AI-assisted coding

Assignment -7.4

Name: V. Vyshnavi

Batch-14

HT.No.: 2303A51968

Task 1: Debugging a Recursive Calculation Module

Scenario

You are maintaining a utility module in a software project that performs mathematical computations. One function is meant to calculate the factorial of a number, but users are reporting crashes or incorrect outputs.

Task Description

You are given a Python function intended to calculate the factorial of a number using recursion, but it contains logical or syntactical errors (such as a missing base condition or incorrect recursive call).

Use GitHub Copilot or Cursor AI to:

• Analyze the faulty code

• Identify the exact cause of the error

• Suggest and apply corrections to make the function work

Correctly Document how the AI detected the issue and what changes were made.

Expected Outcome

• A corrected recursive factorial function

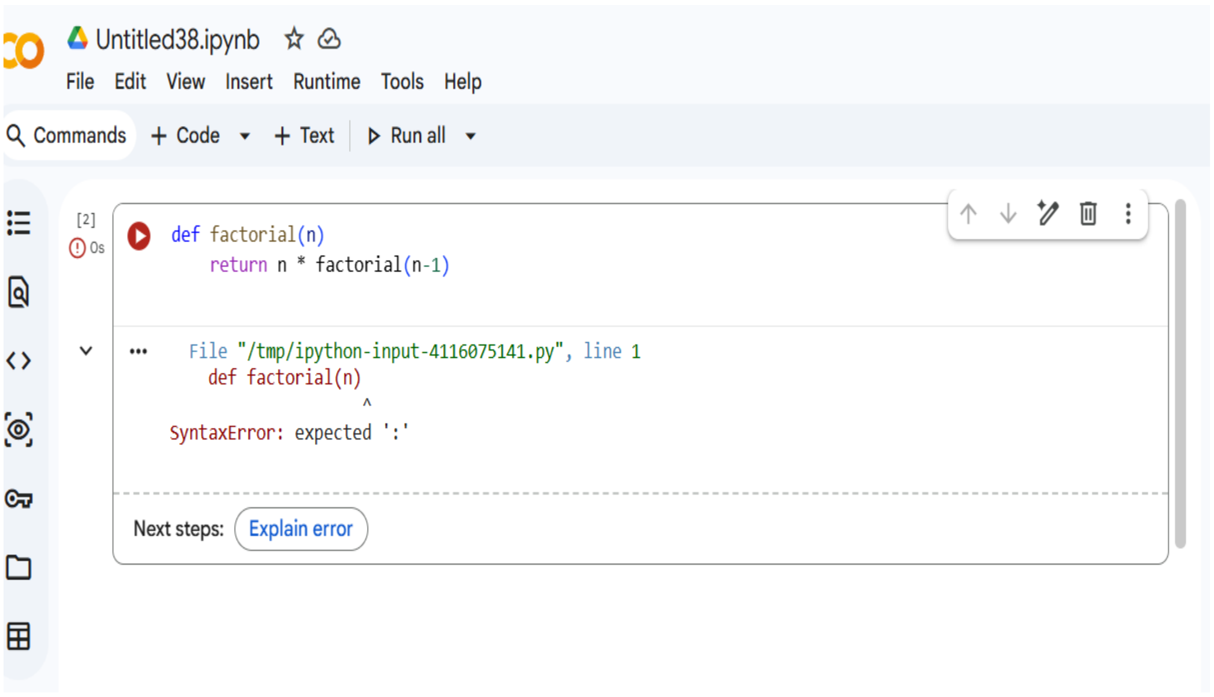
• AI-generated explanation identifying:

o The missing or incorrect base case

o The corrected recursive logic

• Sample input/output demonstrating correct execution

Code with error:

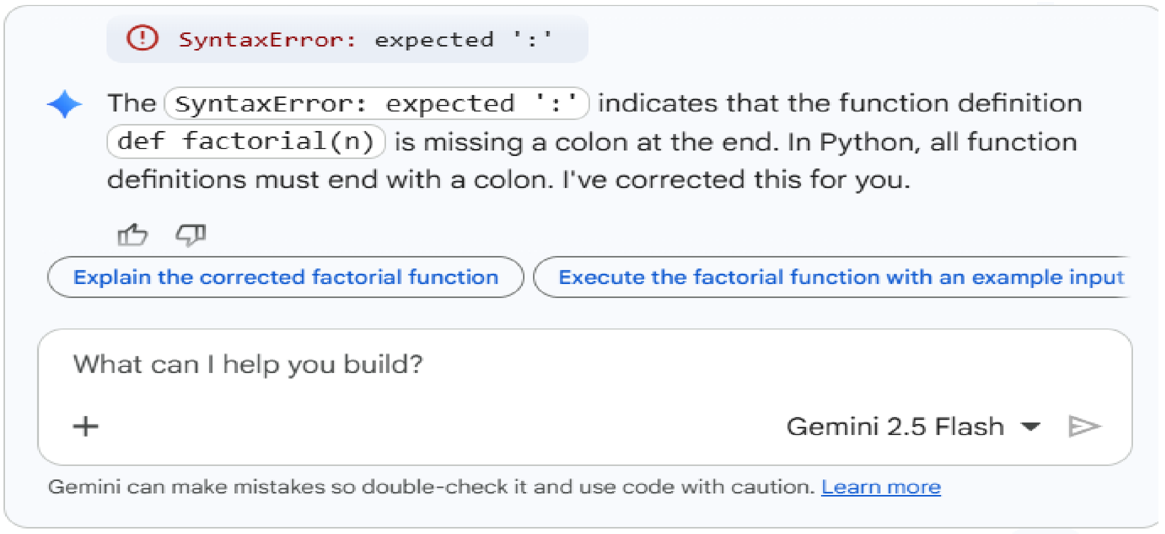


Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[2]
0s
Next steps: Explain error

```
def factorial(n)
    return n * factorial(n-1)
```

```
  File "/tmp/ipython-input-4116075141.py", line 1
    def factorial(n)
                    ^
SyntaxError: expected ':'
```

