AI ASSISTED CODING

NAME :S.Ravikiran

HT NO :2303A52261

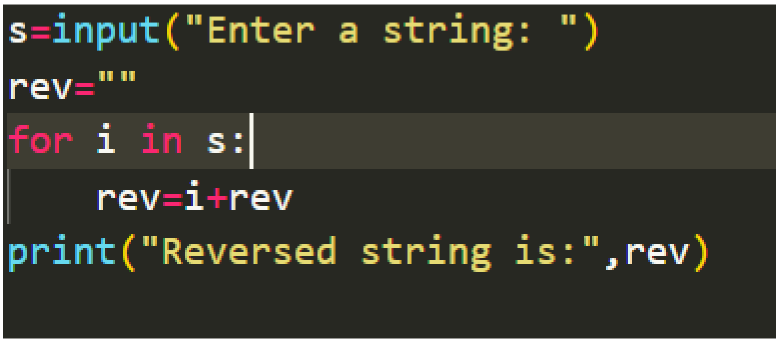
BATCH :36

Lab 1: Environment Setup – GitHub Copilot and VS Code Integration + Understanding AI-assisted Coding Workflow

TASK 1

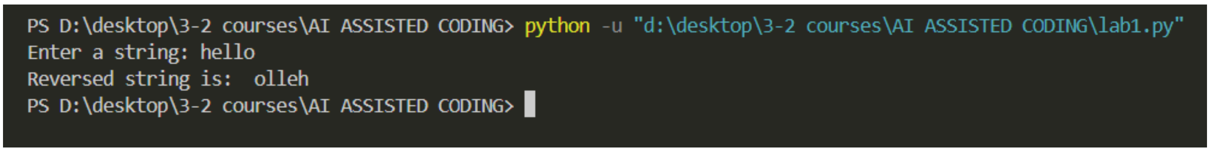
Prompt: write a python program to print reverse of a string without using user defined functions and accepts user input

Code :



```
s=input("Enter a string: ")
rev=""
for i in s:
    rev=i+rev
print("Reversed string is:",rev)
```

Output :



```
PS D:\desktop\3-2 courses\AI ASSISTED CODING> python -u "d:\desktop\3-2 courses\AI ASSISTED CODING\lab1.py"
Enter a string: hello
Reversed string is:  olleh
PS D:\desktop\3-2 courses\AI ASSISTED CODING> 
```

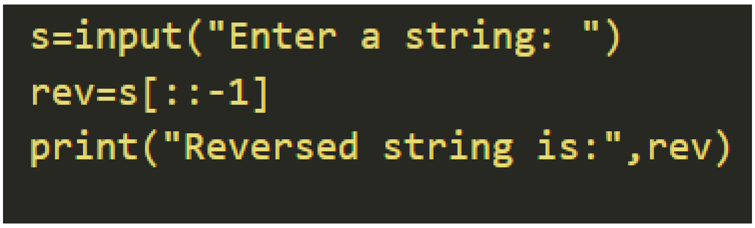
Explanation:

This program takes a string from the user and reverses it. It uses a loop to read each character and adds it to the beginning of a new string, which forms the reverse order. Finally, the reversed string is printed.

TASK 2

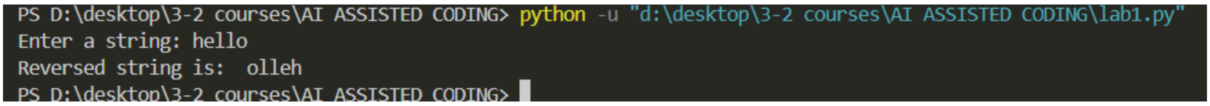
Prompt: write a python program to reverse a string without using functions and optimize the code

Code:



```
s=input("Enter a string: ")
rev=s[::-1]
print("Reversed string is:",rev)
```

