

**School of Computer Science and Artificial Intelligence****Lab Assignment # 10.2**

---

|                         |                             |
|-------------------------|-----------------------------|
| <b>Program</b>          | <b>: B. Tech (CSE)</b>      |
| <b>Specialization</b>   | <b>: -</b>                  |
| <b>Course Title</b>     | <b>: AI Assisted Coding</b> |
| <b>Course Code</b>      | <b>: 23CS002PC304</b>       |
| <b>Semester</b>         | <b>II</b>                   |
| <b>Academic Session</b> | <b>: 2025-2026</b>          |
| <b>Name of Student</b>  | <b>: P.Eshwar</b>           |
| <b>Enrollment No.</b>   | <b>: 2403A51L26</b>         |
| <b>Batch No.</b>        | <b>51</b>                   |
| <b>Date</b>             | <b>: 06/02/26</b>           |

---

**Submission Starts here****Screenshots:****Task Description -1 Task:**(Error Detection and Correction)**Prompt:****Use AI to analyze a Python script and correct all syntax and logical errors.****Sample Input Code:**

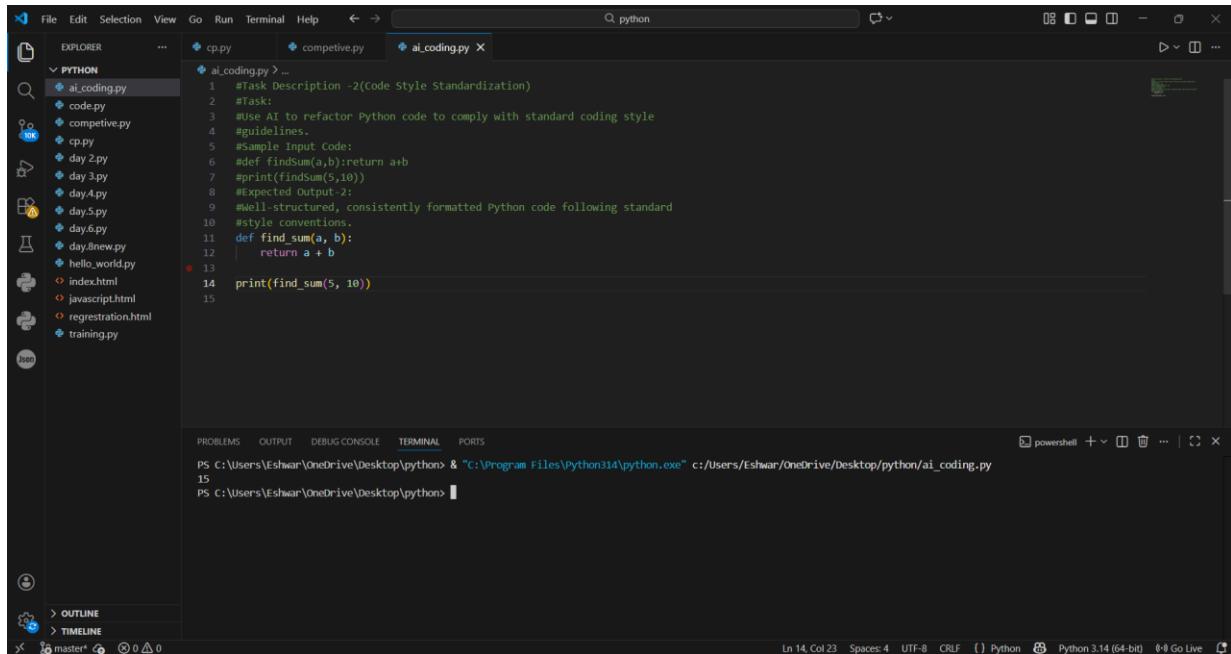
```
def calculate_total(nums)
```

```
    sum = 0
```

```
    for n in nums
```

```
        sum += n
```

```
    return total
```



