

AI ASSISTANT CODING

Lab Assignment 1.5

Name: AMRUTH SAGAR VEMUGANTI

Ht.No: 2403A51L44

Batch : 52

Task 1: AI-Generated Logic Without Modularization (Reverse a String)

Prompt Used: “write a simple python program Reverse a string without using functions”

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left has a folder named 'TRAINING_3-2' containing files '08-01.py', 'with function.py.py', and 'without func.py'. The 'without func.py' file is open in the center editor, displaying the following Python code:

```
#write a program to reverse a string without using function in Python
# Program to reverse a string in Python without using a function
# Taking input from the user
user_input = input("Enter a string to reverse: ")
# Using slicing to reverse the string
reversed_string = user_input[::-1]
# Displaying the result
print("Reversed string:", reversed_string)
```

The terminal at the bottom shows the execution of the script:

```
PS C:\Users\deepthi\Downloads\Training_3-2> & C:\Users\deepthi\AppData\Local\Programs\Python\Python313\python.exe "c:/Users/deepthi/Downloads/Training_3-2/without_func.py"
PS C:\Users\deepthi\Downloads\Training_3-2> & C:\Users\deepthi\AppData\Local\Programs\Python\Python313\python.exe "c:/Users/deepthi/Downloads/Training_3-2/without_func.py"
● Enter a string to reverse: Varshini
Reversed string: inhsrav
● PS C:\Users\deepthi\Downloads\Training_3-2> & C:\Users\deepthi\AppData\Local\Programs\Python\Python313\python.exe "c:/Users/deepthi/Downloads/Training_3-2/without_func.py"
Enter a string to reverse: Harshitha
Reversed string: ahtihsrat
● PS C:\Users\deepthi\Downloads\Training_3-2>
```

A right-hand sidebar titled 'Build with Agent' contains instructions: 'AI responses may be inaccurate.', 'Generate Agent Instructions to onboard AI onto your codebase.', and a text input field 'Describe what to build'.

- Keeps the program simple
- Suitable for small scripts
- Easy for basic understanding
- No function call overhead

Task 2: AI Code Optimization & Cleanup Original Code:

Prompt Used: “optimize this code & simplify logic and improve readability”

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The Explorer sidebar on the left lists several Python files: 08-01.py, recursion.py, with function.py.py, with&without.py, and without func.py. The 'without func.py' tab is currently active, displaying the following Python code:

```
#write a program to reverse a string without using function in Python and optimize code
# Program to reverse a string in Python without using a function
# Taking input from the user
user_input = input("Enter a string to reverse: ")
reversed_string = user_input[::-1]
# Displaying the result
print("Reversed string:", reversed_string)
```

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

```
PS C:\Users\deepthi\Downloads\Training_3-2> & c:\Users\deepthi\AppData\Local\Programs\Python\Python313\python.exe "c:/Users/deepthi/Downloads/Training_3-2/withoutfunc.py"
Enter a string to reverse: Varshini
Reversed string using function: inihsrav
Reversed string without using function: inihsrav
PS C:\Users\deepthi\Downloads\Training_3-2> & c:\Users\deepthi\AppData\Local\Programs\Python\Python313\python.exe "c:/Users/deepthi/Downloads/Training_3-2/withfunc.py"
Enter a string to reverse: Harshita
Reversed string using function: ahtihsratH
Reversed string without using function: ahtihsratH
PS C:\Users\deepthi\Downloads\Training_3-2> [ ]
```

