

ASSIGNMENT -1.5

NAME : AKSHITHA

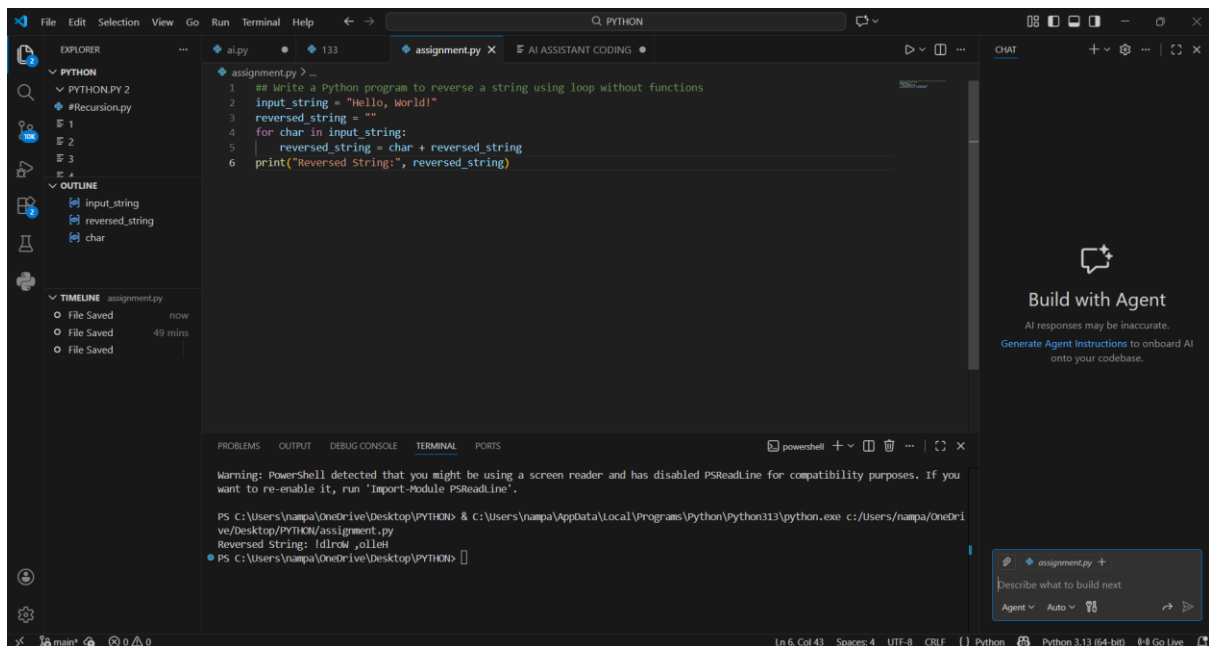
2303A51360

BATCH:29

TASK 1:

PROMPT: AI-GENERATED LOGIC WITHOUT MODULARIZATION
(STRING REVERSAL WITHOUT FUNCTIONS)

CODE:



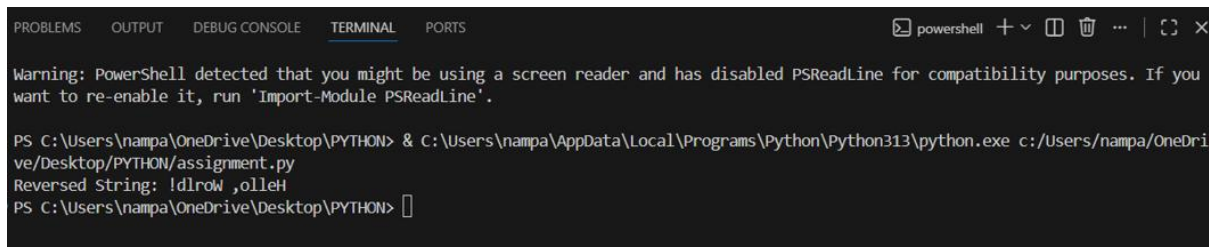
The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project named 'PYTHON' with a file 'assignment.py'. The Outline pane shows variables 'input_string', 'reversed_string', and 'char'. The main editor displays the following Python code:

```
1  ## write a Python program to reverse a string using loop without functions
2  input_string = "Hello, World!"
3  reversed_string = ""
4  for char in input_string:
5      reversed_string = char + reversed_string
6  print("Reversed String:", reversed_string)
```

