**AI ASSISTED CODING – ASSIGNMENT 7.1**

**Task Description #1** (Syntax Errors – Missing Parentheses in Print Statement)  
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.

**Prompt** –

Missing parentheses in print statement

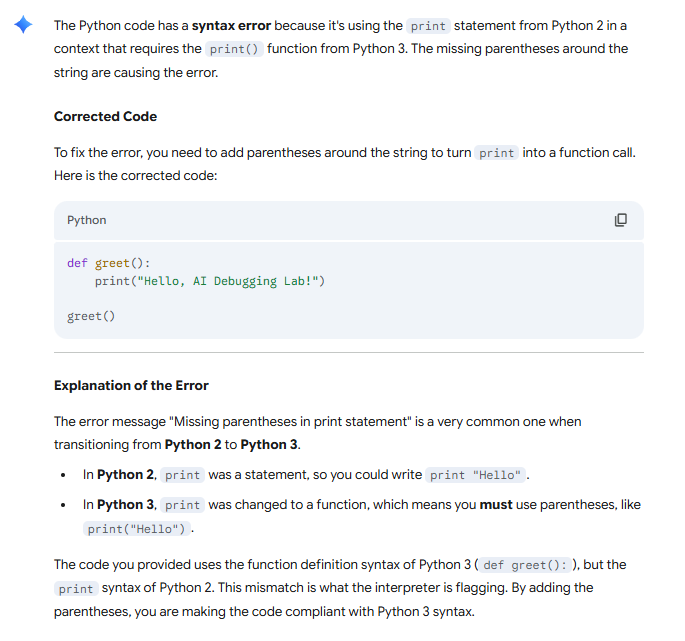
def greet():

print "Hello, AI Debugging Lab!"

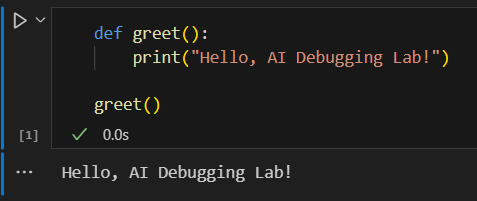
greet()

detect and fix the syntax error.

**Code** –



**Output** –



**Task Description #2** (Logic Error – Incorrect Condition in an If Statement)  
Supply a function where an if-condition mistakenly uses = instead of ==. Let AI identify and fix the issue.

**Prompt** –

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

def check\_number(n):

if n = 10:

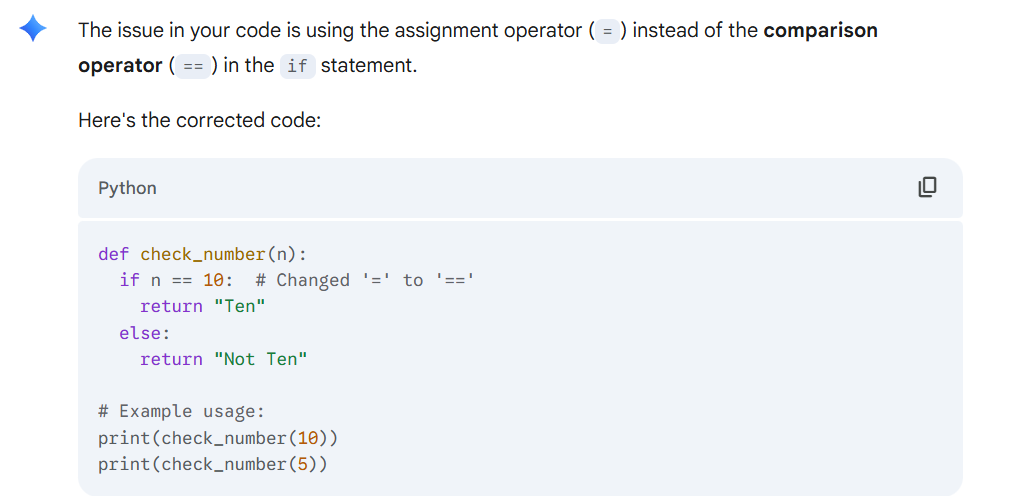
return "Ten"

else:

