

Assingment-7.1

Name: B. Shravya

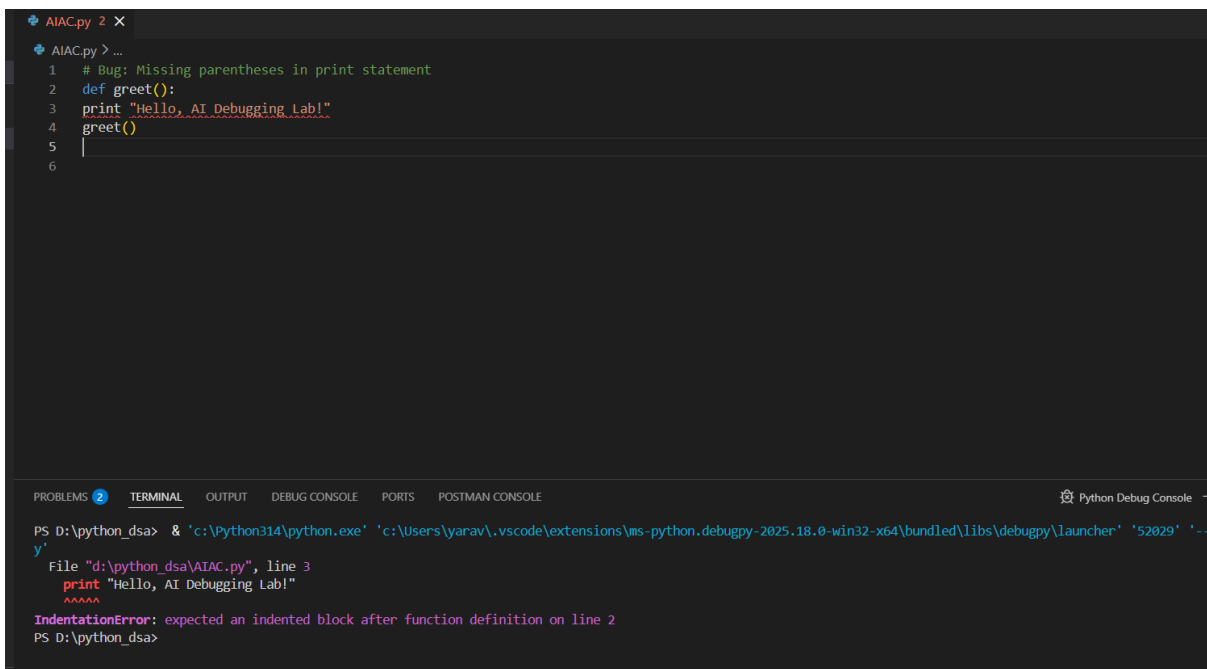
Hall Ticket No:2303A51492

Batch-08

Lab 7: Error Debugging with AI: Systematic approaches to finding and fixing bugs

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.



```
AIAC.py 2 X
AIAC.py > ...
1 # Bug: Missing parentheses in print statement
2 def greet():
3 print "Hello, AI Debugging Lab!"
4 greet()
5
6

PROBLEMS 2 TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console
PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '52029' '--y'
File "d:\python_dsa\AIAC.py", line 3
    print "Hello, AI Debugging Lab!"
    ^^^^^
IndentationError: expected an indented block after function definition on line 2
PS D:\python_dsa>
```

Corrected Code

```
AIAC.py > ...
1 # Bug: Missing parentheses in print statement
2 def greet():
3 | print('Hello, AI Debugging Lab!')
4 greet()
5
6

PROBLEMS 0 TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console + - [ ] [ ]

PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '53869' '--' 'd:\python_dsa\AIAC.py'
Hello, AI Debugging Lab!
PS D:\python_dsa>
```