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

```
PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:/Users/nampa/OneDrive/Desktop/PYTHON/assignment.py
Reversed String: !dlrow ,olleH
PS C:\Users\nampa\OneDrive\Desktop\PYTHON>
```

OUTPUT:



The screenshot shows a terminal window with the following output:

```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:/Users/nampa/OneDrive/Desktop/PYTHON/assignment.py
Reversed String: !dlrow ,olleH
PS C:\Users\nampa\OneDrive\Desktop\PYTHON>
```

OBSERVATION:

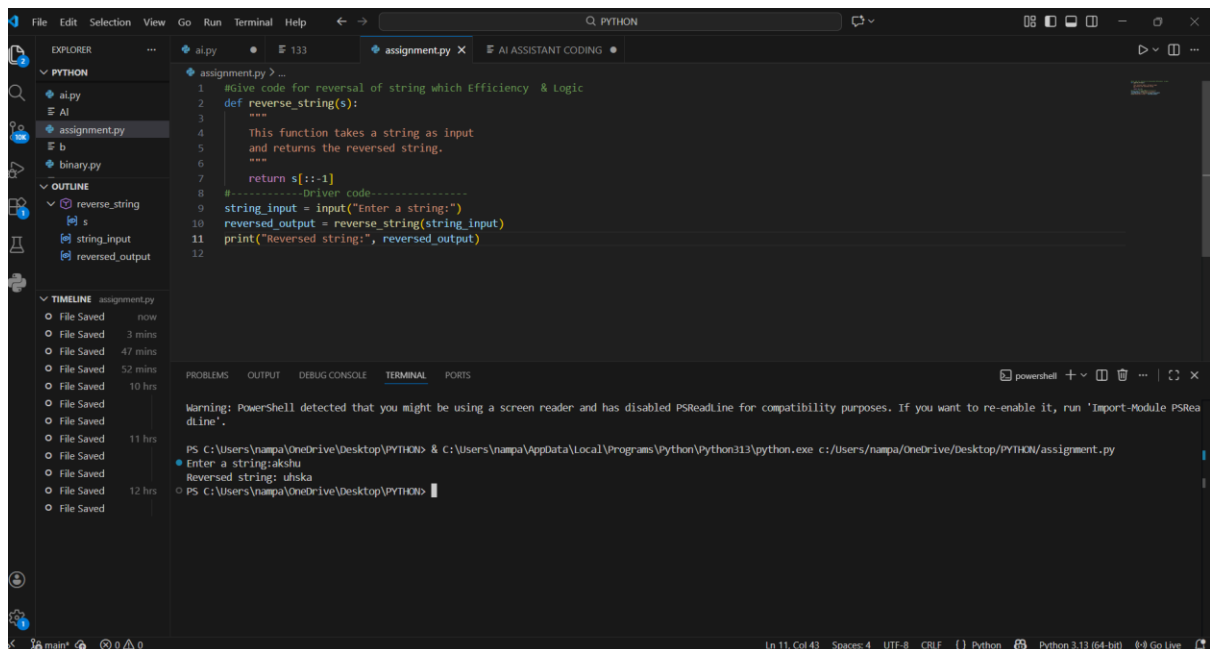
The program successfully reverses the given string using a simple loop without using any functions. Each character is added to the beginning of a new string, which gradually forms the reversed output. The output confirms that the logic works correctly by displaying the reversed string. This approach is easy to understand and suitable for beginners learning basic string operations.

However, for larger programs, a more optimized or modular approach would be better.

