

Lesson 04 Demo 05

Debugging AI-Generated Code Using GitHub Copilot

Objective: To identify and fix errors in AI-generated code using GitHub Copilot

Tools required: VS Code with GitHub Copilot

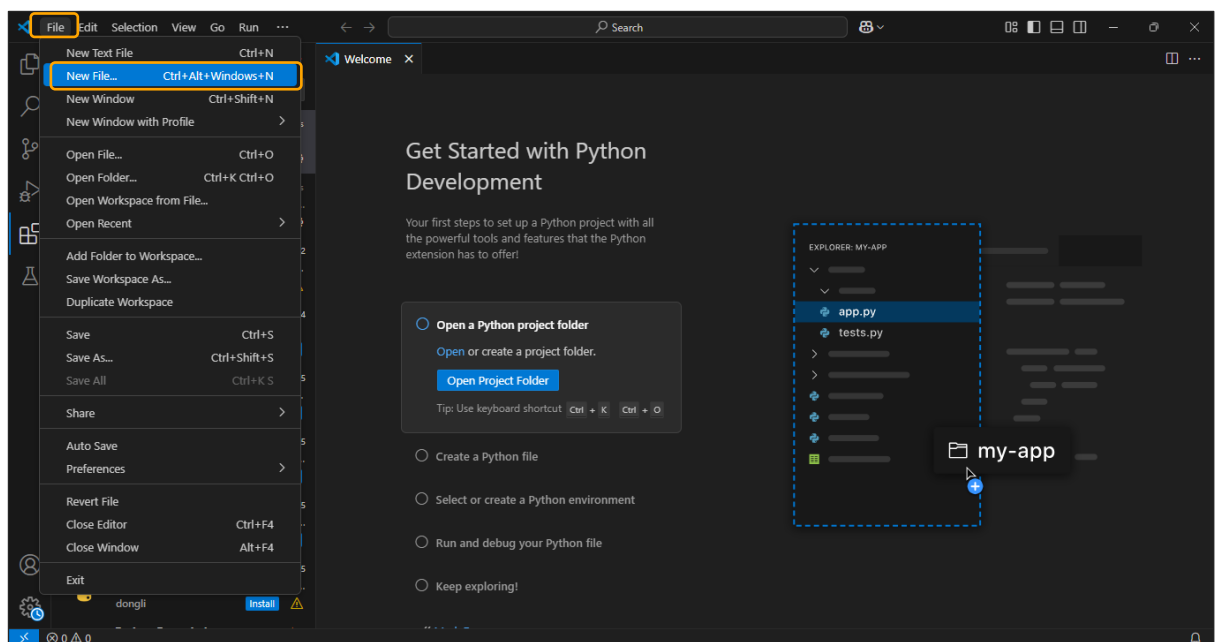
Prerequisites: None

Steps to be followed:

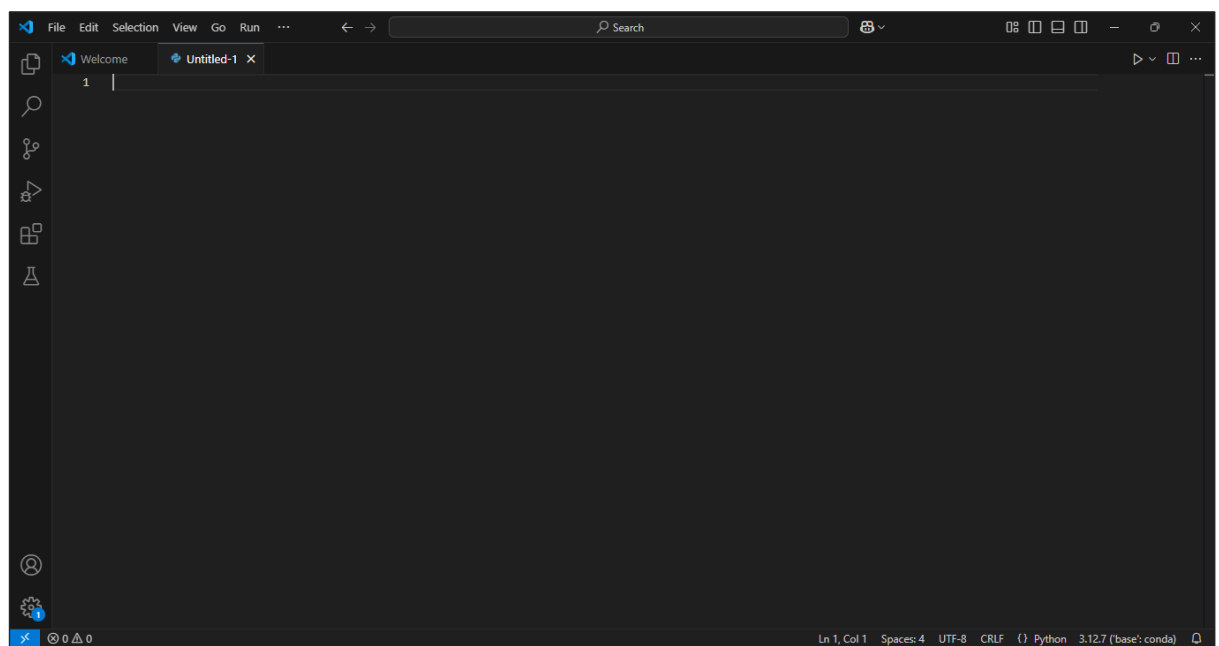
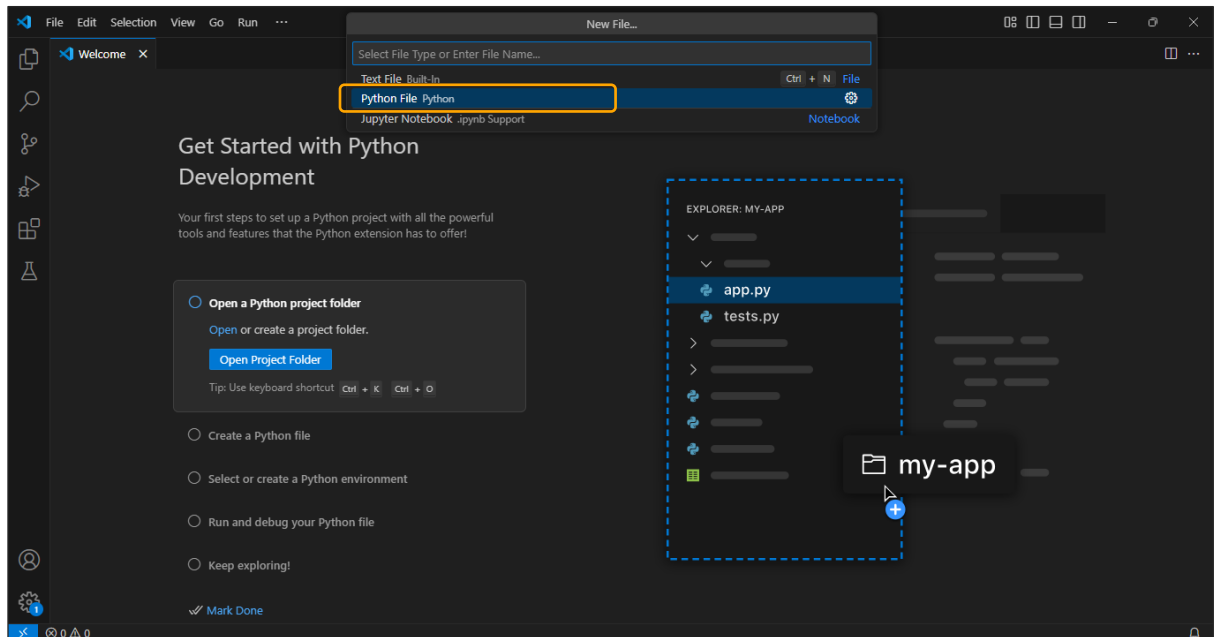
1. Launch VS Code and open a new file
2. Review the erroneous function
3. Identify errors and debug the function with GitHub Copilot
4. Apply the fixes and test the function