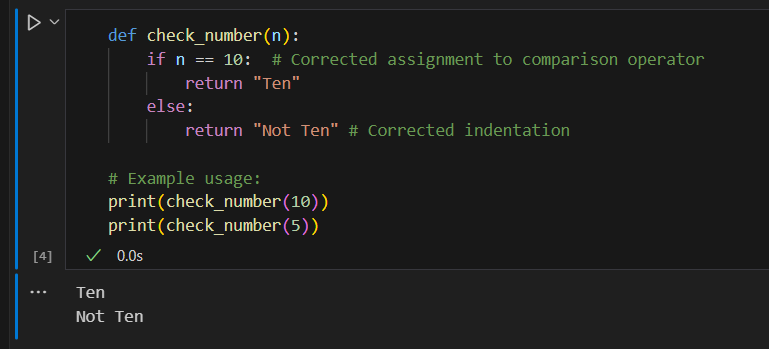
return "Not Ten"

fix the code.

**Code** –



**Output** –



**Task Description #3** (Runtime Error – File Not Found)  
Provide code that attempts to open a non-existent file and crashes. Use AI to apply safe error handling

**Prompt** –

Program crashes if file is missing

def read\_file(filename):

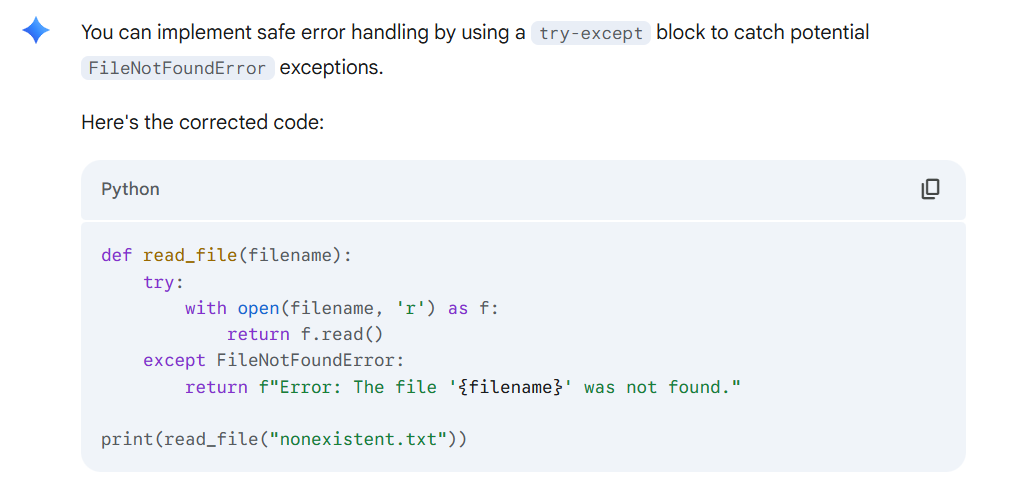
with open(filename, 'r') as f:

return f.read()

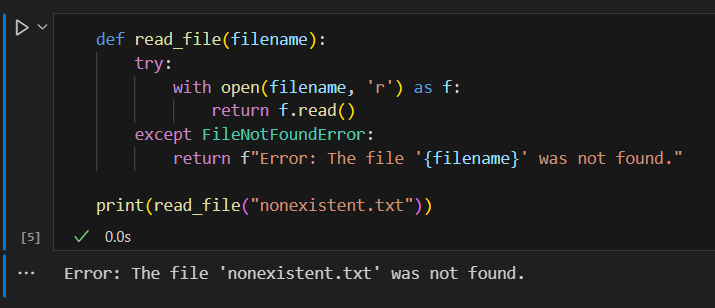
print(read\_file("nonexistent.txt"))

apply safe error handling

**Code** –



**Output** –



**Task Description #4** (AttributeError – Calling a Non-Existent Method)  
Give a class where a non-existent method is called (e.g., obj.undefined\_method()). Use AI to debug and fix.