TASK:2

PROMPT: GIVE CODE FOR REVERSAL OF STRING WHICH EFFICIENCY & LOGIC OPTIMIZATION

CODE:



```
1 #Give code for reversal of string which Efficiency & Logic
2 def reverse_string(s):
3     """
4     This function takes a string as input
5     and returns the reversed string.
6     """
7     return s[::-1]
8 #-----Driver code-----
9 string_input = input("Enter a string:")
10 reversed_output = reverse_string(string_input)
11 print("Reversed string:", reversed_output)
12
```

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

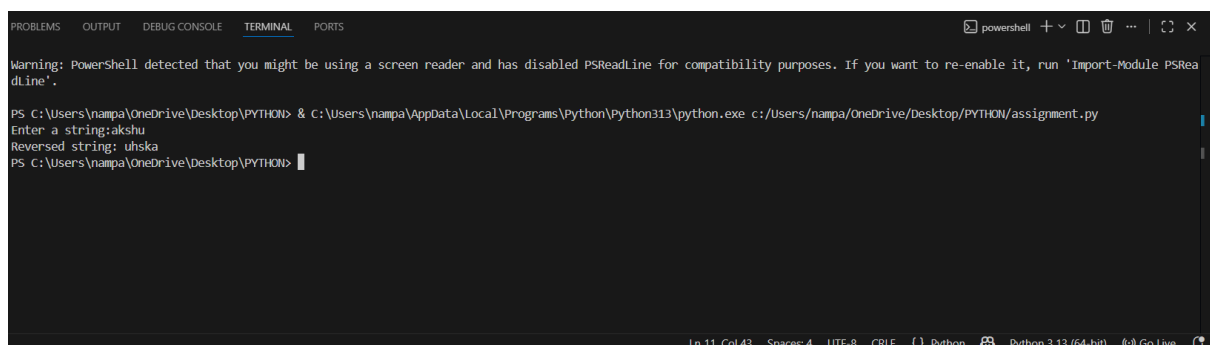
PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:/Users/nampa/OneDrive/Desktop/PYTHON/assignment.py

Enter a string:akshu

Reversed string: uhksa

PS C:\Users\nampa\OneDrive\Desktop\PYTHON>

OUTPUT:



```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.
```

PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:/Users/nampa/OneDrive/Desktop/PYTHON/assignment.py

Enter a string:akshu

Reversed string: uhksa

PS C:\Users\nampa\OneDrive\Desktop\PYTHON>

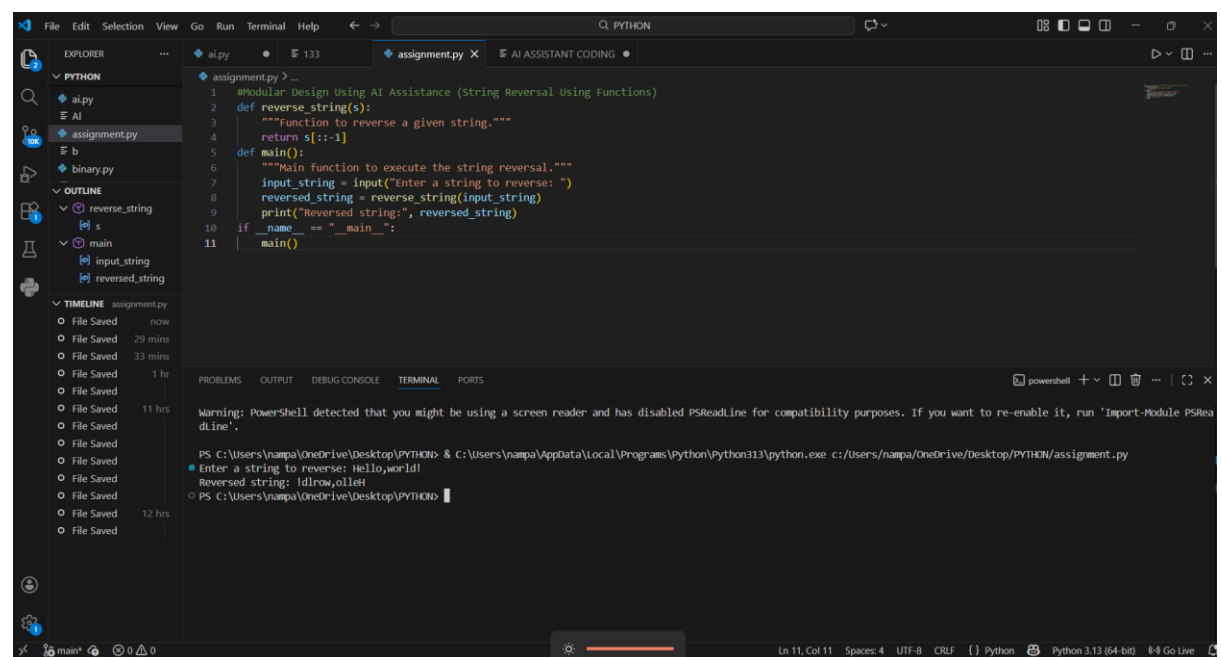
OBSERVATION:

The function uses Python slicing to reverse the string in a single step. No extra variables or loops are used, which makes the code easy to read. The logic is efficient and executes faster than manual reversal methods. Overall, the code is clean, readable, and suitable for review by other developers. The optimized approach reduces unnecessary operations and improves performance. It follows Python best practices, making the code more maintainable and reliable.

TASK:3

PROMPT: MODULAR DESIGN USING AI ASSISTANCE (STRING REVERSAL USING FUNCTIONS)

CODE:



The screenshot shows the Visual Studio Code editor with a Python file named `assignment.py`. The code implements a modular design for string reversal using functions. The `reverse_string(s)` function uses slicing to reverse the string. The `main()` function prompts the user for input, calls the reversal function, and prints the result. The terminal window shows the execution of the script, displaying the warning about PSReadLine and the successful reversal of the input string "Hello,world!" to "ldrow,olleH".

```
1 #Modular Design Using AI Assistance (String Reversal Using Functions)
2 def reverse_string(s):
3     """Function to reverse a given string."""
4     return s[::-1]
5
6 def main():
7     """Main function to execute the string reversal."""
8     input_string = input("Enter a string to reverse: ")
9     reversed_string = reverse_string(input_string)
10    print("Reversed string:", reversed_string)
11
12 if __name__ == "__main__":
13     main()
```

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadline for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadline'.

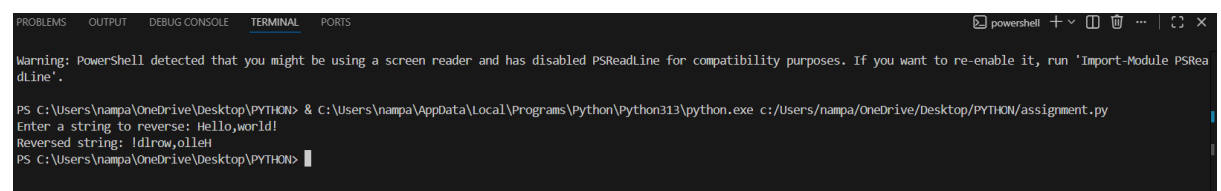
PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:\Users\nampa\OneDrive\Desktop\PYTHON\assignment.py

Enter a string to reverse: Hello,world!

