

# AI ASSISTED CODING

## LAB ASSIGNMENT-7

Name: M.Harshith

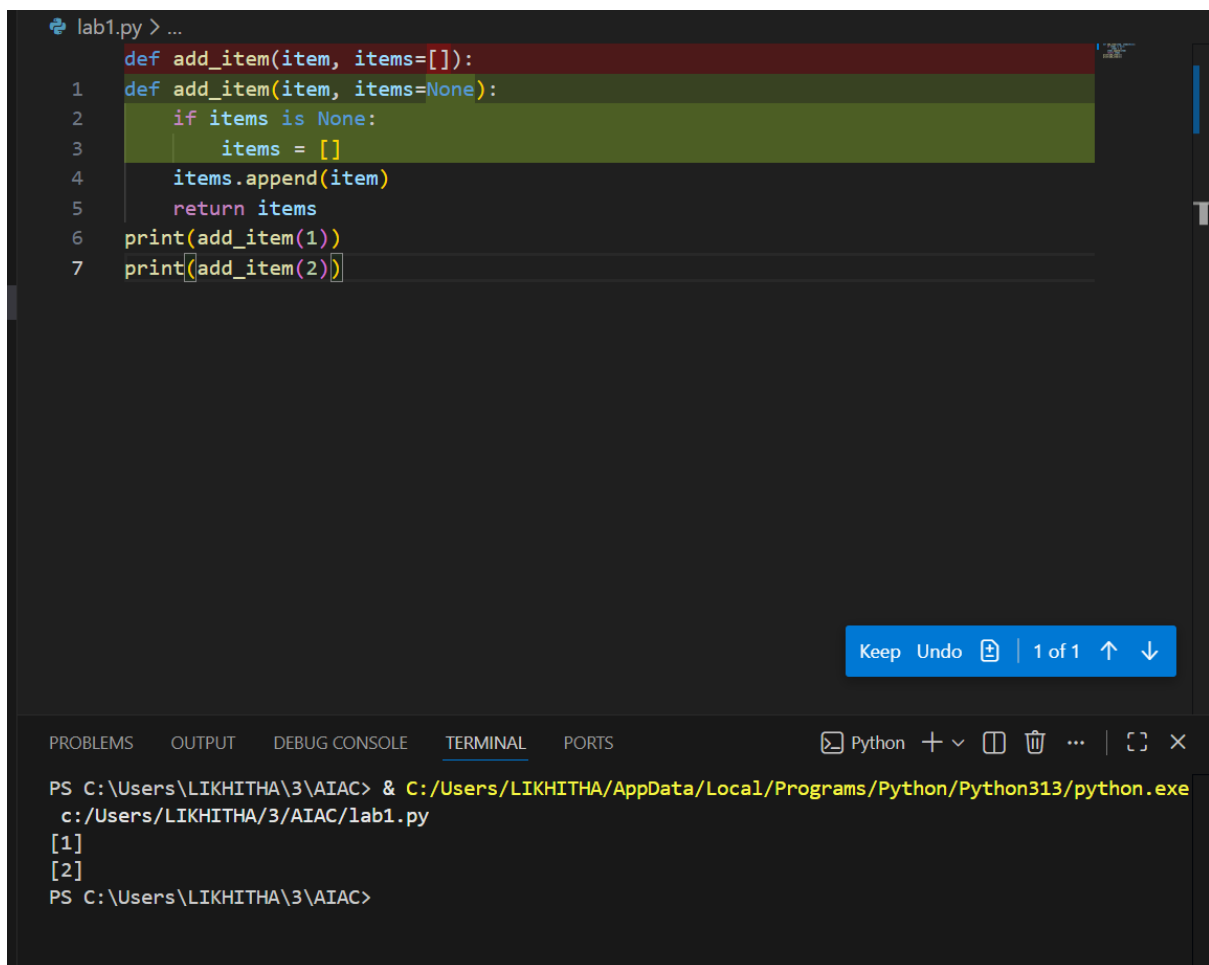
HT.NO: 2303A51444

Batch: 21

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

**Question-1:** Analyse given code where a mutable default argument causes unexpected behaviour. Use AI to fix it.

**Screenshot:**



```
lab1.py > ...
def add_item(item, items=[]):
1 def add_item(item, items=None):
2     if items is None:
3         items = []
4     items.append(item)
5     return items
6 print(add_item(1))
7 print(add_item(2))
```