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.

```
AIAC.py 1 X
AIAC.py > ...
1 # Bug: Using assignment (=) instead of comparison (==)
2 def check_number(n):
3 | if n = 10:
4 | | return 'Ten'
5 | else:
6 | | return 'Not Ten'
7
8
9

PROBLEMS 1 TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console + - [ ] [ ]

PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '53869' '--' 'd:\python_dsa\AIAC.py'
Hello, AI Debugging Lab!
PS D:\python_dsa> ^C
PS D:\python_dsa>
PS D:\python_dsa> cd 'd:\python_dsa'; & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '53869' '--' 'd:\python_dsa\AIAC.py'
File "d:\python_dsa\AIAC.py", line 3
    if n = 10:
        ^^^^^
SyntaxError: invalid syntax. Maybe you meant '==' or ':=' instead of '='?
PS D:\python_dsa>
```

Corrected Code

```
AIAC.py X
AIAC.py > ...
1 # Fixed Version: Using comparison operator (==)
2 def check_number(n):
3     if n == 10:
4         return "Ten"
5     else:
6         return "Not Ten"
7
8
9 # Test Cases
10 assert check_number(10) == "Ten"
11 assert check_number(5) == "Not Ten"
12 assert check_number(-10) == "Not Ten"
13
14 print("All tests passed successfully!")
15
16
17

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console
PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '57860' 'y'
All tests passed successfully!
PS D:\python_dsa>
```


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.

```
AIAC.py X
AIAC.py > ...
1 # Bug: Program crashes if file is missing
2 def read_file(filename):
3     with open(filename, 'r') as f:
4         return f.read()
5
6 print(read_file("nonexistent.txt"))
7
8
9

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console
All tests passed successfully!
PS D:\python_dsa> ^C
PS D:\python_dsa>
PS D:\python_dsa> d:: cd 'd:\python_dsa'; & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '57860' 'y'
Traceback (most recent call last):
  File "d:\python_dsa\AIAC.py", line 6, in <module>
    print(read_file("nonexistent.txt"))
          ~~~~~~^~~~~~
  File "d:\python_dsa\AIAC.py", line 3, in read_file
    with open(filename, 'r') as f:
          ~~~~~~^~~~~~
FileNotFoundError: [Errno 2] No such file or directory: 'nonexistent.txt'
```

Safe Version with Exception Handling



The screenshot shows a Python IDE with a file named `AIAC.py`. The code defines a `read_file` function that takes a `filename` parameter. It attempts to open the file in read mode ('r') and return its content. If the file is not found, it raises a `FileNotFoundError`. If there is an `OSError`, it returns an error message indicating an invalid file path or inability to access the file.

Below the function definition, there are test scenarios commented out with green checkmarks. These tests verify that the `read_file` function works correctly for a file that exists, a file that does not exist, and an invalid file path.

```

1  def read_file(filename):
2      try:
3          with open(filename, 'r') as f:
4              return f.read()
5      except FileNotFoundError:
6          return "Error: File not found."
7      except OSError:
8          return "Error: Invalid file path or unable to access file."
9
10
11 # -----
12 # [x] Test Scenarios
13 # -----
14
15 # [x] File Exists
16 with open("testfile.txt", "w") as f:
17     f.write("Sample content")
18
19 assert read_file("testfile.txt") == "Sample content"
20
21 # [x] File Missing
22 assert read_file("nonexistent.txt") == "Error: File not found."
23
24 # [x] Invalid Path
25 assert read_file("invalid/path/test.txt") == "Error: Invalid file path or unable to access file."
26
27 print("All test cases passed successfully!")
28

```

The IDE interface includes a **TERMINAL** panel at the bottom, which shows the command prompt output. The command `PS D:\python_dsa> & 'c:\python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '49781' '-' 'd:\python_dsa\AIAC.py'` has been executed, and the output is `All test cases passed successfully!`.

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.

