AI Assisted Coding

Assignment – 7.1

M.Sathwik || 2303A51483 || Batch:- 08

Task Description #1 (Syntax Errors – Missing Parentheses in Print

Statement)

Task: Provide a Python snippet with a missing parenthesis in a print

statement (e.g., print "Hello"). Use AI to detect and fix the syntax error.

Bug: Missing parentheses in print statement def

greet():

print "Hello, AI Debugging Lab!" greet()

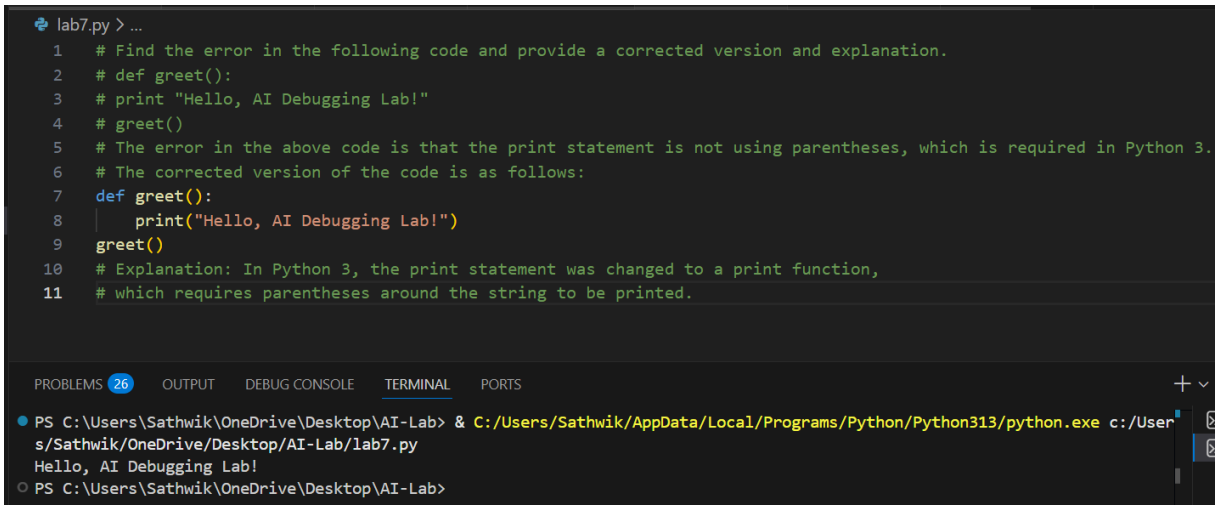
Requirements:

• Run the given code to observe the error.

• Apply AI suggestions to correct the syntax.

• Use at least 3 assert test cases to confirm the corrected code

works.



Task Description #2 (Incorrect condition in an If Statement) Task:

Supply a function where an if-condition mistakenly uses = instead of

==. Let AI identify and fix the issue.

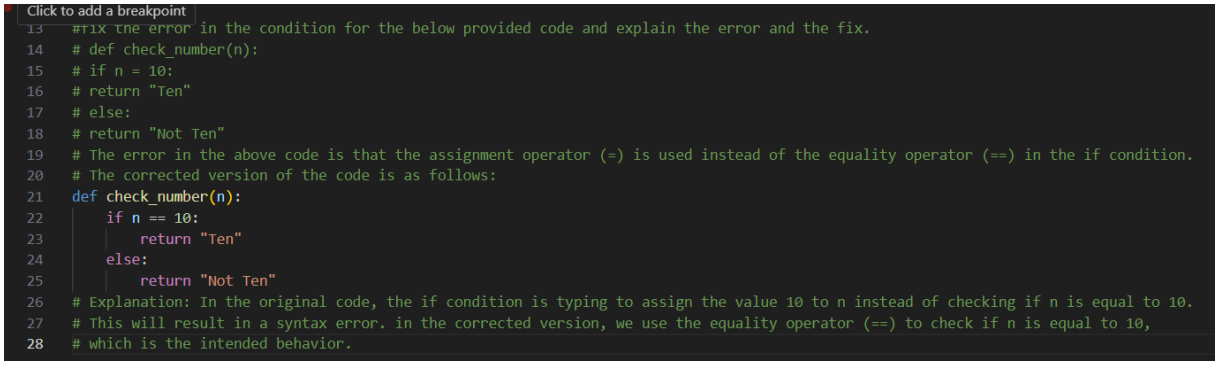
Bug: Using assignment (=) instead of comparison (==)

def check_number(n): if n = 10: return "Ten" else:

return "Not Ten" Requirements:

• Ask AI to explain why this causes a bug.

• Correct the code and verify with 3 assert test cases.



