

# ASSIGNMENT-1

NAME: G. Bala Varshitha

HT.NO: 2403A52050

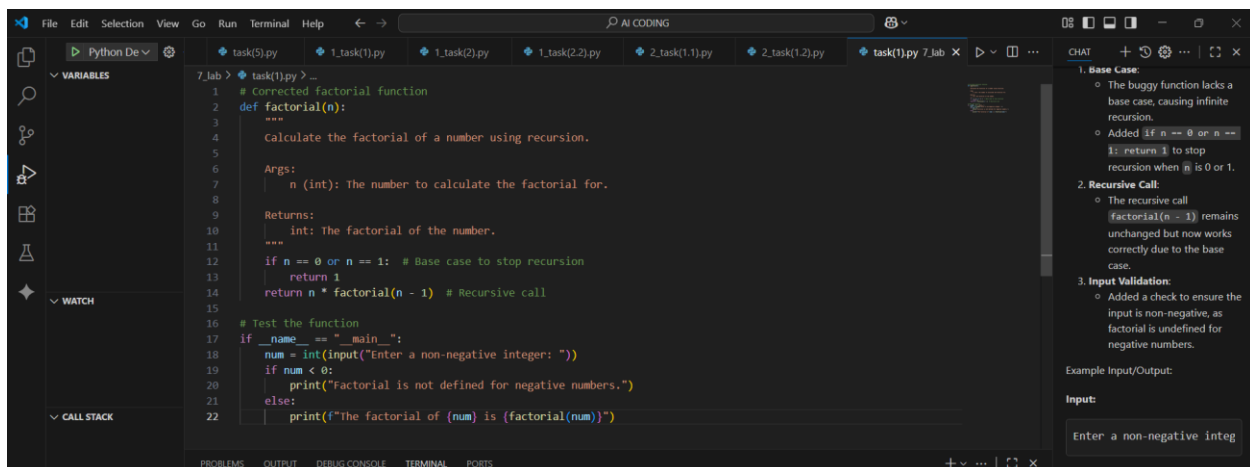
BATCH: AIB03

## Task 1:

Introduce a buggy Python function that calculates the factorial of a number using recursion. Use Copilot or Cursor AI to detect and fix the logical or syntax errors.

Prompt: correct my error.

Code:



The screenshot shows a code editor with a Python file named `task(1).py`. The code defines a `factorial(n)` function. The AI suggestions on the right side of the editor are:

- 1. Base Case:**
  - The buggy function lacks a base case, causing infinite recursion.
  - Added `if n == 0 or n == 1: return 1` to stop recursion when `n` is 0 or 1.
- 2. Recursive Call:**
  - The recursive call `factorial(n - 1)` remains unchanged but now works correctly due to the base case.
- 3. Input Validation:**
  - Added a check to ensure the input is non-negative, as factorial is undefined for negative numbers.

Example Input/Output:

Input: Enter a non-negative integer: 5

Output: The factorial of 5 is 120

OP:



The screenshot shows a terminal window with the following commands and output:

```
PS C:\Users\tharu\OneDrive\Desktop\VAI CODING> cd 'c:\Users\tharu\OneDrive\Desktop\VAI CODING' & 'c:\Program Files\Python11\python.exe' 'c:\Users\tharu\OneDrive\Desktop\VAI CODING\7_lab\task(1).py'
```

Enter a non-negative integer: 5

The factorial of 5 is 120

PS C:\Users\tharu\OneDrive\Desktop\VAI CODING>

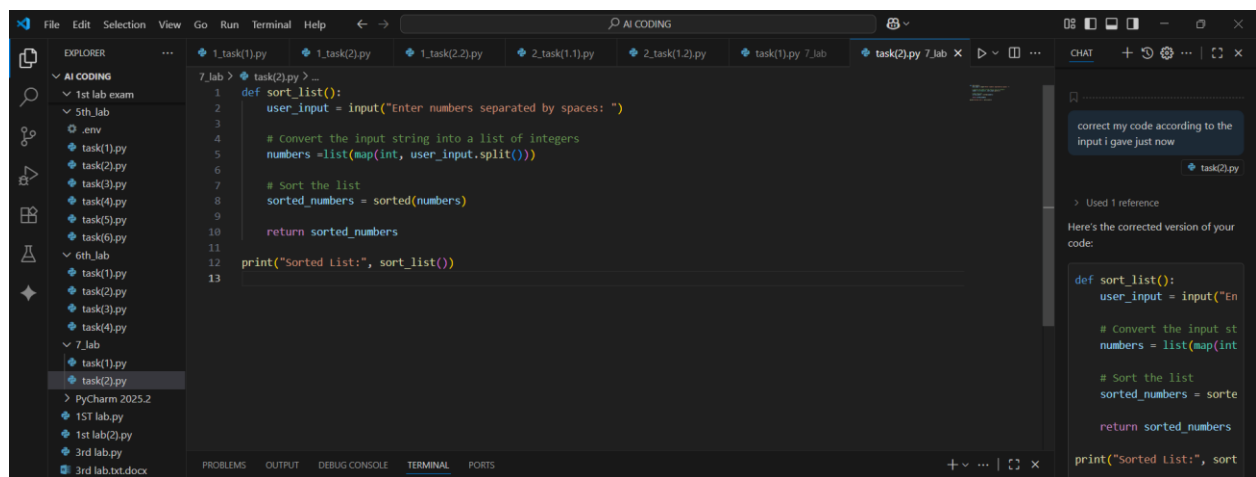
OBSERVATION: The buggy function lacks a base case, causing infinite recursion. Added `if n == 0 or n == 1: return 1` to stop recursion when `n` is 0 or 1. The recursive call `factorial(n - 1)` remains unchanged but now works correctly due to the base case. Added a check to ensure the input is non-negative, as factorial is undefined for negative numbers.