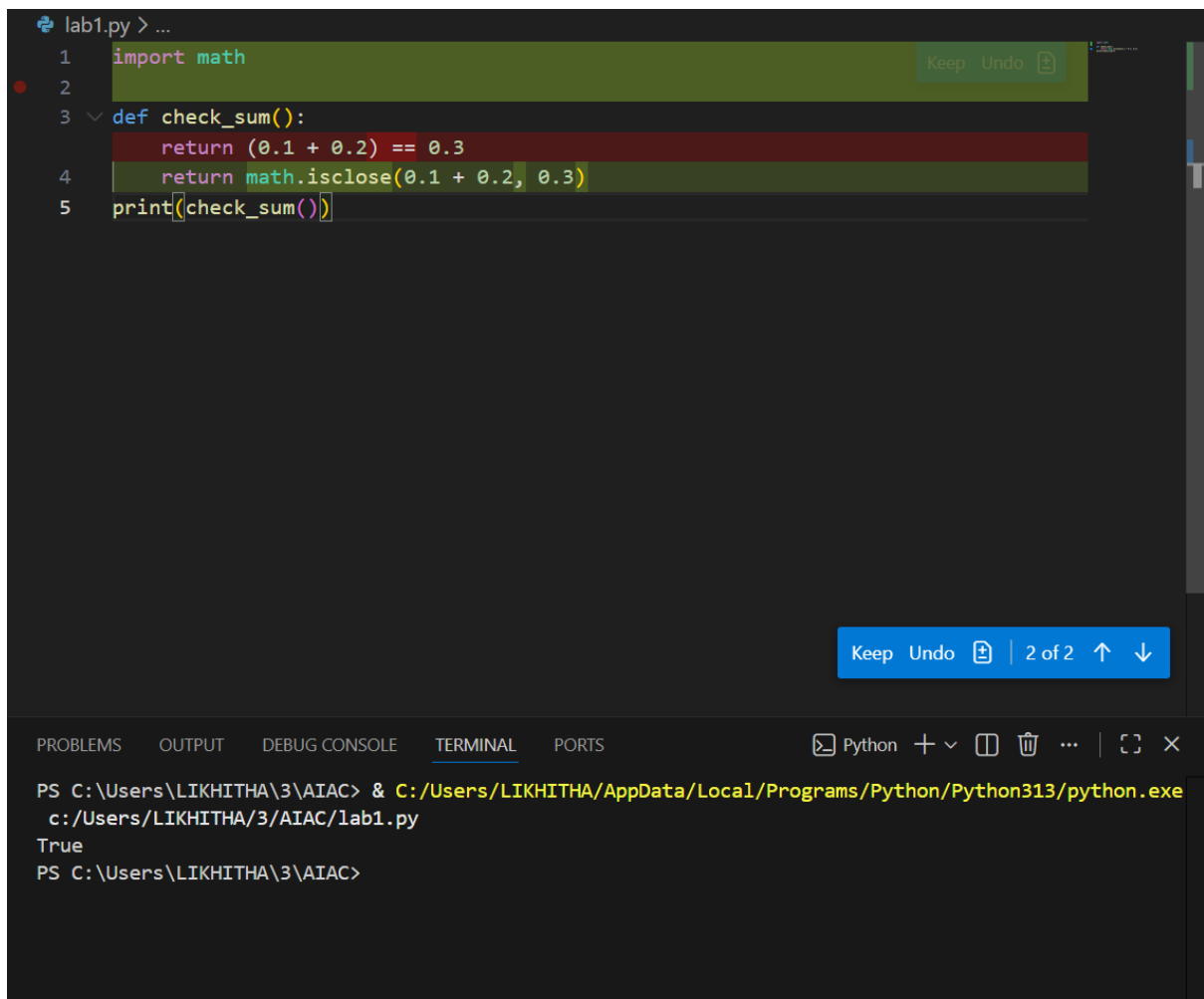
Keep Undo | 1 of 1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + -

```
PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe
c:/Users/LIKHITHA/3/AIAC/lab1.py
[1]
[2]
PS C:\Users\LIKHITHA\3\AIAC>
```

**Question-2:** Analyse given code where floating-point comparison fails. Use AI to correct with tolerance.

Screenshot:



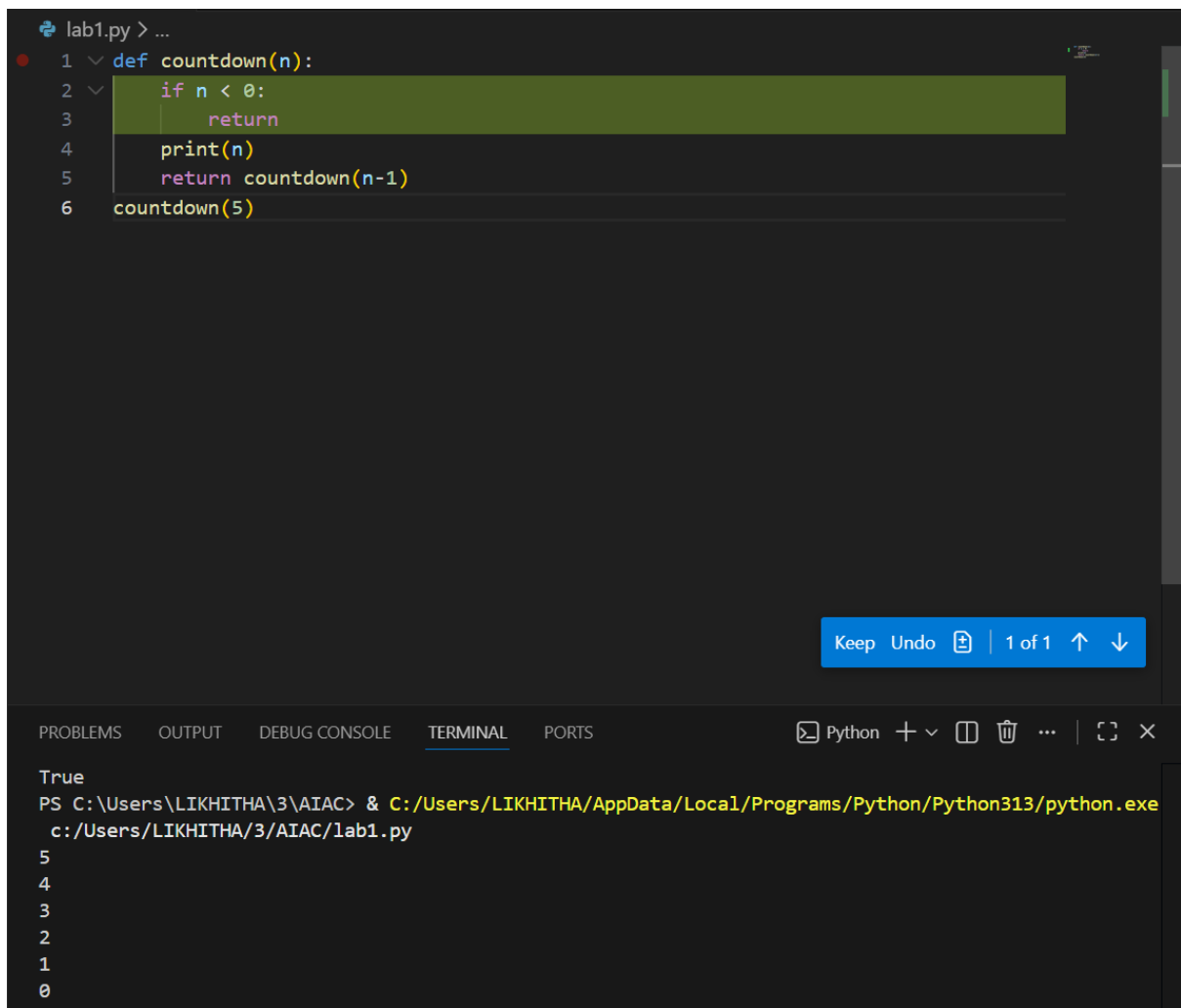
The screenshot shows a code editor with a file named `lab1.py`. The code defines a function `check_sum()` that returns `(0.1 + 0.2) == 0.3`. Below this, it returns `math.isclose(0.1 + 0.2, 0.3)` and prints the result of `check_sum()`. The terminal output shows the command to run the script, which outputs `True`.

```
lab1.py > ...
1 import math
2
3 def check_sum():
4     return (0.1 + 0.2) == 0.3
5     return math.isclose(0.1 + 0.2, 0.3)
6 print(check_sum())
```

PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LIKHITHA/3/AIAC/lab1.py  
True  
PS C:\Users\LIKHITHA\3\AIAC>

**Question-3:** Analyse given code where recursion runs infinitely due to missing base case. Use AI to fix. [Generating using colab:](#)

Screenshot:



The screenshot shows a code editor with a Python file named `lab1.py`. The code defines a recursive function `countdown(n)` and calls it with `countdown(5)`. The function has a base case `if n < 0: return` and a recursive step `print(n)` followed by `return countdown(n-1)`. The terminal output shows the function returning `True` and printing the numbers 5, 4, 3, 2, 1, and 0 in descending order.