**Prompt** –

class Car:

def start(self):

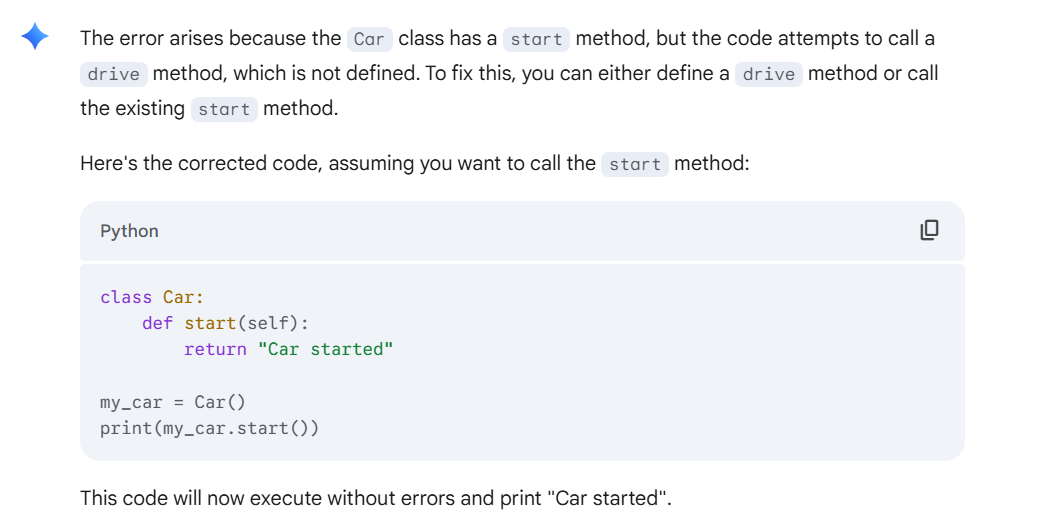
return "Car started"

my\_car = Car()

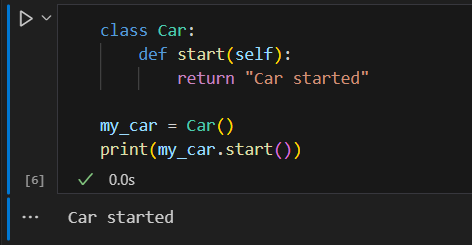
print(my\_car.drive()) # drive() is not defined

debug and fix.

**Code** –



**Output** –



**Task Description #5** (TypeError – Mixing Strings and Integers in Addition)  
Provide code that adds an integer and string ("5" + 2) causing a TypeError. Use AI to resolve the bug.

**Prompt** –

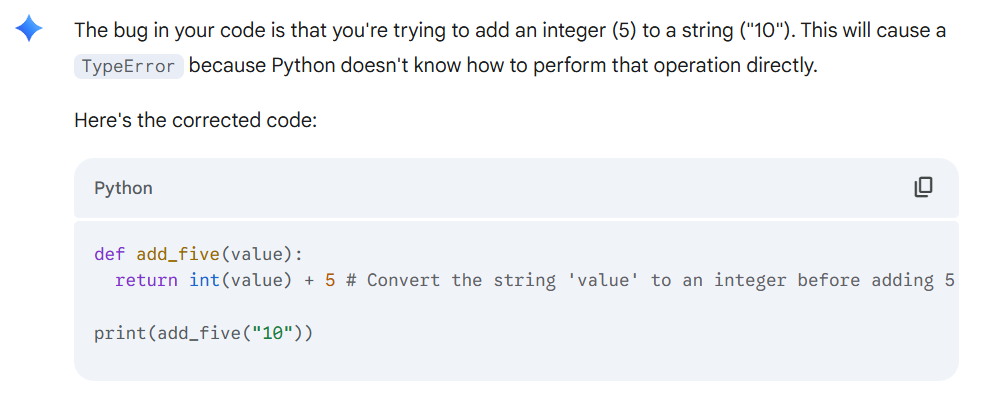
def add\_five(value):

return value + 5

print(add\_five("10"))

resolve the bug.

**Code** –



**Output** –