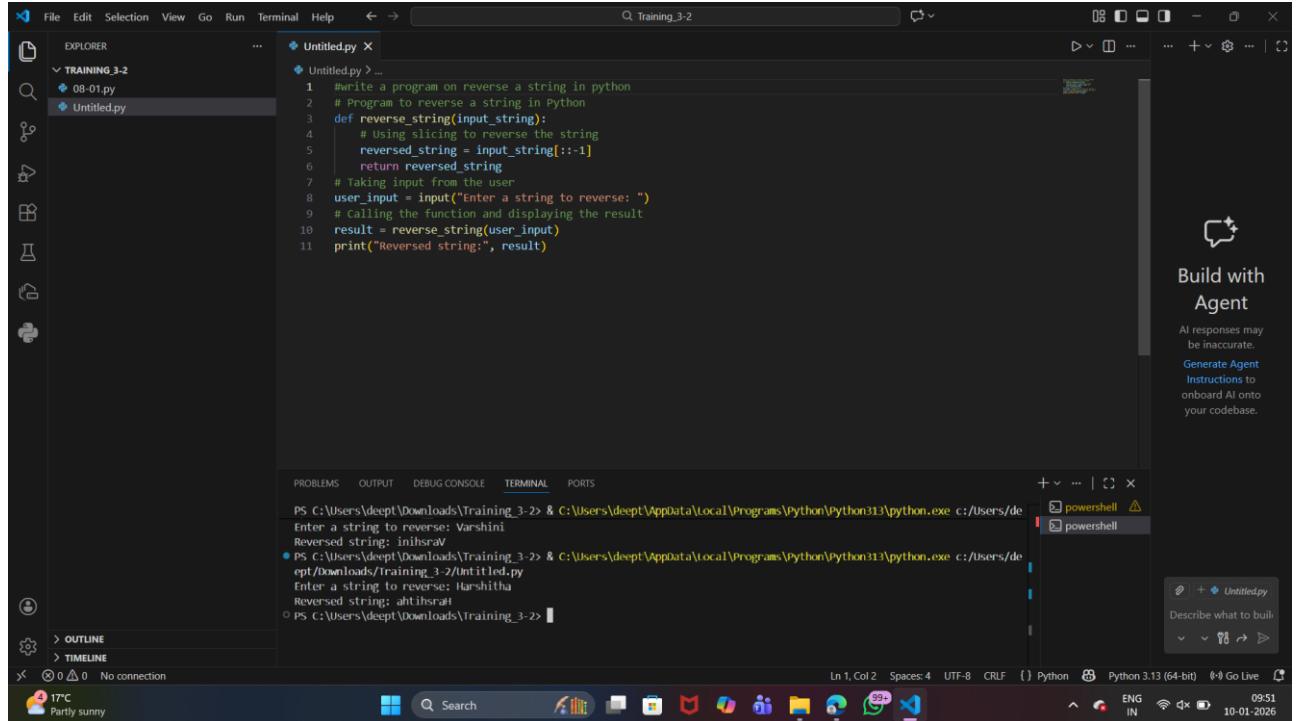
A right-hand sidebar titled 'Build with Agent' is visible, containing a message about AI responses being inaccurate and a 'Generate Agent Instructions' button.

Code is cleaner and easier to maintain

The optimized version improves clarity, maintainability, and readability without affecting performance.

Task 3: Modular Design Using AI Assistance (Reverse a string with Functions)

Prompt Used: “ Write a simple python program of Reverse a string using with function”



The screenshot shows the Visual Studio Code interface. The left sidebar has an 'EXPLORER' view with files '08-01.py' and 'Untitled.py'. The main editor window displays the following Python code:

```
# write a program to reverse a string in python
# Program to reverse a string in Python
def reverse_string(input_string):
    # Using slicing to reverse the string
    reversed_string = input_string[::-1]
    return reversed_string
# Taking input from the user
user_input = input("Enter a string to reverse: ")
# Calling the function and displaying the result
result = reverse_string(user_input)
print("Reversed string:", result)
```

The terminal tab at the bottom shows the output of running the script:

```
PS C:\Users\deepthi\Downloads\Training_3-2> & C:\Users\deepthi\AppData\Local\Programs\Python\Python311\python.exe c:/users/deepthi/downloads/training_3-2/Untitled.py
Enter a string to reverse: Varshini
Reversed string: inisrav
● PS C:\Users\deepthi\Downloads\Training_3-2> & C:\Users\deepthi\AppData\Local\Programs\Python\Python311\python.exe c:/users/deepthi/downloads/training_3-2/Untitled.py
Enter a string to reverse: Harshitha
Reversed string: athsrhai
○ PS C:\Users\deepthi\Downloads\Training_3-2>
```

The status bar at the bottom right shows the date and time: 10-01-2026.

