eliminate repetition.

CODE:

```

OPEN EDITORS 1 unsaved
task2.py
Untitled-2
task 2344.py
ujhtt.py
byghikthby.py
lab - 9.5 Al.py 1
9876543.py > ...
File Edit Selection View Go Run ... < > Q .vscode
EXPLORER
VS CODE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
meera Khan/AppData/Local/Programs/Python/Python313/python.exe" "c :/Users/Sameera Khan/OneDrive/Desktop/9876543.py"
Result: 4
PS C:\Users\Sameera Khan\OneDrive\Desktop\vscode> & "C:/Users/Sameera Khan/AppData/Local/Programs/Python/Python313/python.exe" "c :/Users/Sameera Khan/OneDrive/Desktop/9876543.py"
Hello Ram
Hello Sita
Hello Ravi
PS C:\Users\Sameera Khan\OneDrive\Desktop\.vscode>
Ln 13, Col 25 Spaces: 4 UTF-8 CRLF { } Python 3.13.5 ⓘ Go Live

```

Improvements Made:

- Eliminated Repetition
- Improved Maintainability
- Used Loop for Scalability
- Modular Programming Approach

TASK DESCRIPTION -5(EFFICIENCY ENHANCEMENT)

Task:

Use AI to optimize Python code for better performance.

Sample Input Code:

numbers = []

```

for i in range(1, 500000):
    numbers.append(i * i)
    print(len(numbers))

```

Expected Output-5:

Optimized Python code that achieves the same result with improved performance.

CODE:

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists several files: task2.py, Untitled-2, task 2344.py, ujhtt.py, bygvhiktbhy.py, lab - 9.5 AI.py, 9876543.py (the active file), data.txt, .VS CODE, import hashlib 1.py, import hashlib.py, import logging.py, launch.json, numbers.txt, settings.json, task2.py, tasks.json, and Untitled-1 47.py. The terminal at the bottom shows the execution of the code and its output:

```

meera Khan/AppData/Local/Programs/Python/Python313/python.exe" "c
:Users/Sameera Khan/OneDrive/Desktop/9876543.py"
Hello Ram
Hello Sita
Hello Ravi
PS C:\Users\Sameera Khan\OneDrive\Desktop\vscode> & "C:/Users/Sameera Khan/AppData/Local/Programs/Python/Python313/python.exe" "c
:Users/Sameera Khan/OneDrive/Desktop/9876543.py"
49999
PS C:\Users\Sameera Khan\OneDrive\Desktop\vscode>

```

Issues in the Original Code:

- Uses a manual for loop with .append()
- Slower for large datasets
- Not the most Pythonic approach