## Task 2:

Provide a list sorting function that fails due to a type error (e.g., sorting list with mixed integers and strings). Prompt AI to detect the issue and fix the code for consistent sorting

Prompt: correct my code according to the input i gave just now

Code:



```
def sort_list():
    user_input = input("Enter numbers separated by spaces: ")
    # Convert the input string into a list of integers
    numbers = list(map(int, user_input.split()))
    # Sort the list
    sorted_numbers = sorted(numbers)
    return sorted_numbers

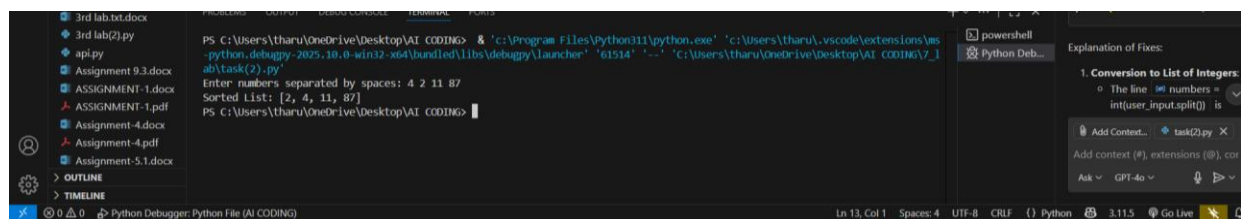
print("Sorted List:", sort_list())
```

correct my code according to the input i gave just now

```
def sort_list():
    user_input = input("En
    # Convert the input st
    numbers = list(map(int
    # Sort the list
    sorted_numbers = sorte
    return sorted_numbers

print("Sorted List:", sort
```

OP:



```
PS C:\Users\tharu\OneDrive\Desktop\AI CODING> & 'c:\Program Files\Python311\python.exe' 'c:\Users\tharu\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundle\libs\debugpy\launcher' '61514' '--' 'c:\Users\tharu\OneDrive\Desktop\AI CODING\7_1'
Enter numbers separated by spaces: 4 2 11 87
Sorted List: [2, 4, 11, 87]
PS C:\Users\tharu\OneDrive\Desktop\AI CODING>
```

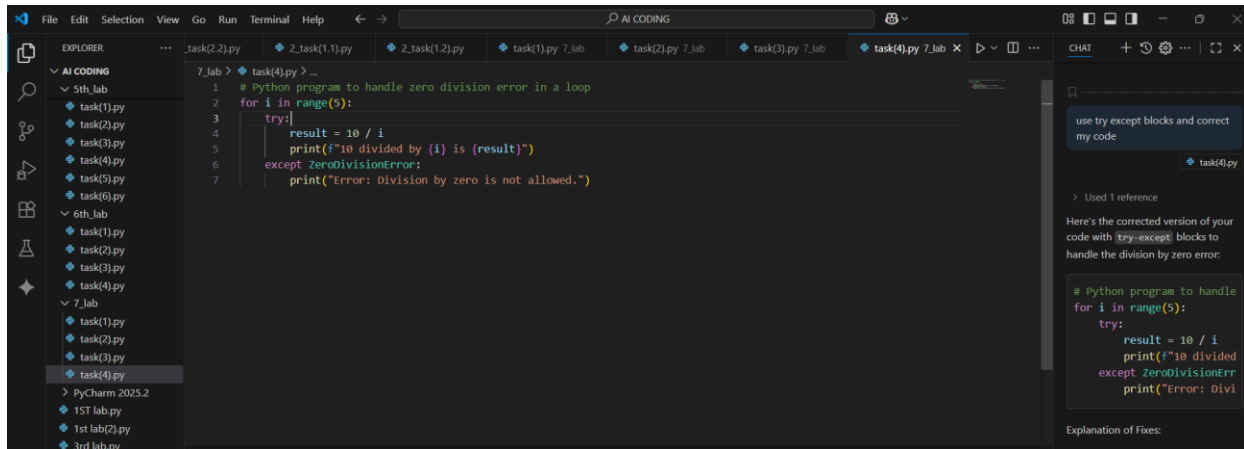
Explanation of Fixes:

1. Conversion to List of Integers: The line `numbers = int(user_input.split())` is incorrect. Fixed by using `list(map(int, user_input.split()))`.

