ASSIGNMENT-7.4

NAME : G.Sahith Reddy

Roll No : 2403a52183

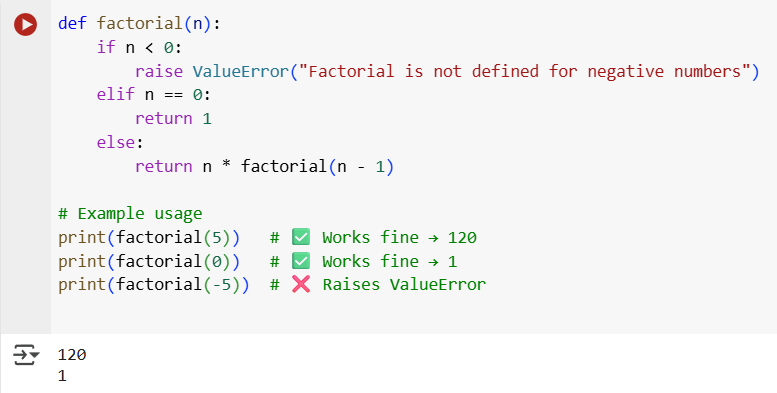
Batch : 07

Task Description #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.  
Expected Outcome #1:  
• Copilot or Cursor AI correctly identifies missing base condition or incorrect recursive call and  
suggests a functional factorial implementation

PROMPT:

Give me a buggy python code that calculates the factorial of a number using recursions and i need to identify the error.

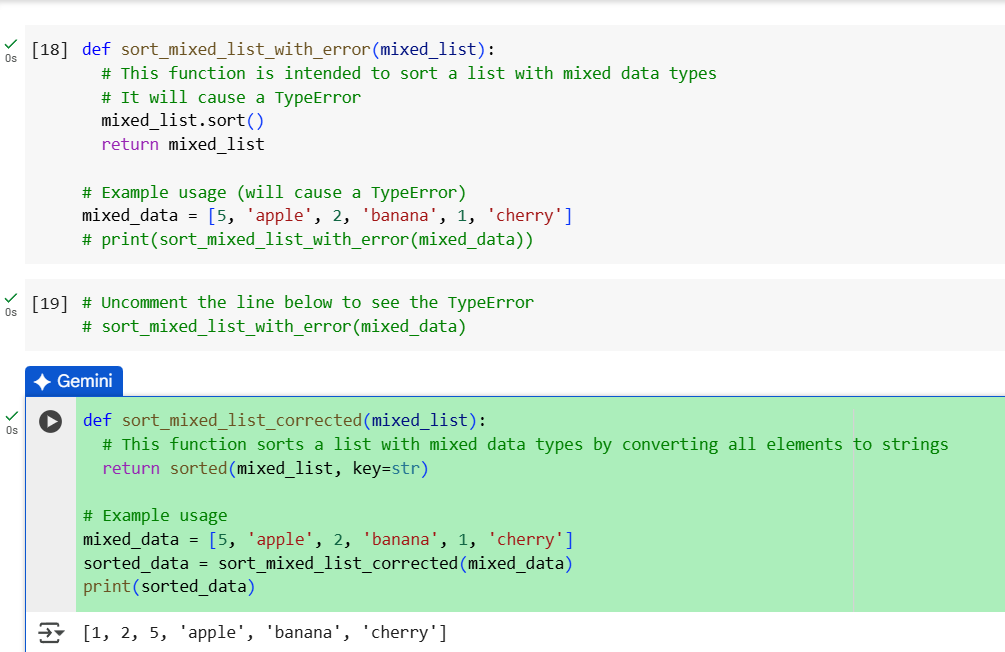




Task Description #2:  
• Provide a list sorting function that fails due to a type error (e.g., sorting list with mixed  
integers and strings). Prompt AI to detect the issue and fix the code for consistent sorting.  
Expected Outcome #2:  
• AI detects the type inconsistency and either filters or converts list elements, ensuring  
successful sorting without a crash.