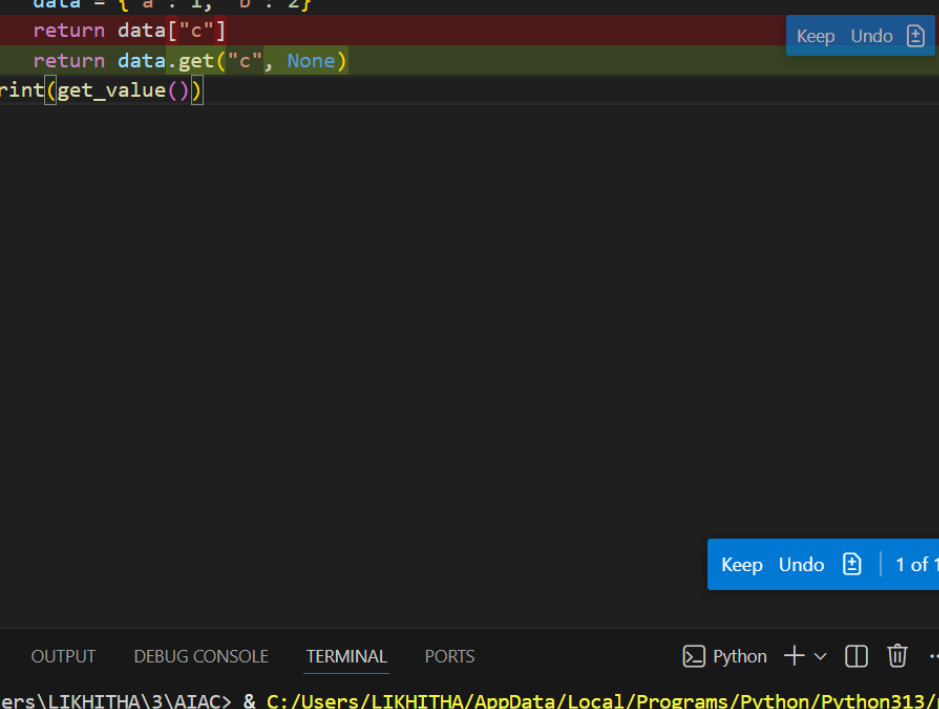
```
lab1.py > ...
1 def countdown(n):
2     if n < 0:
3         return
4     print(n)
5     return countdown(n-1)
6 countdown(5)
```

Keep Undo | 1 of 1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + -

True  
PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe  
c:/Users/LIKHITHA/3/AIAC/lab1.py  
5  
4  
3  
2  
1  
0

Screenshot:



The screenshot displays a code editor with a Python script named `lab1.py`. The script defines a function `get_value()` that returns the value for key 'c' from a dictionary, or `None` if the key is not present. The dictionary `data` contains keys 'a' and 'b'. The function is then called and its result is printed.