Reversed string: ldrow,olleH

PS C:\Users\nampa\OneDrive\Desktop\PYTHON>

OUTPUT:



The terminal output shows the execution of the script, displaying the warning about PSReadline and the successful reversal of the input string "Hello,world!" to "ldrow,olleH".

```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadline for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadline'.
```

```
PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:\Users\nampa\OneDrive\Desktop\PYTHON\assignment.py
```

```
Enter a string to reverse: Hello,world!
```

```
Reversed string: ldrow,olleH
```

```
PS C:\Users\nampa\OneDrive\Desktop\PYTHON>
```

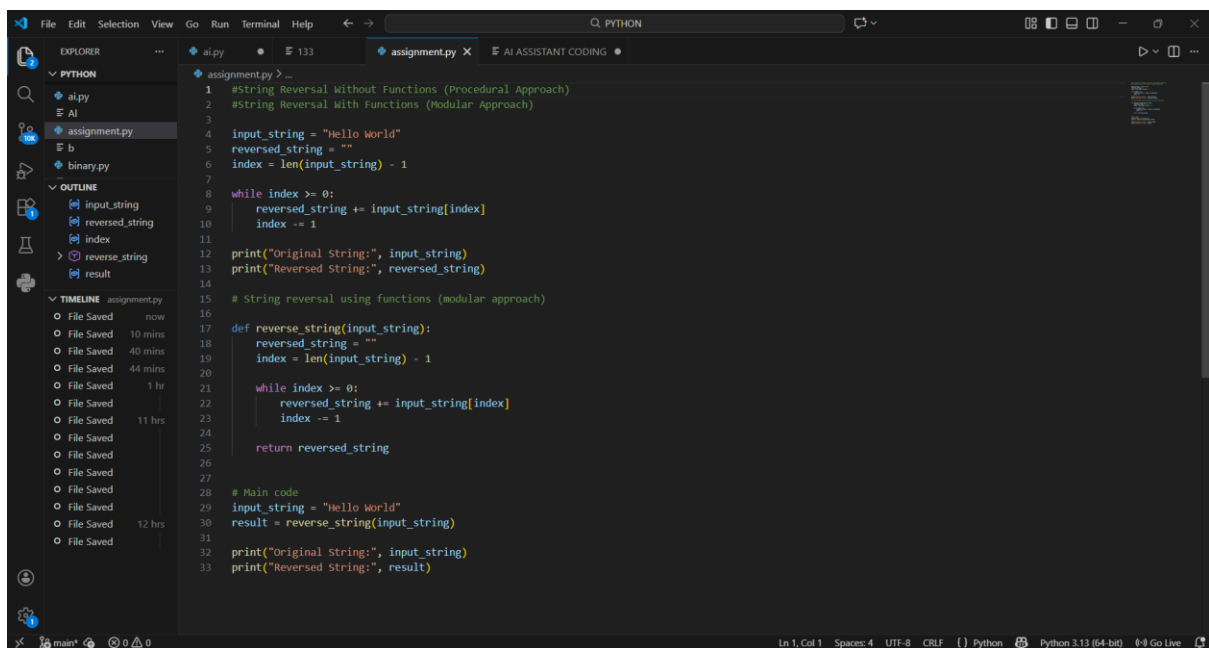
OBSERVATION:

This program follows a modular design by separating the string reversal logic into a reusable function. The use of clear function names and meaningful comments makes the code easy to understand and maintain. Since the reversal logic is written only once, it can be reused in multiple parts of the application without duplication. Overall, the structure improves readability, reusability, and makes future modifications simple.

TASK:4

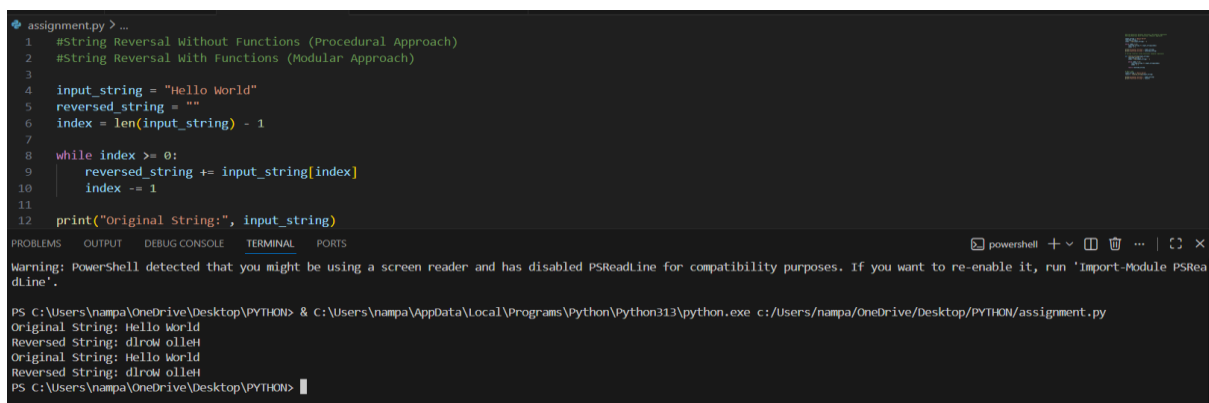
PROMPT: COMPARATIVE ANALYSIS – PROCEDURAL VS MODULAR APPROACH (WITH VS WITHOUT FUNCTIONS)