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists files in a 'PYTHON' folder, including 'ai\_coding.py', 'code.py', 'competitive.py', 'cp.py', 'day 2.py', 'day 3.py', 'day 4.py', 'day 5.py', 'day 6.py', 'day8new.py', 'hello\_world.py', 'index.html', 'javascript.html', 'registration.html', and 'training.py'. The 'ai\_coding.py' file is selected and shown in the main editor area. The code is as follows:

```

1 #Task Description -2(Code Style Standardization)
2 #Task:
3 #use AI to refactor Python code to comply with standard coding style
4 #guidelines.
5 #sample Input Code:
6 #def findSum(a,b):return a+b
7 #print(findSum(5,10))
8 #Expected Output -2:
9 #Well-structured, consistently formatted Python code following standard
10 #style conventions.
11 def find_sum(a, b):
12     return a + b
13
14 print(find_sum(5, 10))
15

```

The Terminal pane at the bottom shows the command line output:

```

PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
15
PS C:\Users\Eshwar\OneDrive\Desktop\python>

```

At the bottom right, status bar details include: Ln 14, Col 23, Spaces: 4, UTF-8, CRLF, Python 3.14 (64-bit), Go Live.

## Task Description -2(Code Style Standardization)

**Prompt:**

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

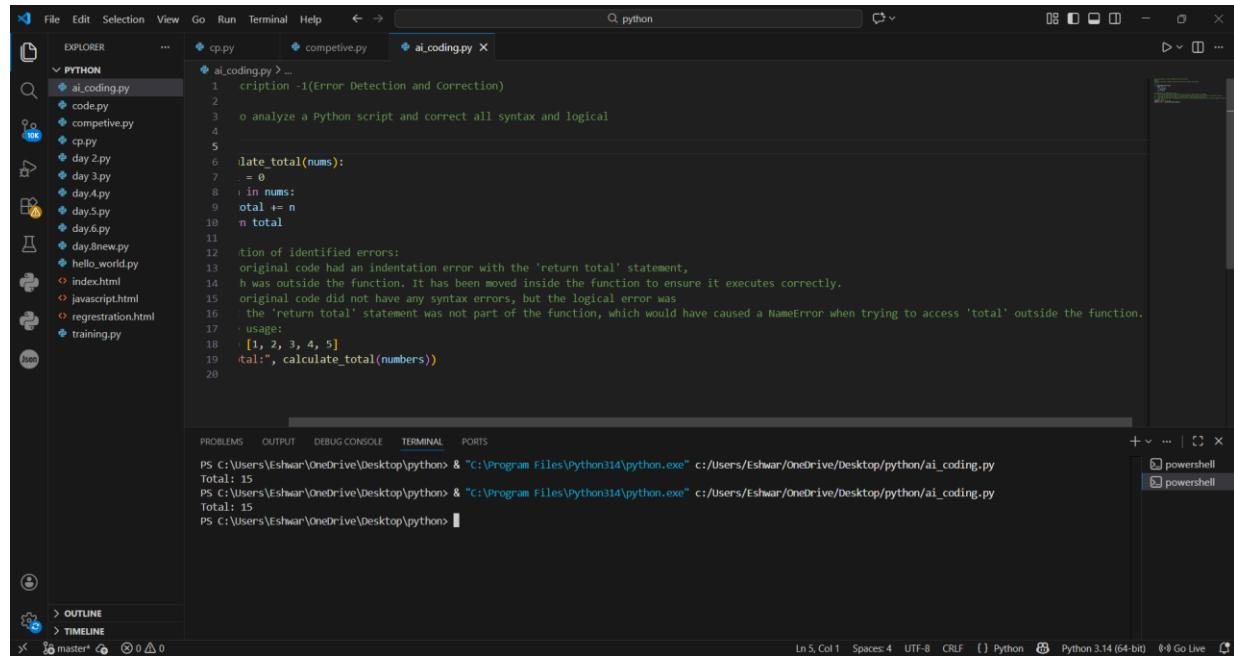
**Sample Input Code:**

```

def findSum(a,b):return a+b

print(findSum(5,10))