PROMPT:

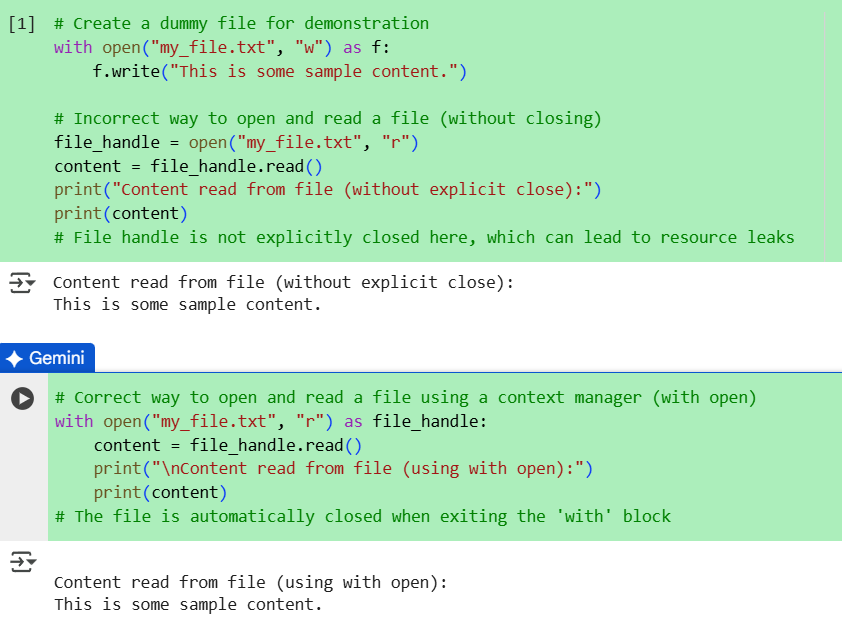
Wrote a function to sort a list, but it fails with a TypeError because the list has both integers and strings. Can you explain the issue and rewrite the function so it sorts the list consistently.



Task Description #3:  
• Write a Python snippet for file handling that opens a file but forgets to close it. Ask Copilot or  
Cursor AI to improve it using the best practice (e.g., with open() block).  
Expected Outcome #3:  
• AI refactors the code to use a context manager, preventing resource leakage and runtime  
warnings.

PROMPT:

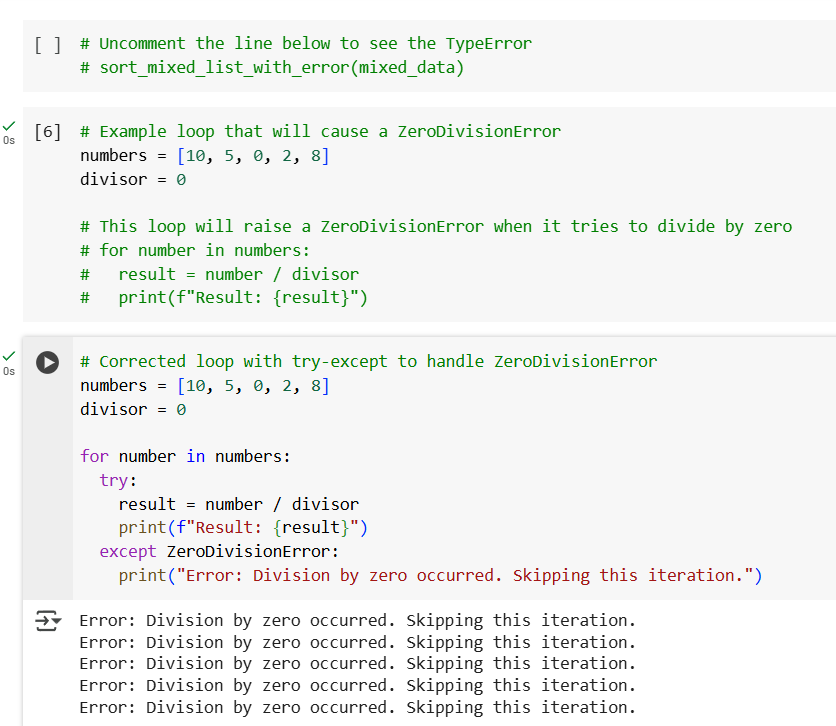
I wrote a Python snippet that opens a file but I forgot to close it. Can you refactor it using best practices like a context manager (with open) to avoid resource leaks



Task Description #4:  
• Provide a piece of code with a ZeroDivisionError inside a loop. Ask AI to add error handling  
using try-except and continue execution safely.  
Expected Outcome #4:  
• Copilot adds a try-except block around the risky operation, preventing crashes and printing a  
meaningful error message.

PROMPT:

I wrote a loop in Python that causes a ZeroDivisionError. Can you add try-except so it handles the error and keeps running safely



Task Description #5:  
• Include a buggy class definition with incorrect \_\_init\_\_ parameters or attribute references. Ask  
AI to analyze and correct the constructor and attribute usage.  
Expected Outcome #5:  
• Copilot identifies mismatched parameters or missing self references and rewrites the class with  
accurate initialization and usage

PROMPT:

I wrote a Python class with mistakes in the \_\_init\_\_ method and attribute references. Can you fix the constructor and correct the attribute usage