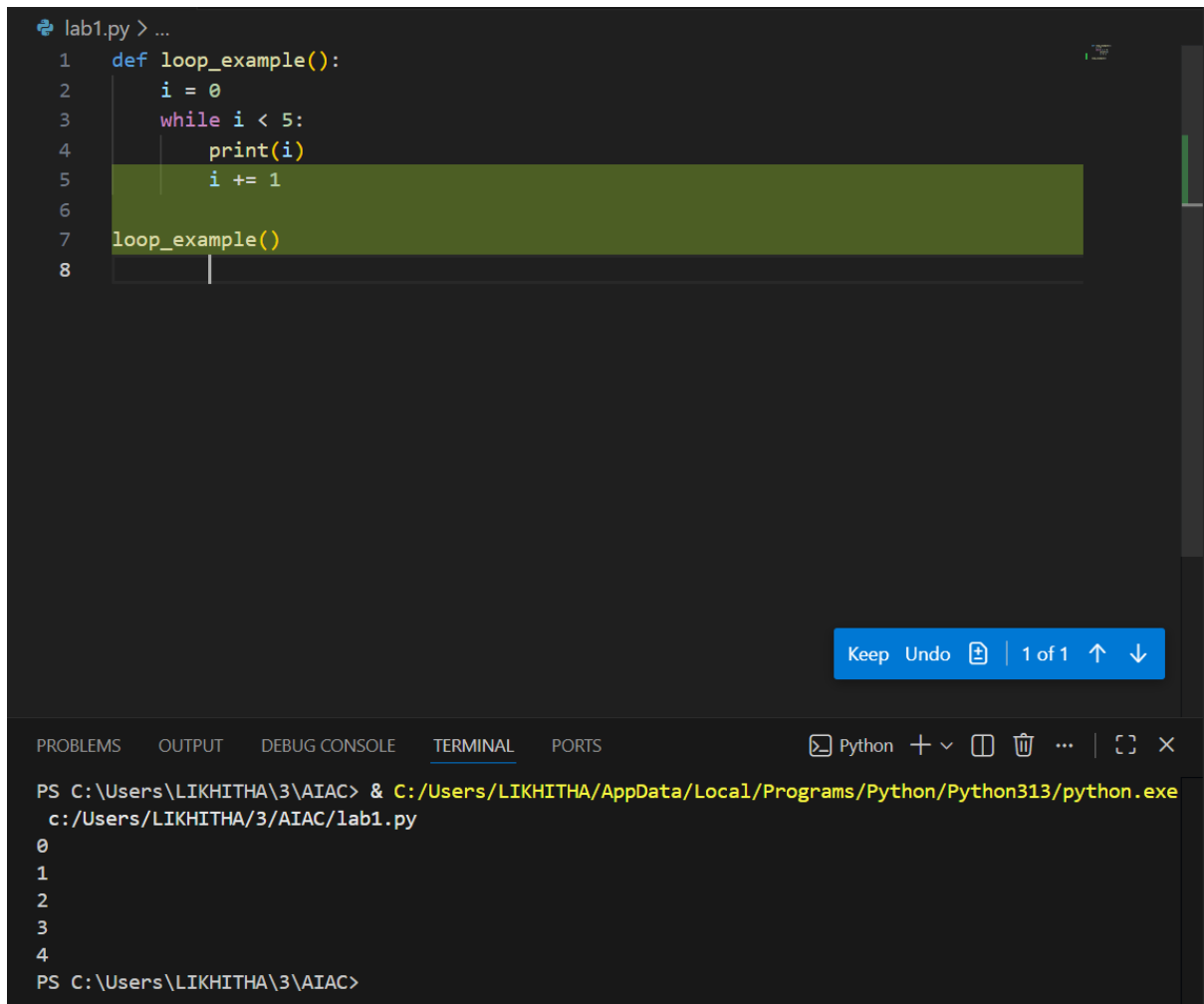
```
1 def get_value():
2     data = {"a": 1, "b": 2}
3     return data["c"]
4     return data.get("c", None)
5 print(get_value())
```

The terminal output shows the execution of the script, resulting in `None` being printed.

```
PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LIKHITHA/3/AIAC/lab1.py
None
PS C:\Users\LIKHITHA\3\AIAC>
```

**Question-5:** Analyse given code where loop never ends. Use AI to detect and fix it.

Screenshot:



The screenshot shows a code editor with a Python file named `lab1.py`. The code defines a function `loop_example()` that initializes `i = 0` and enters a `while i < 5:` loop. Inside the loop, it prints `i` and increments `i` by 1. However, the increment is `i += 1`, which is correct, but the loop condition is `i < 5`. The code then calls `loop_example()`. The terminal output shows the execution of the code, printing the numbers 0 through 4, which indicates the loop is working correctly. The code is as follows:

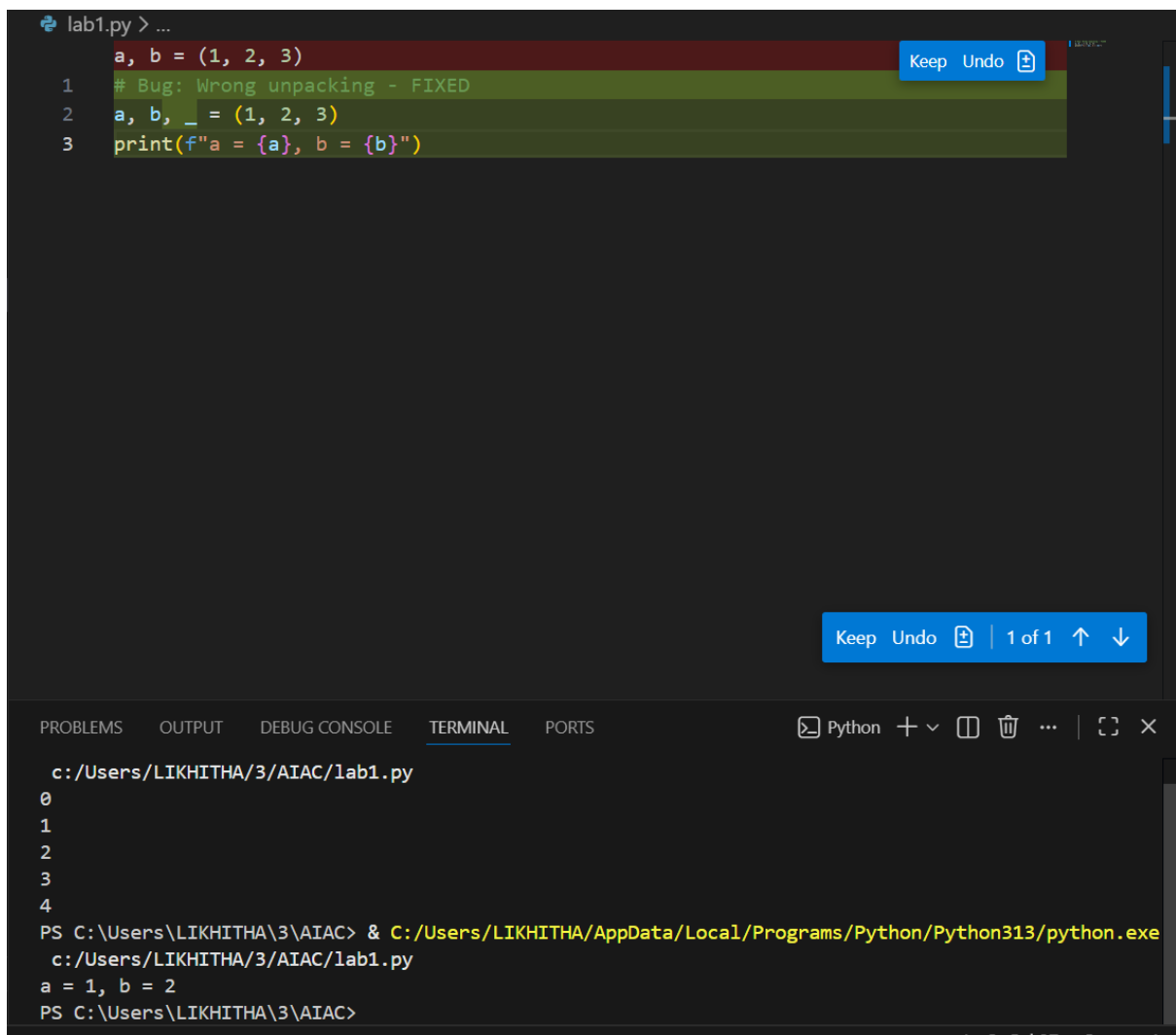
```
1 def loop_example():
2     i = 0
3     while i < 5:
4         print(i)
5         i += 1
6
7 loop_example()
8
```

The terminal output shows the execution of the code, printing the numbers 0 through 4, which indicates the loop is working correctly. The code is as follows:

```
PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe
c:/Users/LIKHITHA/3/AIAC/lab1.py
0
1
2
3
4
PS C:\Users\LIKHITHA\3\AIAC>
```

**Question-6:** Analyse given code where tuple unpacking fails. Use AI to fix it.

Screenshot:



The screenshot shows a code editor with a Python file named `lab1.py`. The code contains three lines:

```
a, b = (1, 2, 3)
1 # Bug: Wrong unpacking - FIXED
2 a, b, _ = (1, 2, 3)
3 print(f"a = {a}, b = {b}")
```