Step 1: Launch VS Code and open a new file

1.1 Launch VS Code and click on **File** and then **New File**



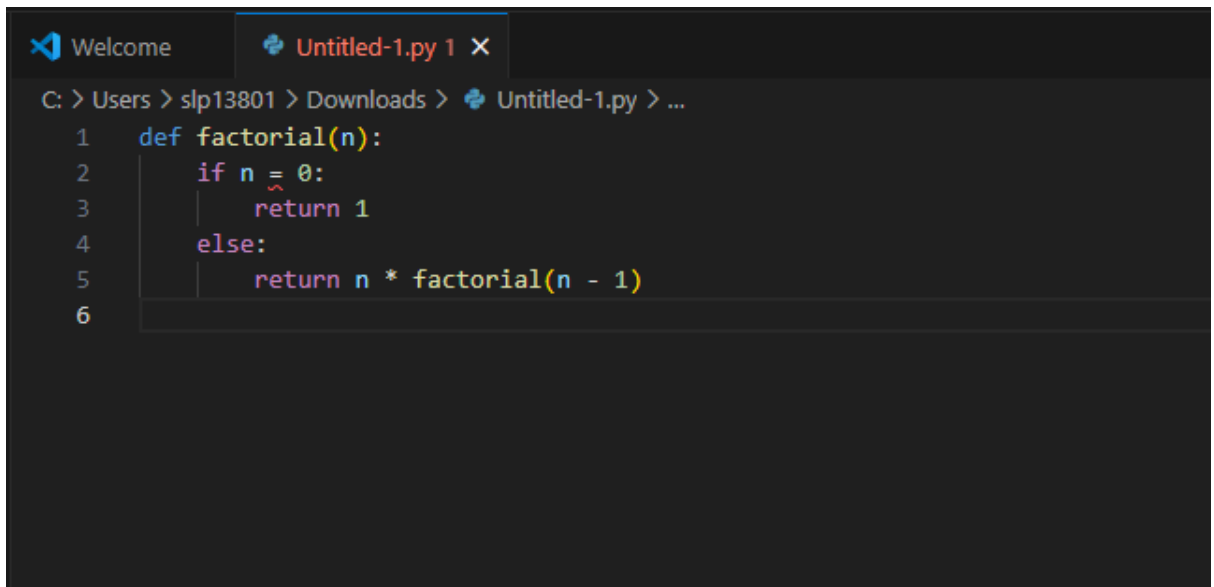
1.2 Select the **Python File** option from the name bar on top and a new Python file named Untitled-1 will open



Step 2: Review the erroneous function

2.1 Copy and paste the following erroneous function, which is supposed to calculate the factorial of a number but contains logical and syntax errors:

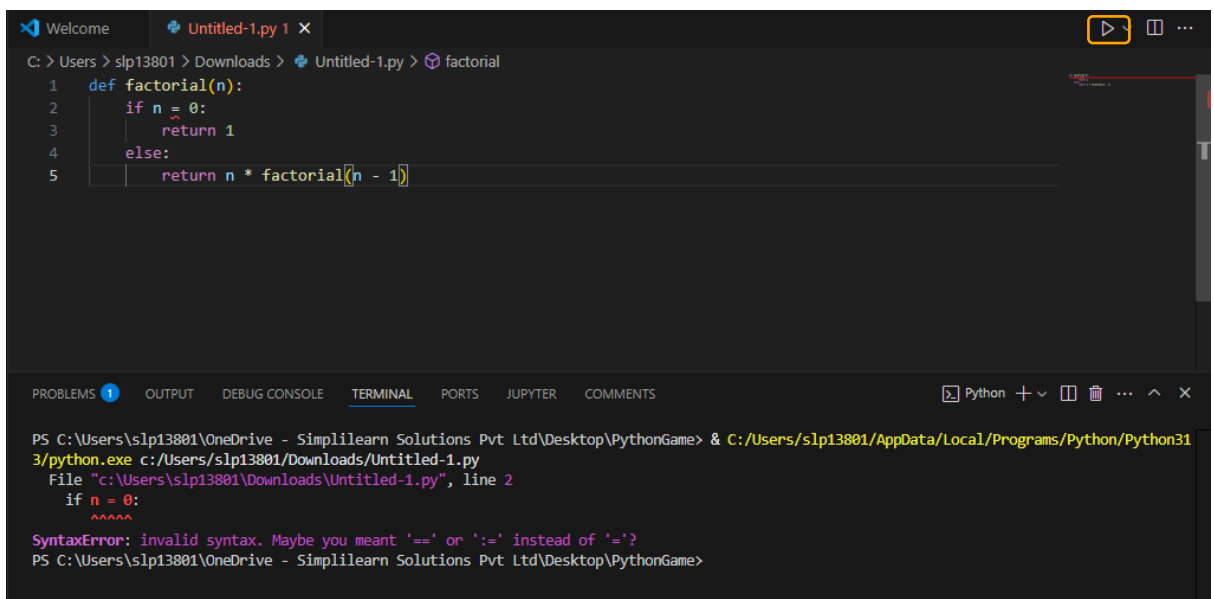
```
def factorial(n):  
    if n = 0:  
        return 1  
    else:  
        return n * factorial(n - 1)
```



A screenshot of a code editor window titled 'Untitled-1.py'. The editor shows the following Python code:

```
1 def factorial(n):  
2     if n = 0:  
3         return 1  
4     else:  
5         return n * factorial(n - 1)  
6
```

2.2 Click the play button to run the function and observe the error messages



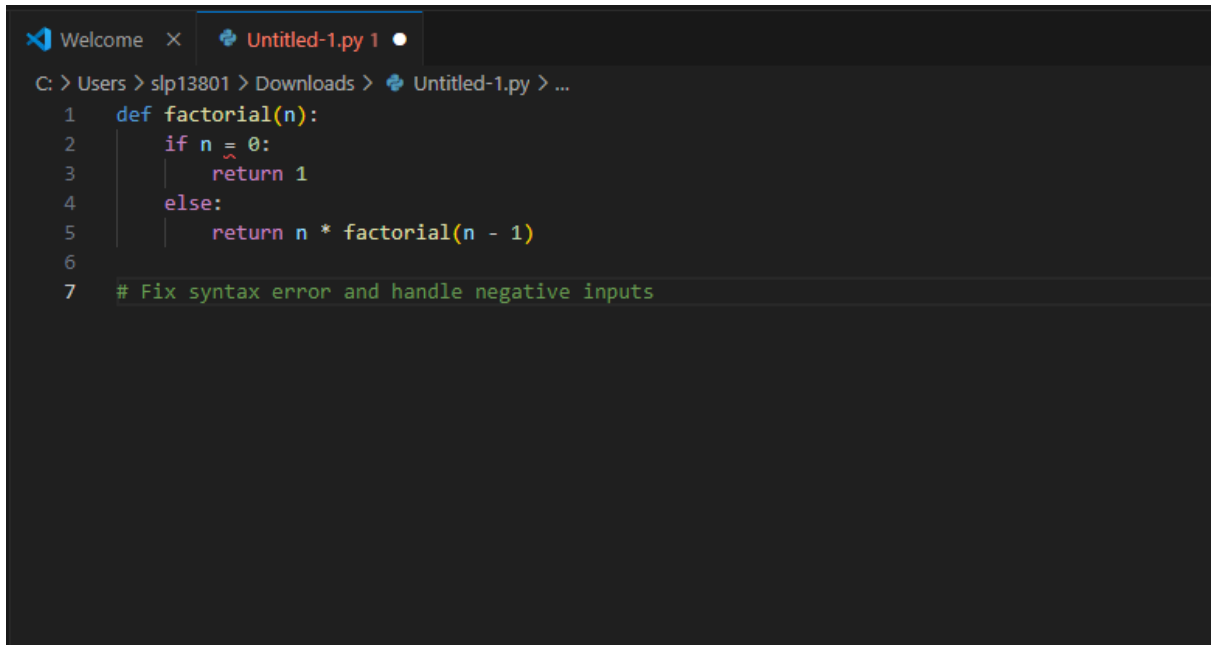
A screenshot of a code editor window titled 'Untitled-1.py' showing the same Python code as before. The code is now highlighted, and a play button (a yellow square with a black triangle) is visible in the top right corner of the editor. Below the code editor, the terminal window is open, displaying the following output:

```
PS C:\Users\slp13801\OneDrive - Simplilearn Solutions Pvt Ltd\Desktop\PythonGame> & C:/Users/slp13801/AppData/Local/Programs/Python/Python313/python.exe c:/Users/slp13801/Downloads/Untitled-1.py  
File "c:\Users\slp13801\Downloads\Untitled-1.py", line 2  
    if n = 0:  
        ^^^^^  
SyntaxError: invalid syntax. Maybe you meant '==' or ':=' instead of '='?  
PS C:\Users\slp13801\OneDrive - Simplilearn Solutions Pvt Ltd\Desktop\PythonGame>
```