Explanation:

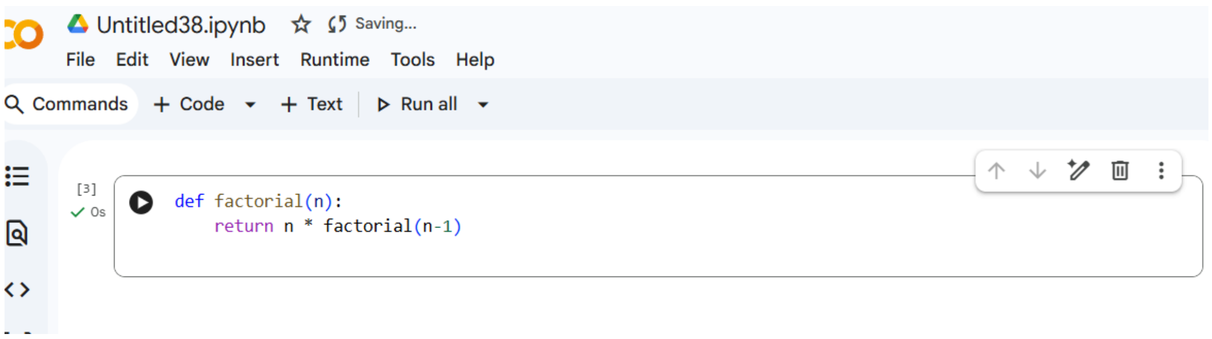


```
SyntaxError: expected ':'
```

The `SyntaxError: expected ':'` indicates that the function definition `def factorial(n)` is missing a colon at the end. In Python, all function definitions must end with a colon. I've corrected this for you.

Explain the corrected factorial function
Execute the factorial function with an example input
What can I help you build?
Gemini 2.5 Flash
Gemini can make mistakes so double-check it and use code with caution. Learn more

Actual code:



```python
def factorial(n):
    return n * factorial(n-1)
```

Task 2: Fixing Data Type Errors in a Sorting Utility

Scenario

You are developing a data processing script that sorts user input values.

The program crashes when users enter mixed data types.

Task Description

You are provided with a list-sorting function that fails due to a

TypeError caused by mixed data types (e.g., integers and strings).

Use GitHub Copilot or Cursor AI to:

• Detect the root cause of the runtime error

• Modify the code to ensure consistent sorting (by filtering or type

conversion)

• Prevent the program from crashing Explain the debugging steps followed by the AI.

Expected Outcome

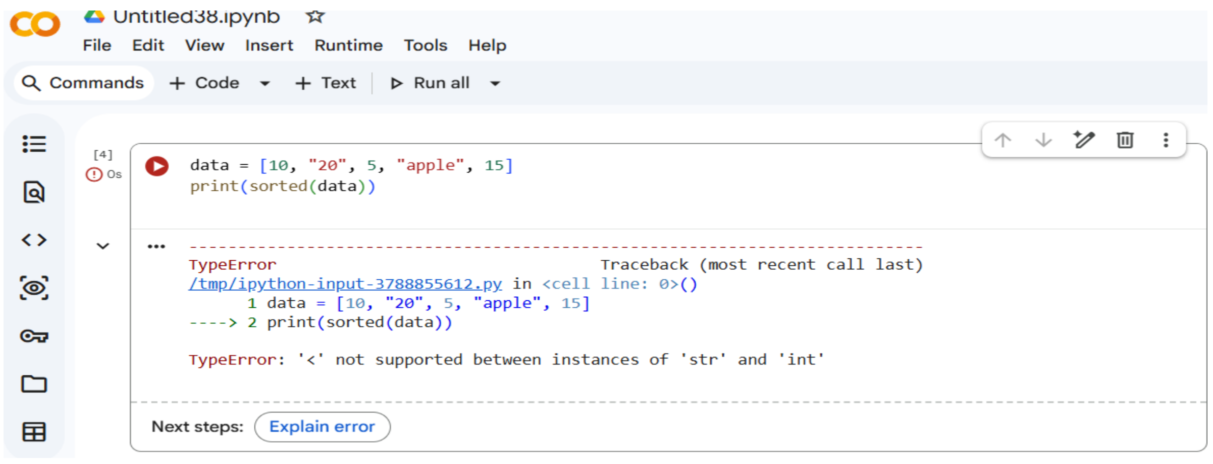
• A corrected sorting function

• AI-generated solution handling type inconsistencies

• Successful sorting without runtime errors

• Explanation of how the fix improves robustness

Code with error:



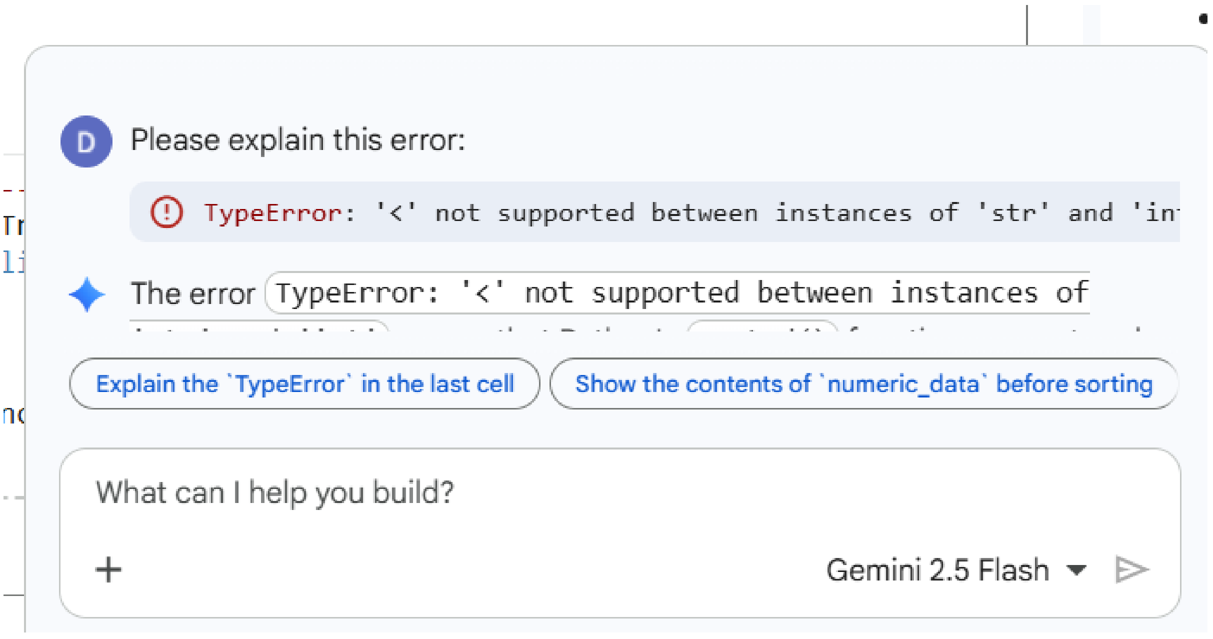
Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all

```
data = [10, "20", 5, "apple", 15]
print(sorted(data))
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-3788855612.py in <cell line: 0>()
      1 data = [10, "20", 5, "apple", 15]
----> 2 print(sorted(data))

TypeError: '<' not supported between instances of 'str' and 'int'
```

Next steps: Explain error

Explanation:



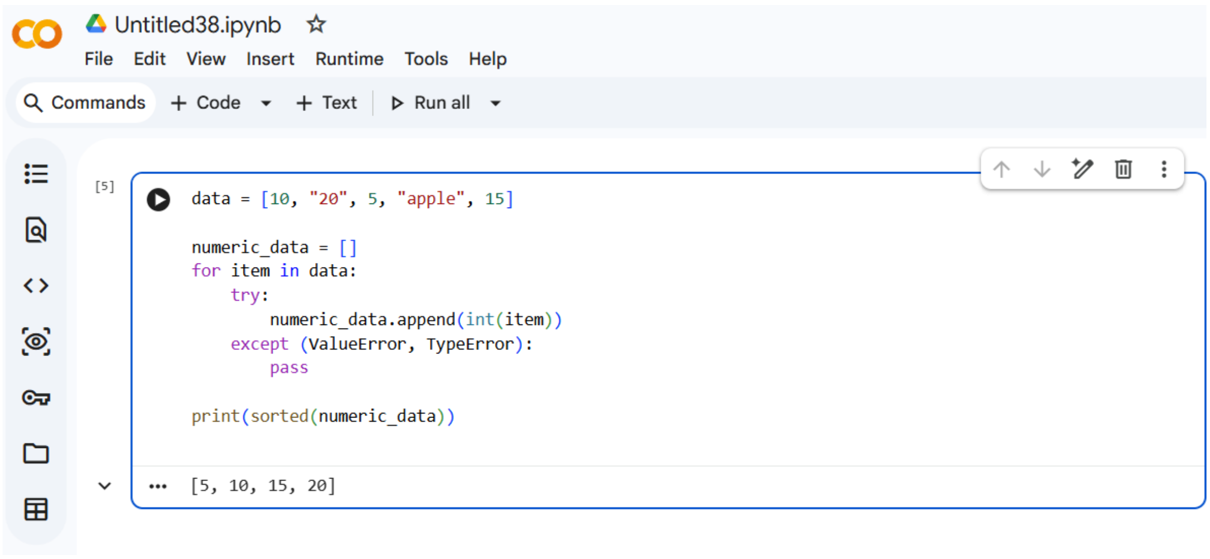
Please explain this error:

```
TypeError: '<' not supported between instances of 'str' and 'in
```

The error `TypeError: '<' not supported between instances of`

Explain the `TypeError` in the last cell
Show the contents of `numeric_data` before sorting
What can I help you build?
Gemini 2.5 Flash

Actual code:


Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[5]

```python
data = [10, "20", 5, "apple", 15]

numeric_data = []
for item in data:
    try:
        numeric_data.append(int(item))
    except (ValueError, TypeError):
        pass

print(sorted(numeric_data))
```

```
[5, 10, 15, 20]
```

Task 3: Improving File Handling Reliability

Scenario

A backend script reads data from files regularly. Over time, the system shows performance issues due to improper resource management.