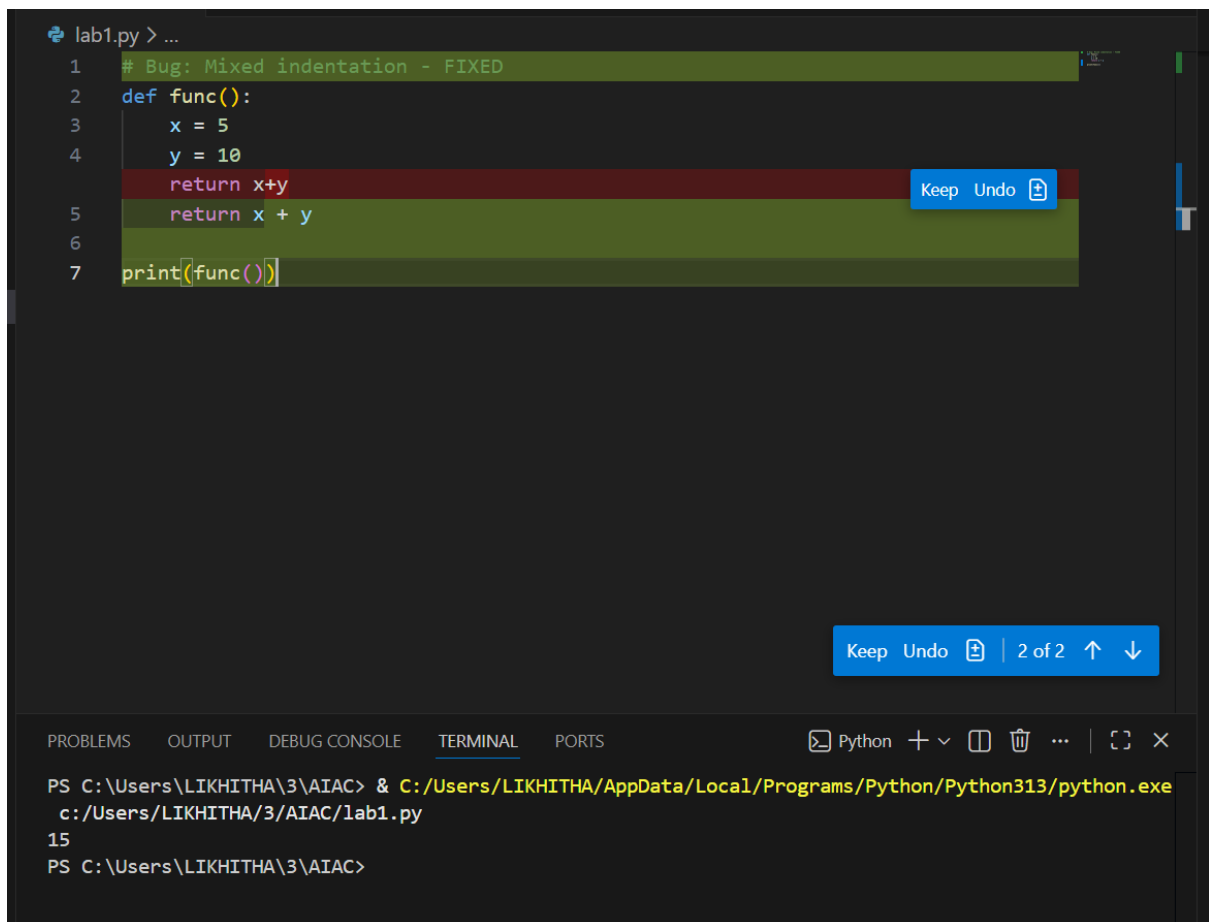
The first line, `a, b = (1, 2, 3)`, is highlighted in red, indicating an error. A blue button with the text "Keep Undo" and a trash icon is visible next to it. The second line, `a, b, _ = (1, 2, 3)`, is highlighted in green. The third line, `print(f"a = {a}, b = {b}")`, is highlighted in dark green. A blue button with the text "Keep Undo" and a trash icon is also visible next to the second line.