```



The screenshot shows the Visual Studio Code interface with the Python extension installed. The Explorer sidebar on the left lists various Python files and HTML documents. The main editor window displays a Python script named 'ai\_coding.py'. A code editor overlay from the AI Coding Assistant provides feedback on line 12, which contains an indentation error. The terminal at the bottom shows the command 'python ai\_coding.py' being run, and the output indicates a syntax error due to the logical error in the original code.

```

1  #!/usr/bin/python
2  # This script analyzes a Python script and corrects all syntax and logical
3  # errors.
4
5  def calculate_total(nums):
6      total = 0
7      for num in nums:
8          total += num
9
10     return total
11
12     print("The function has errors!")
13     print("The original code had an indentation error with the 'return total' statement,")
14     print("it was outside the function. It has been moved inside the function to ensure it executes correctly.")
15     print("The original code did not have any syntax errors, but the logical error was")
16     print("the 'return total' statement was not part of the function, which would have caused a NameError when trying to access 'total' outside the function.")
17
18 if __name__ == "__main__":
19     calculate_total([1, 2, 3, 4, 5])
20

```

## Task Description -3(Code Clarity Improvement)

### Prompt:

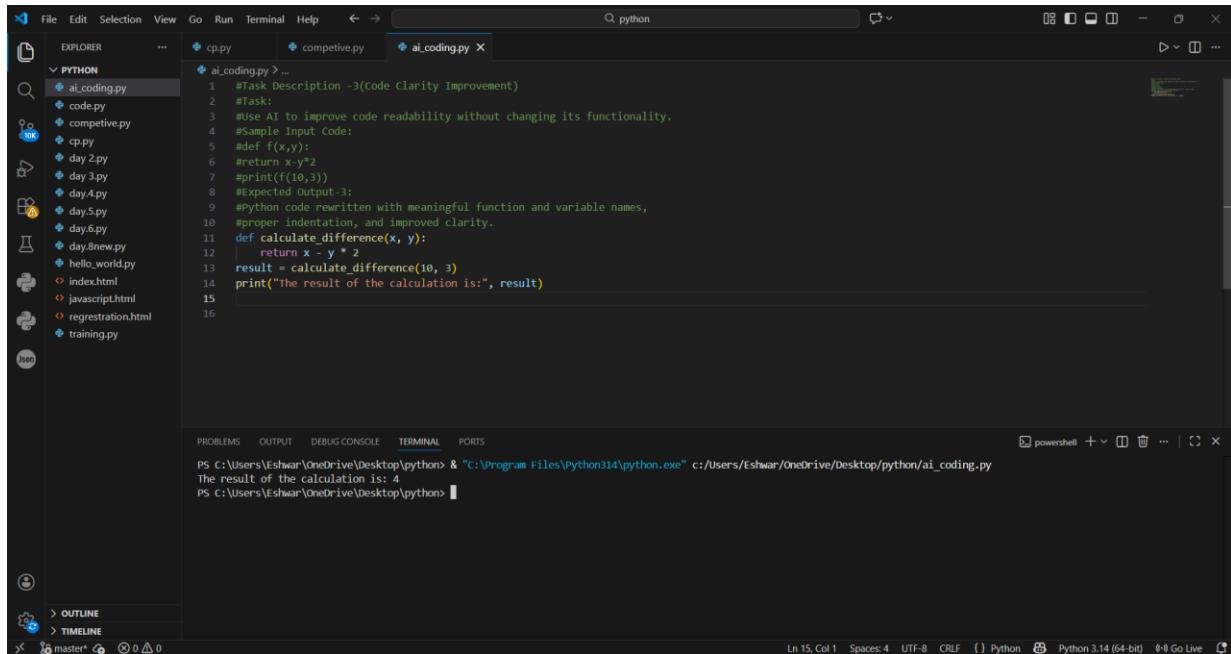
Use AI to improve code readability without changing its functionality.