Task Description

You are given a Python file-handling snippet that opens a file but does not explicitly close it.

Use GitHub Copilot or Cursor AI to:

• Identify the potential problem in the code

• Refactor it using best practices (such as a context manager)

• Ensure safe and reliable file handling Briefly describe why the revised approach is better.

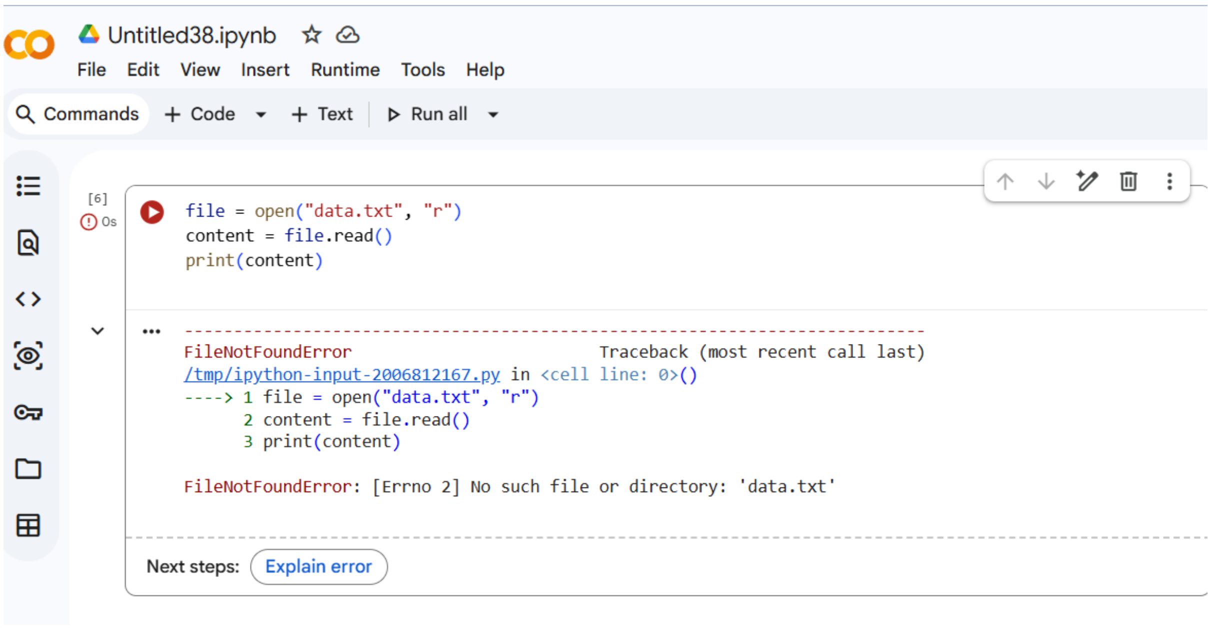
Expected Outcome

• Refactored code using the with open() statement

• AI explanation highlighting prevention of resource leaks

• Clean execution without warnings or errors

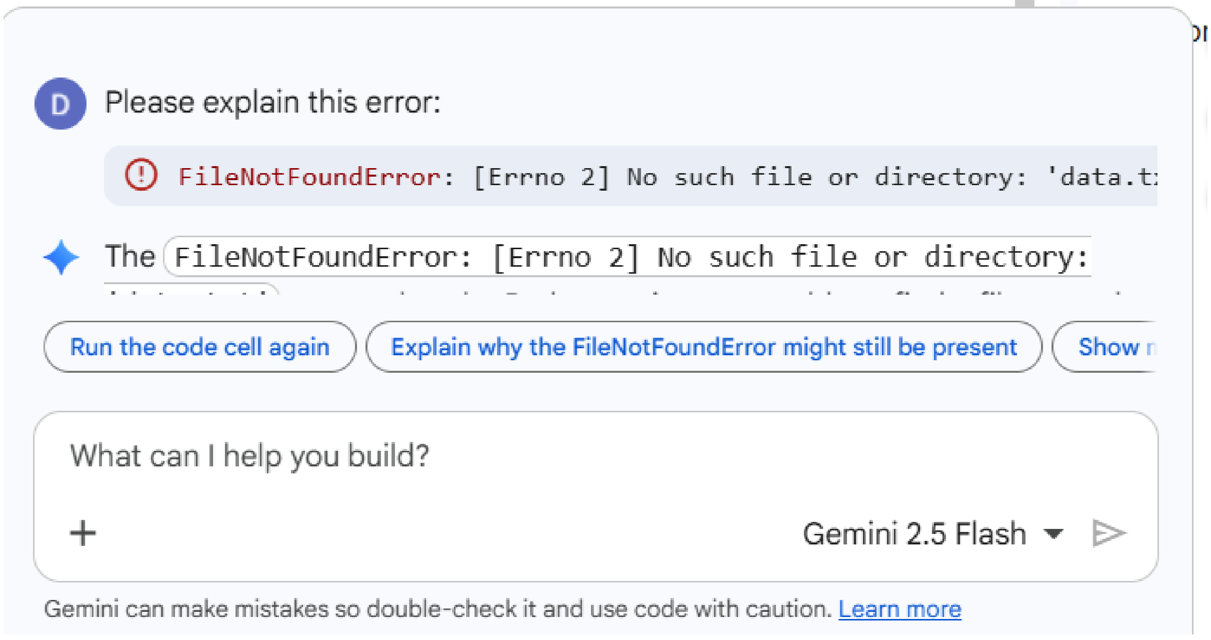
Code with error:


Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[6]
0s
Next steps: Explain error

```
file = open("data.txt", "r")
content = file.read()
print(content)
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
/tmp/ipython-input-2006812167.py in <cell line: 0>()
----> 1 file = open("data.txt", "r")
      2 content = file.read()
      3 print(content)

FileNotFoundError: [Errno 2] No such file or directory: 'data.txt'
```

Explanation:


D
Run the code cell again
Explain why the FileNotFoundError might still be present
Show r
What can I help you build?
Gemini 2.5 Flash

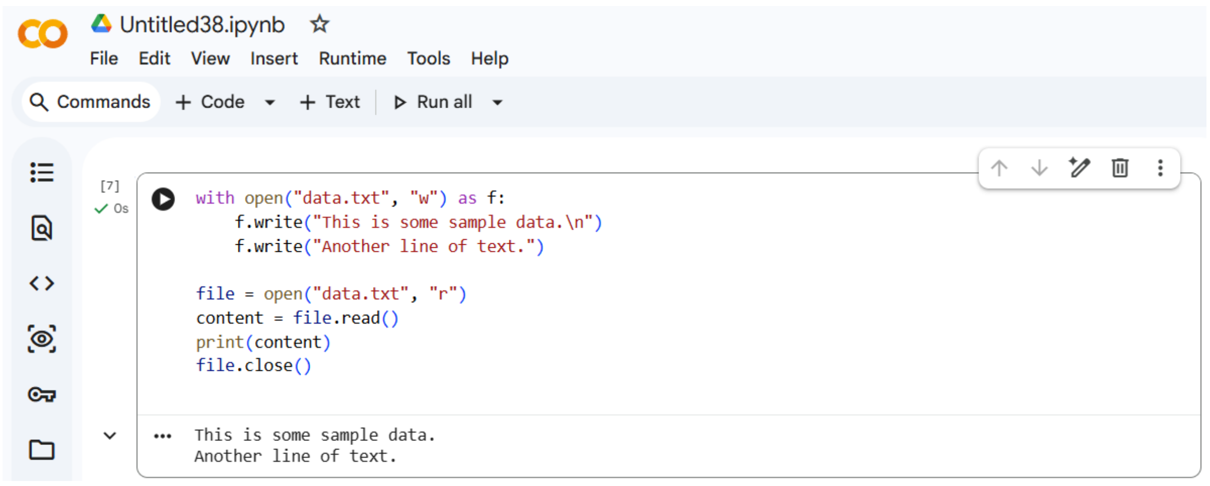
Please explain this error:

`FileNotFoundError: [Errno 2] No such file or directory: 'data.t`

The `FileNotFoundError: [Errno 2] No such file or directory:`

Gemini can make mistakes so double-check it and use code with caution. Learn more

Actual code:


Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[7]
0s

```python
with open("data.txt", "w") as f:
    f.write("This is some sample data.\n")
    f.write("Another line of text.")

file = open("data.txt", "r")
content = file.read()
print(content)
file.close()
```

```
This is some sample data.
Another line of text.
```

Task 4: Handling Runtime Errors Gracefully in Loops Scenario

You are working on a data analysis script that processes a list of values.

Some values cause runtime errors, but the program should continue processing remaining data.

Task Description

You are provided with a code snippet containing a ZeroDivisionError inside a loop.

Use GitHub Copilot or Cursor AI to:

• Detect the exact location of the error

• Add appropriate exception handling using try-except

• Ensure the loop continues executing safely

Document how AI improved the fault tolerance of the program.