Below the code editor, the terminal window shows the execution of the script:

```
c:/Users/LIKHITHA/3/AIAC/lab1.py
0
1
2
3
4
PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe
c:/Users/LIKHITHA/3/AIAC/lab1.py
a = 1, b = 2
PS C:\Users\LIKHITHA\3\AIAC>
```

**Question-7:** Analyse given code where mixed indentation breaks execution. Use AI to fix it.

Screenshot:



```
lab1.py > ...
1 # Bug: Mixed indentation - FIXED
2 def func():
3     x = 5
4     y = 10
5     return x+y
6     return x + y
7 print(func())
```

Keep Undo

Keep Undo | 2 of 2 ↑ ↓

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + -

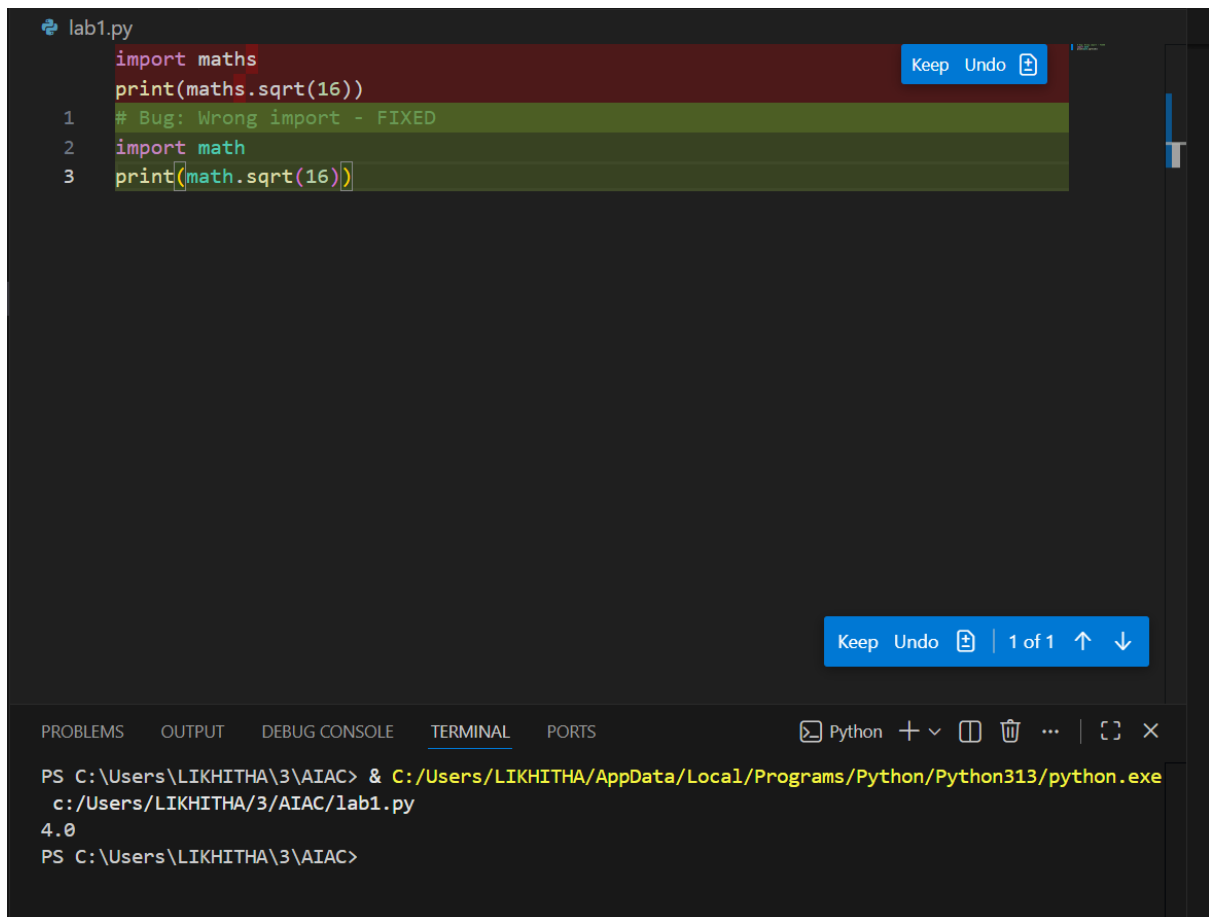
PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LIKHITHA/3/AIAC/lab1.py

15

PS C:\Users\LIKHITHA\3\AIAC>

Question-8: Analyse given code with incorrect import. Use AI to fix.

Screenshot:



```
lab1.py
import maths
print(maths.sqrt(16))
1 # Bug: Wrong import - FIXED
2 import math
3 print(math.sqrt(16))
```

Keep Undo

Keep Undo | 1 of 1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + -

PS C:\Users\LIKHITHA\3\AIAC> & C:/Users/LIKHITHA/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LIKHITHA/3/AIAC/lab1.py

4.0

PS C:\Users\LIKHITHA\3\AIAC>