CODE:



```
1 #String Reversal Without Functions (Procedural Approach)
2 #String Reversal With Functions (Modular Approach)
3
4 input_string = "Hello World"
5 reversed_string = ""
6 index = len(input_string) - 1
7
8 while index >= 0:
9     reversed_string += input_string[index]
10    index -= 1
11
12 print("Original String:", input_string)
13 print("Reversed String:", reversed_string)
14
15 # String reversal using functions (modular approach)
16
17 def reverse_string(input_string):
18     reversed_string = ""
19     index = len(input_string) - 1
20
21     while index >= 0:
22         reversed_string += input_string[index]
23         index -= 1
24
25     return reversed_string
26
27
28 # Main code
29 input_string = "Hello World"
30 result = reverse_string(input_string)
31
32 print("Original String:", input_string)
33 print("Reversed String:", result)
```

OUTPUT:



```
PS C:\Users\nampa\OneDrive\Desktop\PYTHON> C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:/Users/nampa/OneDrive/Desktop/PYTHON/assignment.py
Original String: Hello World
Reversed String: dlrow olleH
Original String: Hello World
Reversed String: dlrow olleH
PS C:\Users\nampa\OneDrive\Desktop\PYTHON>
```

OBSERVATION:

The procedural approach places all logic in one block, making the code harder to reuse and maintain. The modular approach separates logic into a function, improving clarity and structure. Functions allow easy reuse of code without duplication.

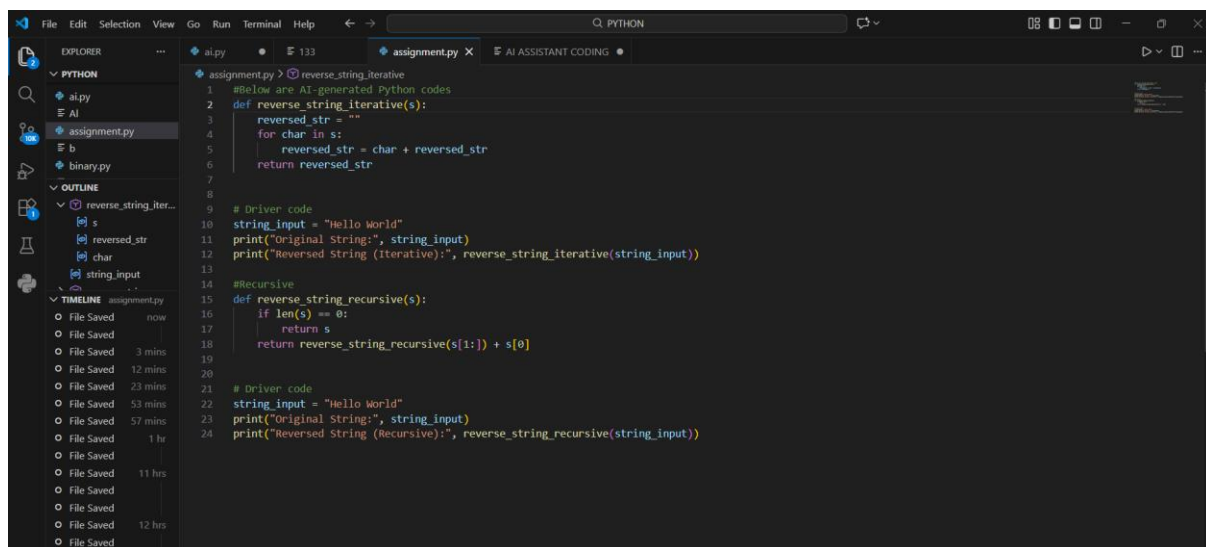
Debugging is simpler in the modular approach because errors can be isolated.