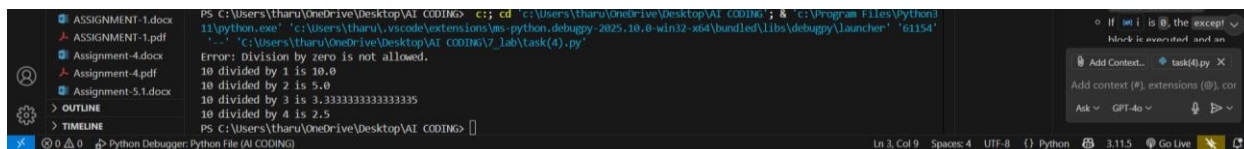
OBSERVATION: The line `number=int(user_input.split())` is incorrect because `split()` returns a list of strings, and `int()` cannot directly convert a list. Fixed by using `list(map(int, user_input.split()))`, which converts each string in the list to an integer. The `sorted()` function works correctly once the input is properly converted to a list of integers.

## Task 3:

Code:



OP:



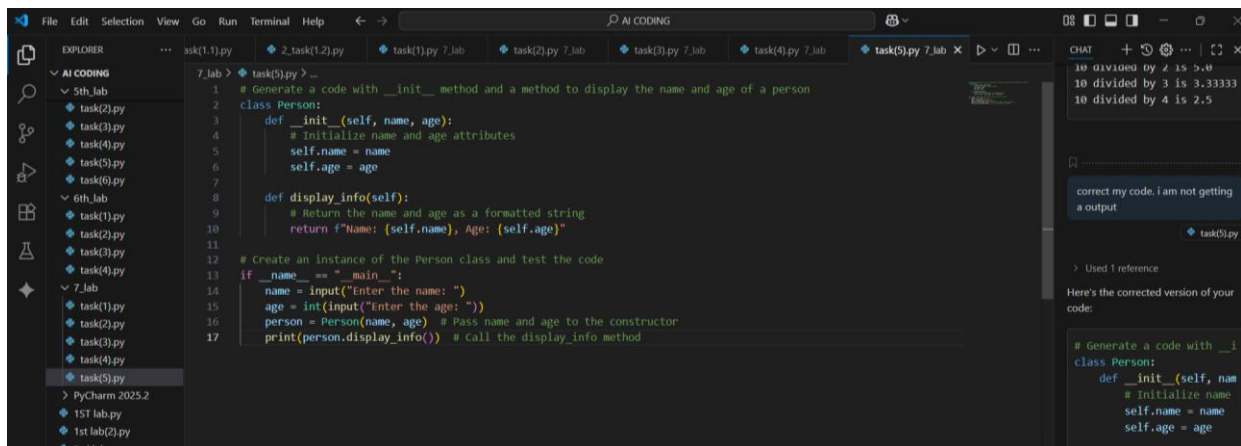
OBSERVATION: Wrapped the division operation `10 / i` in a try block to catch any `ZeroDivisionError`. If `i` is 0, the except block is executed, and an error message is printed. The program continues execution even if a division by zero occurs, thanks to the try-except block.

### Task 5:

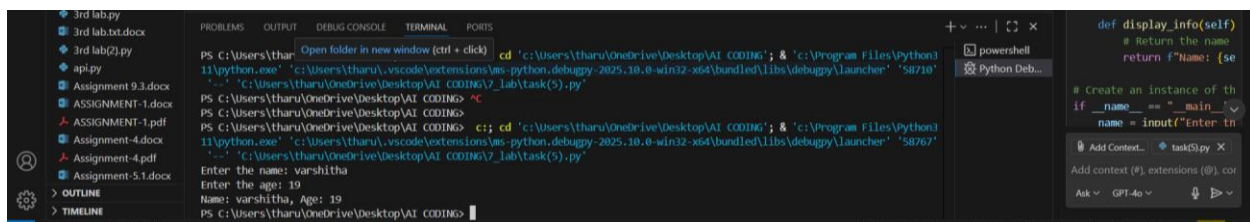
Include a buggy class definition with incorrect `__init__` parameters or attribute references. Ask AI to analyze and correct the constructor and attribute usage

Prompt: correct my code. i am not getting a output

Code:



OP:



OBSERVATION: Added name and age as parameters to the `__init__` method so that they can be passed when creating an instance of the person class. Added input prompts to allow the user to enter the name and age dynamically. Created an instance of the person class using the user-provided name and age. Called the `display_info` method to display the name and age.