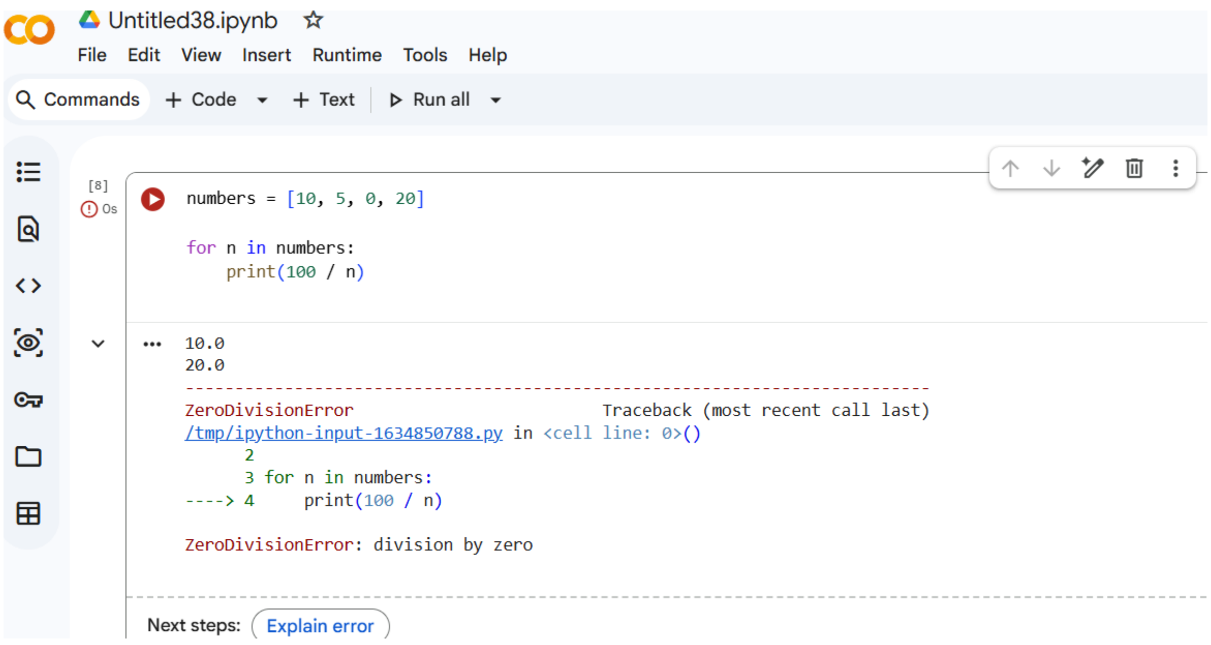
Expected Outcome

• Updated code with proper exception handling

• Meaningful error messages instead of program crashes

• Successful execution for all valid inputs

Code with error:



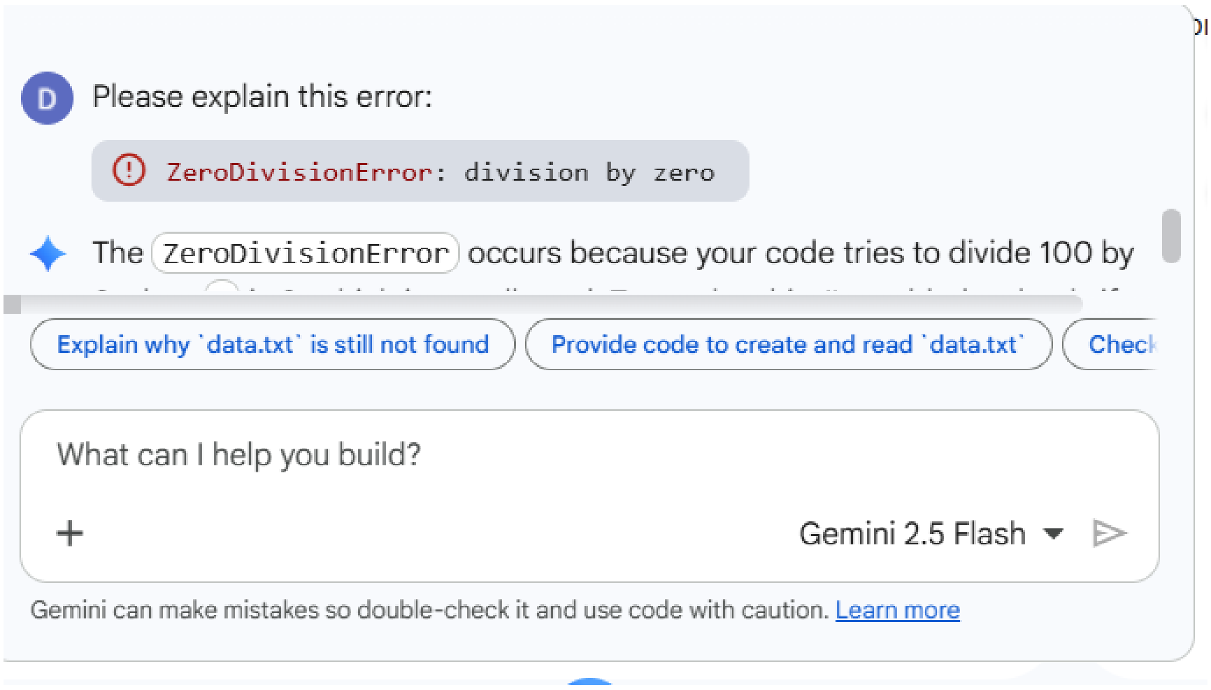
Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands
+ Code
+ Text
Run all
[8]
0s

```
numbers = [10, 5, 0, 20]

for n in numbers:
    print(100 / n)
```

```
10.0
20.0
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
/tmp/ipython-input-1634850788.py in <cell line: 0>()
      2 
      3 for n in numbers:
----> 4     print(100 / n)

ZeroDivisionError: division by zero
```

Next steps: Explain error

Explanation:



Please explain this error:

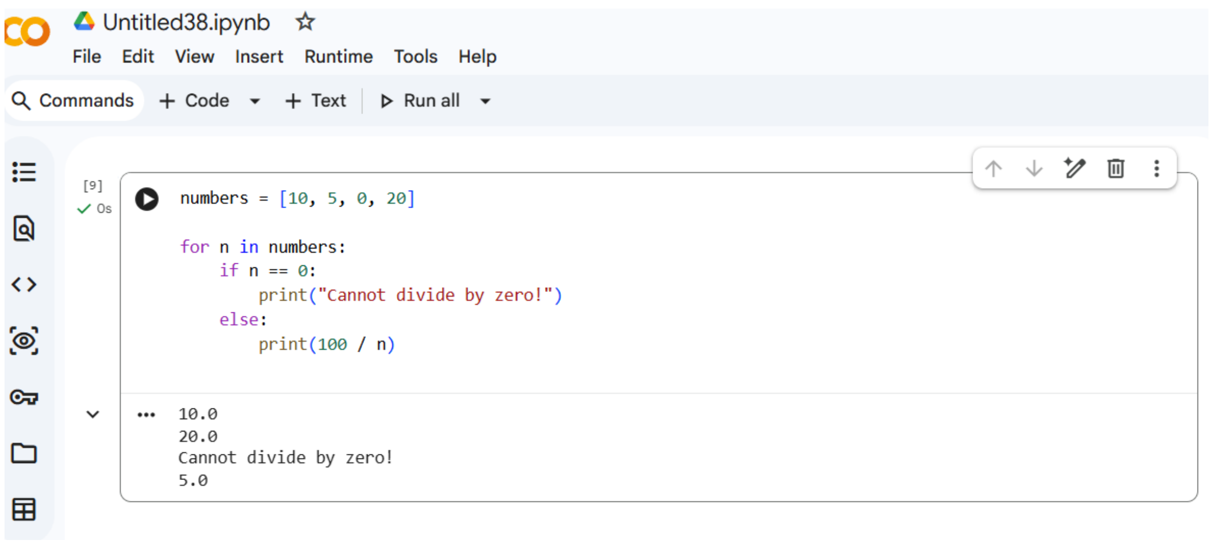
ZeroDivisionError: division by zero

The ZeroDivisionError occurs because your code tries to divide 100 by

Explain why `data.txt` is still not found
Provide code to create and read `data.txt`
What can I help you build?
Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. Learn more

Actual code:


Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[9]
0s

```
numbers = [10, 5, 0, 20]

for n in numbers:
    if n == 0:
        print("Cannot divide by zero!")
    else:
        print(100 / n)
```

```
10.0
20.0
Cannot divide by zero!
5.0
```

Task 5: Debugging Class Initialization Errors Scenario

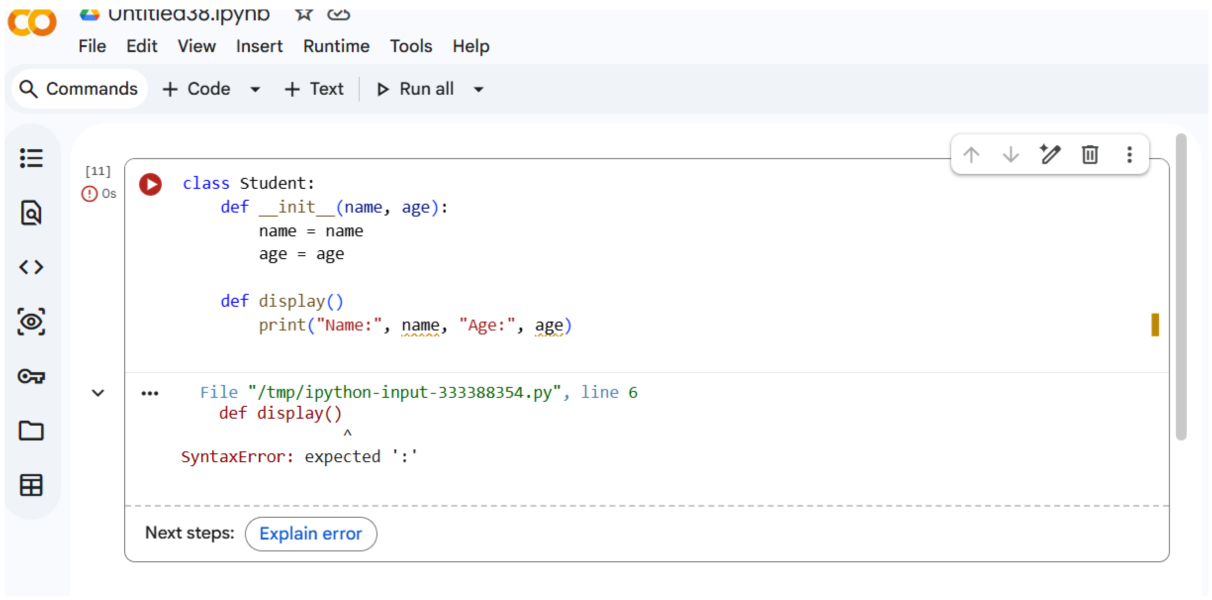
A class written by a junior developer is throwing unexpected errors when objects are created or attributes are accessed.