### Sample Input Code:

```
def f(x,y):
```

```
    return x-y**2
```

```
print(f(10,3))
```



```

1 #Task Description -3(Code Clarity Improvement)
2 #Task:
3 #use AI to improve code readability without changing its functionality.
4 #sample input Code:
5 def f(x,y):
6     #return x-y**2
7     #print(f(10,3))
8     #Expected Output-3:
9     #python code rewritten with meaningful function and variable names,
10    #proper indentation, and improved clarity.
11    def calculate_difference(x, y):
12        return x - y ** 2
13    result = calculate_difference(10, 3)
14    print("The result of the calculation is:", result)
15
16

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai\_coding.py  
The result of the calculation is: 4  
PS C:\Users\Eshwar\OneDrive\Desktop\python>

Ln 15, Col 1 Spaces: 4 UTR-8 CRLF {} Python Python 3.14 (64-bit) Go Live

## Task Description -4(Structural Refactoring)

**prompt:**

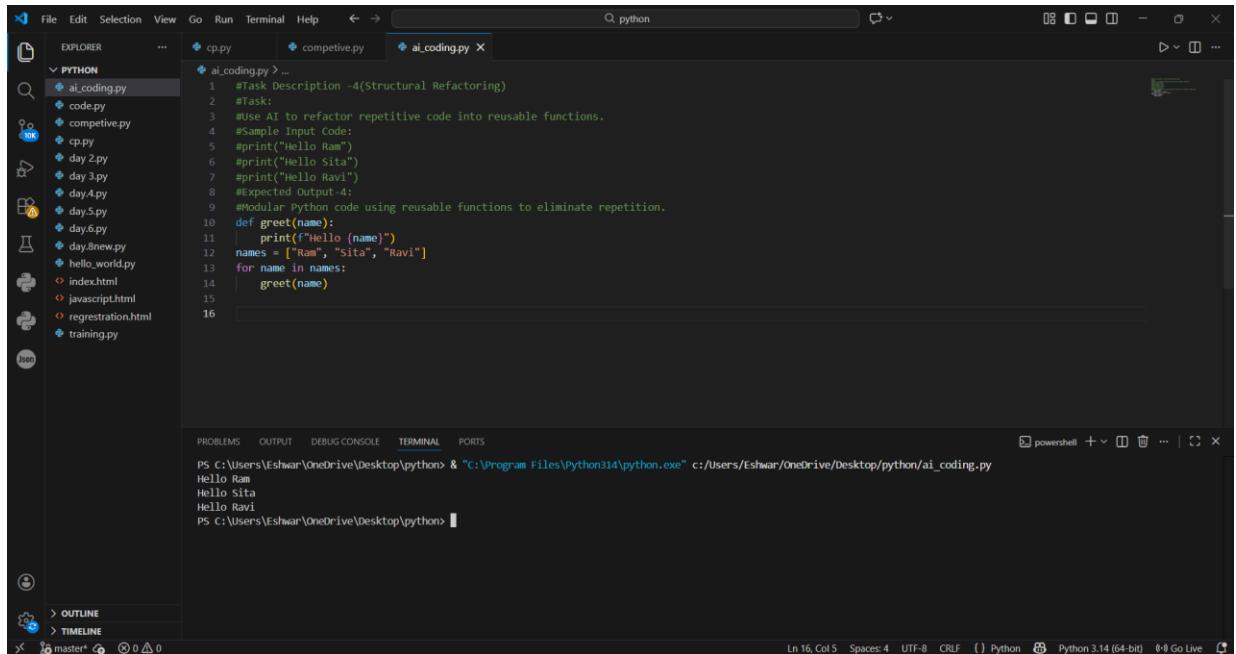
Use AI to refactor repetitive code into reusable functions.

**Sample Input Code:**

```
print("Hello Ram")
```

```
print("Hello Sita")
```

```
print("Hello Ravi")
```



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists files including 'ai\_coding.py', 'code.py', 'competitive.py', 'cp.py', 'day 2.py', 'day 3.py', 'day 4.py', 'day 5.py', 'day 6.py', 'day8new.py', 'hello\_world.py', 'index.html', 'javascript.html', 'registration.html', and 'training.py'. The 'ai\_coding.py' file is open in the main editor area, containing the following Python code:

```

1 #Task Description -4(Structural Refactoring)
2 #Task:
3 #use AI to refactor repetitive code into reusable functions.
4 #sample input code:
5 #print("Hello Ram")
6 #print("Hello Sita")
7 #print("Hello Ravi")
8 #Expected Output-4:
9 #Modular Python code using reusable functions to eliminate repetition.
10 def greet(name):
11     print("Hello " + name)
12 names = ["Ram", "Sita", "Ravi"]
13 for name in names:
14     greet(name)
15
16

```

The terminal at the bottom shows the command 'python ai\_coding.py' being run, with the output 'Hello Ram', 'Hello Sita', and 'Hello Ravi' displayed.

## Task Description -5(Efficiency Enhancement)

### Prompt:

Use AI to optimize Python code for better performance.

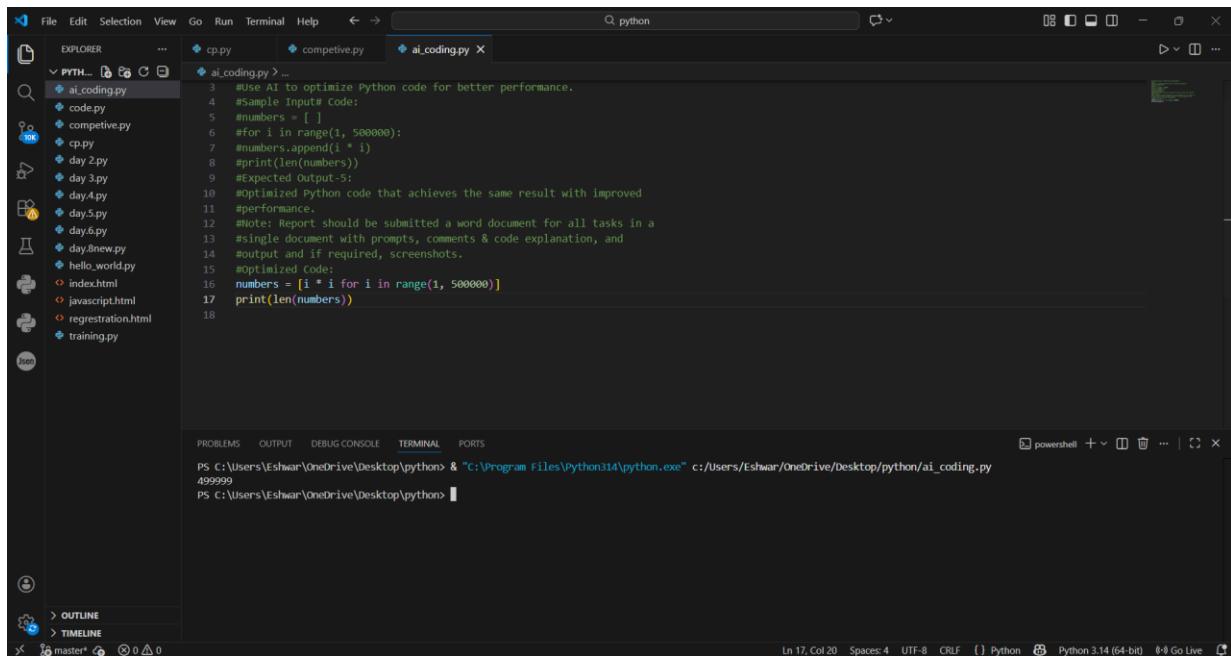
### Sample Input Code:

```
numbers = [ ]
```

```
for i in range(1, 500000):
```

```
    numbers.append(i * i)
```

```
print(len(numbers))
```



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar displays a file tree under 'PYTH...' with files like ai\_coding.py, cp.py, competitive.py, day 2.py, day 3.py, day 4.py, day 5.py, day 6.py, day 8new.py, hello\_world.py, index.html, javascript.html, registration.html, and training.py. The main editor area contains the following Python code:

```
#use AI to optimize Python code for better performance.
#Sample Input# Code:
numbers = []
for i in range(1, 500000):
    numbers.append(i * 5)
#print(len(numbers))
#Expected Output-5:
#Optimized Python code that achieves the same result with improved
#performance.
#Note: Report should be submitted a word document for all tasks in a
single document with prompts, comments & code explanation, and
#output and if required, screenshots.
#Optimized Code:
numbers = [i * i for i in range(1, 500000)]
print(len(numbers))
```

The terminal at the bottom shows the command being run and its output:

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
499999
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

VS Code status bar details: Ln 17, Col 20, Spaces: 4, UTF-8, CRLF, Python 3.14 (64-bit), Go Live.