Task Description #3 (Runtime Error – File Not Found)

Task: Provide code that attempts to open a non-existent file and

crashes. Use AI to apply safe error handling. # Bug: Program crashes

if file is missing def read_file(filename): with open(filename, 'r') as

f:

return f.read() print(read_file("nonexistent.txt"))

Requirements:

• Implement a try-except block suggested by AI.

• Add a user-friendly error message.

• Test with at least 3 scenarios: file exists, file missing, invalid path.

```python
#Handle the error in the below code by try and except block and explain the error and add the user friendly error message.
# def read_file(filename):
# with open(filename, 'r') as f:
# return f.read()
# print(read_file("nonexistent.txt"))
# The error in the above code is that it tries to read a file that does not exist, which will raise a FileNotFoundError.
# The corrected version of the code with error handling is as follows:
def read_file(filename):
    try:
        with open(filename, 'r') as f:
            return f.read()
    except FileNotFoundError:
        return f"Error: The file '{filename}' was not found. Please check the filename and try again."
print(read_file("nonexistent.txt"))
# Explanation: In the original code, if the file "nonexistent.txt" does not exist,
# it will raise a FileNotFoundError and the program will crash.
# In the corrected version, we wrap the file reading operation in a try-except block to catch the FileNotFoundError.
# If the file is not found,we return a user-friendly error message that informs the user about the issue and suggests checking the filename.
# This way, the program does not crash and provides helpful feedback to the user.
```

```
PROBLEMS 26  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\Sathwik\OneDrive\Desktop\AI-Lab> & C:/Users/Sathwik/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Sathwik/OneDrive/Desktop/AI-Lab/l
py
Error: The file 'nonexistent.txt' was not found. Please check the filename and try again.
PS C:\Users\Sathwik\OneDrive\Desktop\AI-Lab>
```

Task Description #4 (Calling a Non-Existent Method)

Task: Give a class where a non-existent method is called (e.g.,

obj.undefined_method()). Use AI to debug and fix.

Bug: Calling an undefined method

class Car: def start(self): return "Car

started" my_car = Car()

print(my_car.drive()) # drive() is not defined Requirements:

- Students must analyze whether to define the missing method or correct the method call.
- Use 3 assert tests to confirm the corrected class works.

```python
#In the below code fix the bug related to the calling an undefined method and explain the error and the fix.
# class Car:
# def start(self):
# return "Car started"
# my_car = Car()
# print(my_car.drive()) #drive() is not defined
# The error in the above code is that the method `drive()` is called on the `my_car` object, but it is not defined in the `Car` class.
# The corrected version of the code is as follows:
class Car:
    def start(self):
        return "Car started"

    def drive(self):
        return "Car is driving"
my_car = Car()
print(my_car.drive()) #Now drive() is defined
# Explanation: In the original code, the `drive` method is called on the `my_car` object,
# but since it is not defined in the `Car` class, it will raise an AttributeError.
# In the corrected version, we define the `drive()` method within the `Car` class,
# which allows us to call it on the `my_car` object without any errors.
```

```
PROBLEMS 26  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\Sathwik\OneDrive\Desktop\AI-Lab> & C:/Users/Sathwik/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Sathwik/OneDrive/Desktop/AI-
py
Car is driving
PS C:\Users\Sathwik\OneDrive\Desktop\AI-Lab>
```

Task Description #5 (TypeError – Mixing Strings and Integers in

Addition)

Task: Provide code that adds an integer and string ("5" + 2) causing a

TypeError. Use AI to resolve the bug.

Bug: TypeError due to mixing string and integer

def add_five(value): return value + 5

print(add_five("10")) Requirements:

• Ask AI for two solutions: type casting and string concatenation.

• Validate with 3 assert test cases.

```
#In the below code there is a TypeError due to mixing string and integer. You Fix the error and add that numbers.
# def add_five(value):
# return value + 5
# print(add_five("10"))
# The error in the above code is that it tries to add an integer (5) to a string ("10"), which will raise a TypeError.
# The corrected version of the code is as follows:
def add_five(value):
    try:
        return int(value) + 5
    except ValueError:
        return f"Error: The value '{value}' is not a valid number. Please provide a numeric value."
print(add_five("10")) #This will now return 15
# Explanation: In the original code, the function `add_five` attempts to add an integer (5) to a string ("10"),
# which is not allowed in Python and results in a TypeError.
# In the corrected version, we convert the input `value` to an integer using `int(value)`.
```

```
PROBLEMS 26   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

PS C:\Users\Sathwik\OneDrive\Desktop\AI-Lab> & C:/Users/Sathwik/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Sathwik/OneDr
py
15
PS C:\Users\Sathwik\OneDrive\Desktop\AI-Lab>
```