Task Description

You are given a Python class with:

• Incorrect __init__ parameters

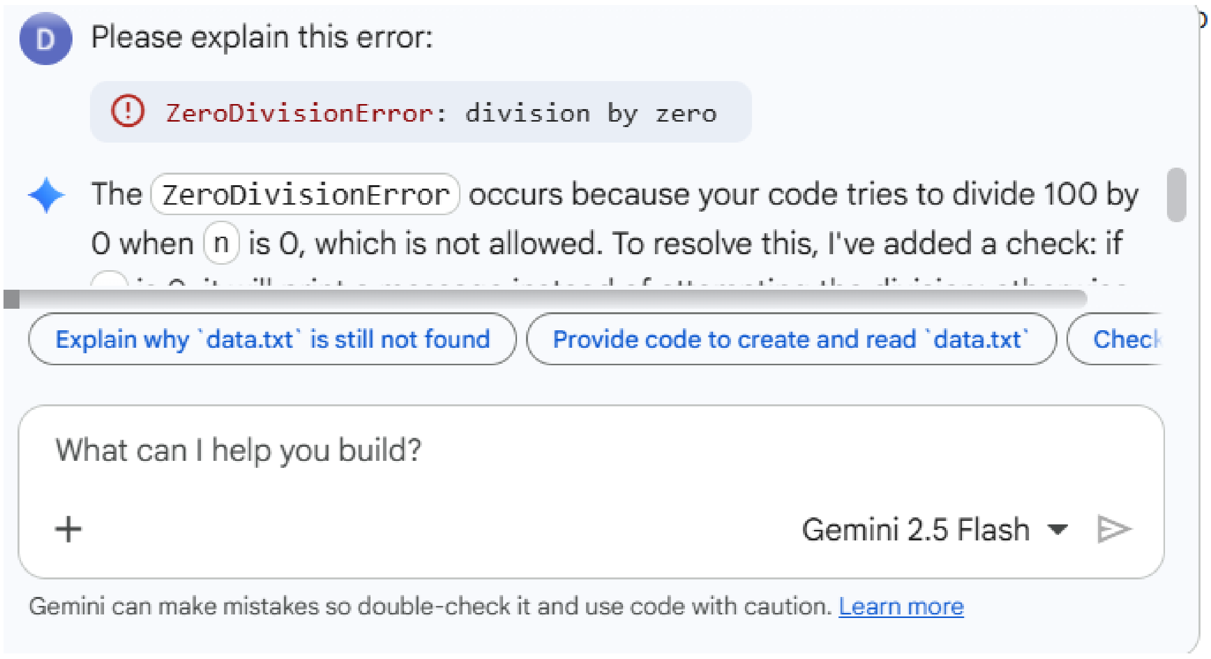
Code with error:


Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[11]
0s
Next steps: Explain error

```
class Student:
    def __init__(name, age):
        name = name
        age = age

    def display()
        print("Name:", name, "Age:", age)
```

```
  File "/tmp/ipython-input-333388354.py", line 6
    def display()
                 ^
SyntaxError: expected ':'
```

Explanation:



D Please explain this error:

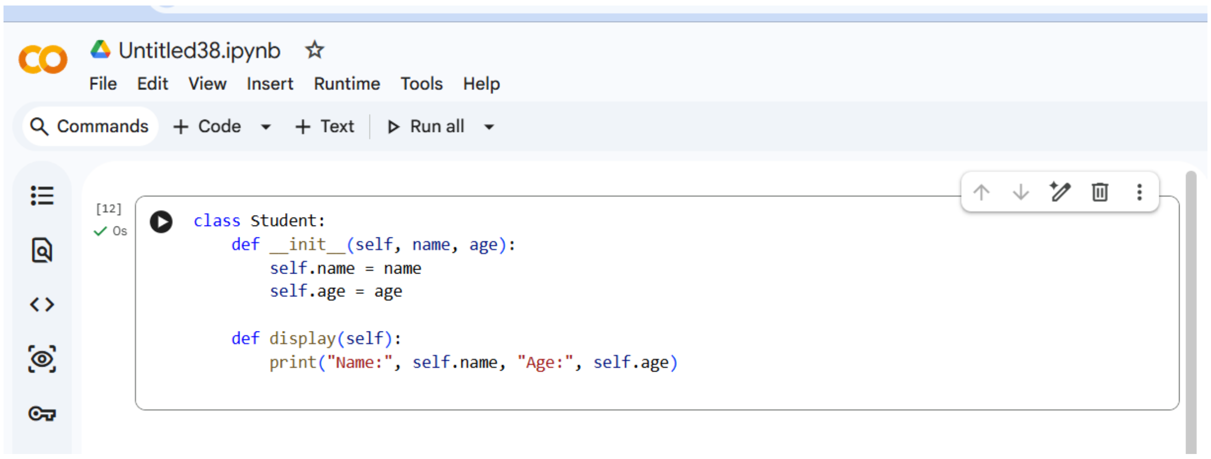
```
ZeroDivisionError: division by zero
```

The `ZeroDivisionError` occurs because your code tries to divide 100 by 0 when `n` is 0, which is not allowed. To resolve this, I've added a check: if

Explain why `data.txt` is still not found
Provide code to create and read `data.txt`
Check
What can I help you build?
Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. Learn more

Actual code:



Untitled38.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[12]
0s

```
class Student:
    def __init__(self, name, age):
        self.name = name
        self.age = age

    def display(self):
        print("Name:", self.name, "Age:", self.age)
```