The screenshot shows a code editor window titled "AIAc.py". The code defines a class named "Car" with a method "start(self)" that returns the string "car started". Below the class definition, there is an instance creation "my_car = Car()" followed by a call to "print(my_car.drive())". A comment next to the print statement says "# drive() is not defined". The bottom status bar indicates a "Python Debug Console" is active. The terminal output shows a traceback from the file "d:\python_dsa\AIAc.py" at line 7, where the error "AttributeError: 'Car' object has no attribute 'drive'" occurred. The prompt "PS D:\python_dsa>" is visible at the bottom.

```
AIAc.py > ...  
1 # Bug: Calling an undefined method  
2 class Car:  
3     def start(self):  
4         return "car started"  
5  
6 my_car = Car()  
7 print(my_car.drive()) # drive() is not defined  
8  
9  
10  
11  
12
```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE

Python Debug Console + - [] [X] [Y] [Z]

y'
Traceback (most recent call last):
File "d:\python_dsa\AIAc.py", line 7, in <module>
 print(my_car.drive()) # drive() is not defined
~~~~~  
AttributeError: 'Car' object has no attribute 'drive'  
PS D:\python\_dsa>

### Corrected Code

```
AIAC.py X
AIAC.py > ...
6 | | | return "Car is driving"
7 |
8 |
9 | # Create object
10 | my_car = Car()
11 |
12 | # Test calls
13 | print(my_car.drive())
14 |
15 |
16 | # Assert Test Cases
17 | assert my_car.start() == "Car started"
18 | assert my_car.drive() == "Car is driving"
19 | assert isinstance(my_car, Car)
20 |
21 | print("All tests passed successfully!")
22 |
23 |
24 |
25 |
26 |
27 |

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE
Python Debug Console + - [ ] ... | [ ] X

PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '50613' '--' 'd:\python_dsa\AIAC.py'
Car is driving
All tests passed successfully!
PS D:\python_dsa>
```

## 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

```
AIAC.py X
AIAC.py > ...
1 | # Bug: TypeError due to mixing string and integer
2 | def add_five(value):
3 |     return value + 5
4 |
5 | print(add_five("10"))
6 |
7 |
8 |
9 |
10 |
11 |
12 |

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE
Python Debug Console + - [ ] ... | [ ] X

PS D:\python_dsa> ^C
PS D:\python_dsa> d;; cd 'd:\python_dsa'; & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '62882' -- 'd:\python_dsa\AIAC.py'
Traceback (most recent call last):
  File "d:\python_dsa\AIAC.py", line 5, in <module>
    print(add_five("10"))
           ~~~~~~
 File "d:\python_dsa\AIAC.py", line 3, in add_five
 return value + 5
           ~~~~~~
TypeError: can only concatenate str (not "int") to str
```

## Solution 1: Type Casting (Convert to Integer)

```
AIAC.py X
AIAC.py > ...
1 # Solution 1: Type Casting
2 def add_five(value):
3     return int(value) + 5
4
5
6 # Test Cases
7 assert add_five("10") == 15
8 assert add_five(20) == 25
9 assert add_five("0") == 5
10
11 print("Solution 1 tests passed!")
12
13
14
15
16
17
18
19

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console + - [ ]
PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '59738' '--' 'd:\python_dsa\python_dsa.py'
Solution 1 tests passed!
PS D:\python_dsa>
```

## Solution 2: String Concatenation

```
AIAC.py X
AIAC.py > ...
1 # Solution 2: String Concatenation
2 def add_five(value):
3     return str(value) + "5"
4
5
6 # Test Cases
7 assert add_five("10") == "105"
8 assert add_five(20) == "205"
9 assert add_five("") == "5"
10
11 print("Solution 2 tests passed!")
12
13
14
15
16
17
18
19
20

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE PORTS POSTMAN CONSOLE Python Debug Console + - [ ]
PS D:\python_dsa> & 'c:\Python314\python.exe' 'c:\Users\yarav\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '63944' '--' 'd:\python_dsa\python_dsa.py'
Solution 2 tests passed!
PS D:\python_dsa>
```