Using functions improves reusability because the same logic can be called multiple times.

It also improves readability and debugging.

Modular code is easier to maintain in large projects.

Task 4: With and Without Using Functions(Reverse a string)

Prompt Used: “ Write a simple python program of Reverse a string using with function and without using function”

```
1 #write a program to reverse a string using with and without function in python
2 # Program to reverse a string in Python
3 def reverse_string(input_string):
4     #Using slicing to reverse the string
5     reversed_string = input_string[::-1]
6     return reversed_string
7 #Taking input from the user
8 user_input = input("Enter a string to reverse: ")
9 #calling the function and displaying the result
10 result = reverse_string(user_input)
11 print("Reversed string using function:", result)
12 #Reversing the string without using function
13 reversed_string_no_func = user_input[::-1]
14 print("Reversed string without using function:", reversed_string_no_func)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\deep\Downloads\Training_3-2> & c:\users\deep\appdata\local\programs\python\python313\python.exe "c:/Users/d...<br/>eopt/Downloads/Training_3-2/with&without.py"<br/>Enter a string to reverse: Varshini<br/>Reversed string using function: inihsrav<br/>Reversed string without using function: inihsrav<br/>PS C:\Users\deep\Downloads\Training_3-2> & c:\users\deep\appdata\local\programs\python\python313\python.exe "c:/Users/d...<br/>eopt/Downloads/Training_3-2/with&without.py"<br/>Enter a string to reverse: Harsitha<br/>Reversed string using function: ahtihsrah<br/>Reversed string without using function: ahtihsrah<br/>PS C:\Users\deep\Downloads\Training_3-2>
```

Ln 14, Col 74 Spaces: 4 UTF-8 CRLF Python Go Live ENG IN 10:07 10-01-2026

Without using functions: Helps beginners clearly understand the basic logic and step-by-step execution of an Reverse a string program.

Using functions: Makes the code modular, reusable, and easier to read and maintain.

Overall: Using functions follows good programming practices, especially for larger or real-world programs.

Task 5: Iterative vs Recursive AI Code

Prompt Used: “Generate iterative and recursive Reverse a string program in Python”

```
1 #write a program on reverse a string using recursion in python
2 #Program to reverse a string using recursion in Python
3 def reverse_string_recursive(input_string):
4     #Base case:if the string is empty or has one character
5     if(len(input_string) <= 1):
6         return input_string
7     #Recursive case:reverse the substring and append the first character at the end
8     return input_string[-1] + reverse_string_recursive(input_string[:-1])
9 #Taking Input from the user
10 user_input = input("Enter a string to reverse: ")
11 #calling the function and displaying the result
12 result = reverse_string_recursive(user_input)
13 print("Reversed String:", result)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\deepthi\Downloads\Training_3-2> & C:/Users/Deepthi/AppData/Local/Programs/Python/Python313/python.exe c:/Users/deept/Downloads/Training_3-2/recursion.py
● PS C:\Users\deepthi\Downloads\Training_3-2> & C:/Users/Deepthi/AppData/Local/Programs/Python/Python313/python.exe c:/Users/deept/Downloads/Training_3-2/08-01.py
Enter a string to reverse: Varchini
Reversed String: inisrhav
PS C:\Users\deepthi\Downloads\Training_3-2> & C:/Users/Deepthi/AppData/Local/Programs/Python/Python313/python.exe c:/Users/deept/Downloads/Training_3-2/recursion.py
● PS C:\Users\deepthi\Downloads\Training_3-2> Enter a string to reverse: Harshitha
Reversed String: ahtihsrH#
```

LN 13, COL 34 SPACES: 4 UTF-8 CRLF () Python ENG IN 10:01 10-01-2026

Execution Flow Explanation

- Iterative version uses loops
- Recursive version uses function calls
- Recursive calls stack memory