Overall, modular design is better suited for large and scalable applications.

TASK:5

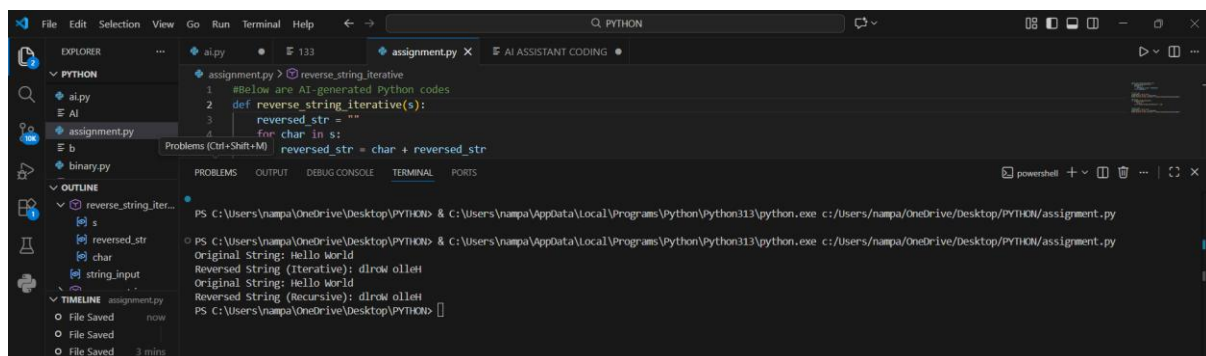
PROMPT:AI-GENERATED PYTHON CODES ITERATIVE VS RECURSION

CODE:



```
1 #Below are AI-generated Python codes
2 def reverse_string_iterative(s):
3     reversed_str = ""
4     for char in s:
5         reversed_str = char + reversed_str
6     return reversed_str
7
8
9 # Driver code
10 string_input = "Hello World"
11 print("Original String:", string_input)
12 print("Reversed String (Iterative):", reverse_string_iterative(string_input))
13
14 #Recursive
15 def reverse_string_recursive(s):
16     if len(s) == 0:
17         return s
18     return reverse_string_recursive(s[1:]) + s[0]
19
20
21 # Driver code
22 string_input = "Hello World"
23 print("Original String:", string_input)
24 print("Reversed String (Recursive):", reverse_string_recursive(string_input))
```

OUTPUT:



```
PS C:\Users\nampa\OneDrive\Desktop\PYTHON> & C:\Users\nampa\AppData\Local\Programs\Python\Python313\python.exe c:\Users\nampa\OneDrive\Desktop\PYTHON\assignment.py
Original String: Hello World
Reversed String (Iterative): dlrow olleH
Original String: Hello World
Reversed String (Recursive): dlrow olleH
PS C:\Users\nampa\OneDrive\Desktop\PYTHON>
```

OBSERVATION:

The iterative approach reverses the string by looping through each character, which makes the execution flow easy to follow but slightly slower due to repeated string concatenation. The recursive approach breaks the problem into smaller parts, which is conceptually clean but uses more memory because of function calls and stack usage. Both methods have linear time complexity, but recursion adds extra overhead. For large input strings, the iterative approach is generally safer and more efficient. The recursive method is better suited for learning and understanding recursion rather than performance-critical applications.