Output:



```
PS D:\desktop\3-2 courses\AI ASSISTED CODING> python -u "d:\desktop\3-2 courses\AI ASSISTED CODING\lab1.py"
Enter a string: hello
Reversed string is:  olleh
PS D:\desktop\3-2 courses\AI ASSISTED CODING> 
```

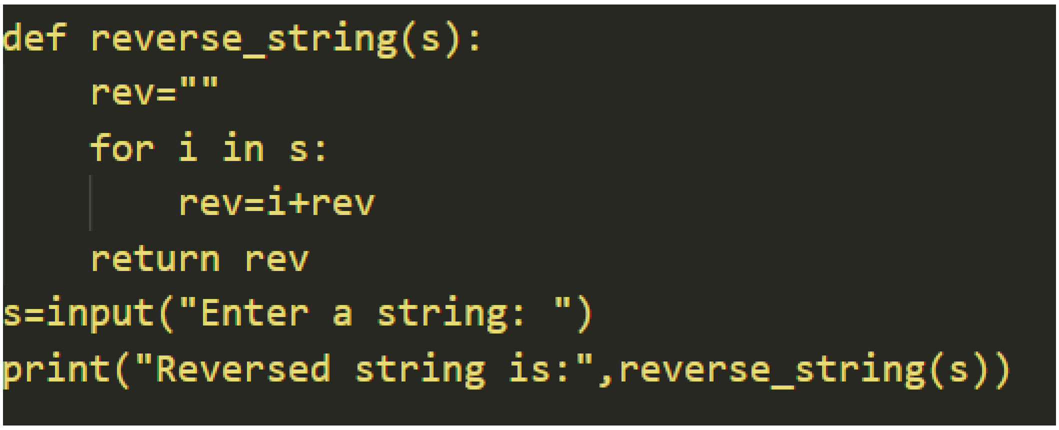
Explanation:

This program takes a string from the user and reverses it using slicing. The expression s[::-1] reads the string from the last character to the first, creating a reversed string. Finally, the reversed string is printed.

TASK-3

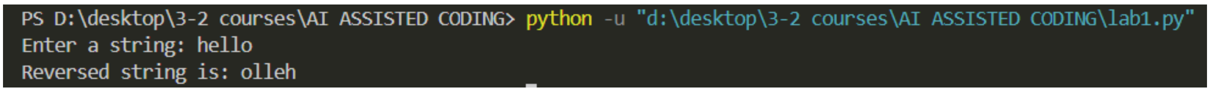
Prompt:write a python program to reverse a string using functions and optimize the code

Code:



```
def reverse_string(s):
    rev=""
    for i in s:
        rev=i+rev
    return rev
s=input("Enter a string: ")
print("Reversed string is:",reverse_string(s))
```

Output:



```
PS D:\desktop\3-2 courses\AI ASSISTED CODING> python -u "d:\desktop\3-2 courses\AI ASSISTED CODING\lab1.py"
Enter a string: hello
Reversed string is: olleh
```

Explanation:

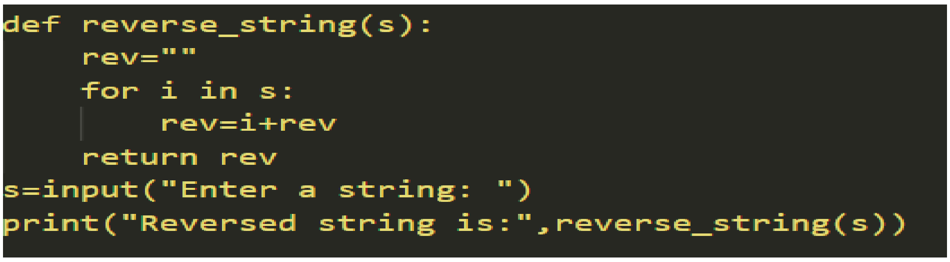
This program defines a function to reverse a string. The function uses slicing to reverse the given string and returns it. The user enters a string, which is passed to the function, and the reversed string is then printed.

TASK -4

Prompt:write a python program to reverse a string using functions and without functions

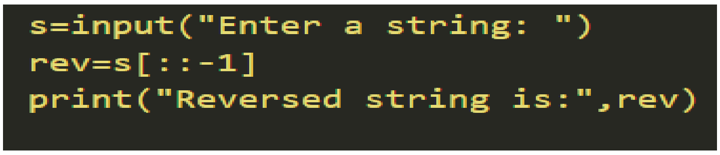
Code:

With functions:



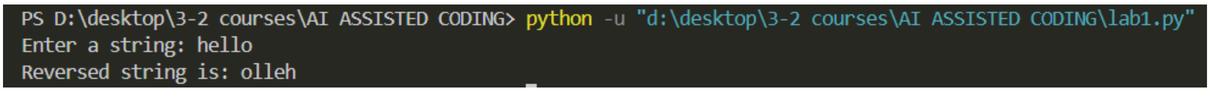
```
def reverse_string(s):
    rev=""
    for i in s:
        rev=i+rev
    return rev
s=input("Enter a string: ")
print("Reversed string is:",reverse_string(s))
```

Without functions:



```
s=input("Enter a string: ")
rev=s[::-1]
print("Reversed string is:",rev)
```

Output:



```
PS D:\desktop\3-2 courses\AI ASSISTED CODING> python -u "d:\desktop\3-2 courses\AI ASSISTED CODING\lab1.py"
Enter a string: hello
Reversed string is: olleh
```

Explanation:

Both programs are used to reverse a string, but they differ in structure and usage.

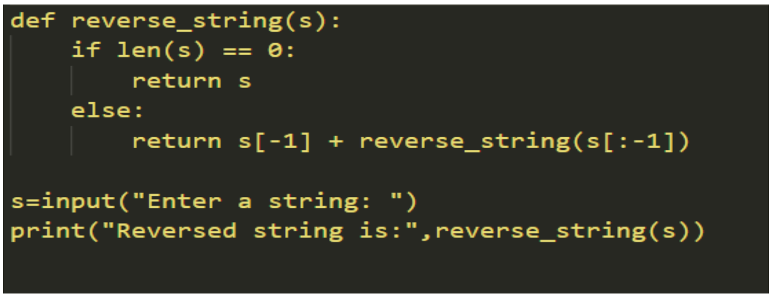
In the first program, the logic for reversing the string is written inside a function called reverse_string. This makes the code more reusable and organized, because the same function can be used to reverse any string in the future. The user input is passed to the function, and the returned reversed string is printed.

In the second program, the string is reversed directly using slicing without using a function. This approach is simpler and shorter, but the logic cannot be reused easily. It is suitable for small programs where the operation is needed only once.

TASK -5

Prompt:write a python program to reverse a string using recursion

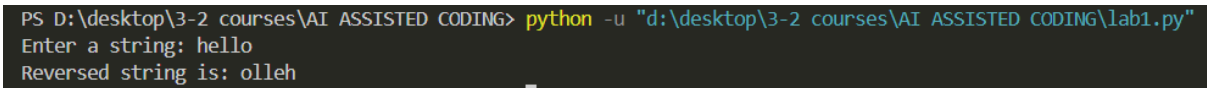
Code:



```
def reverse_string(s):
    if len(s) == 0:
        return s
    else:
        return s[-1] + reverse_string(s[:-1])

s=input("Enter a string: ")
print("Reversed string is:",reverse_string(s))
```

Output:



```
PS D:\desktop\3-2 courses\AI ASSISTED CODING> python -u "d:\desktop\3-2 courses\AI ASSISTED CODING\lab1.py"
Enter a string: hello
Reversed string is: olleh
```

Explanation:

This program reverses a string using **recursion**. The function checks if the string is empty, and if so, it returns the string as it is. Otherwise, it takes the last character of the string and adds it in front of the result of the recursive call made on the remaining part of the string. This process continues until the string becomes empty, and finally the reversed string is printed.