Step 3: Identify errors and debug the function with GitHub Copilot

3.1 Place your cursor on the erroneous code and add a comment like:

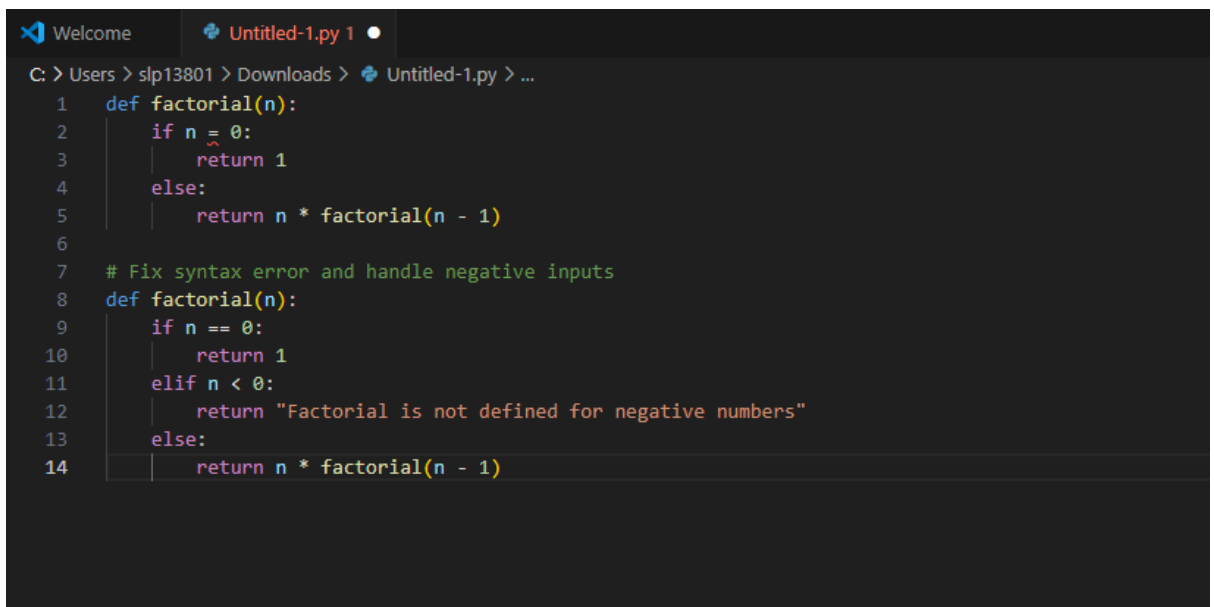
Fix syntax errors and handle negative inputs



The screenshot shows a code editor with a dark theme. The top bar has a 'Welcome' tab and an 'Untitled-1.py 1' tab. The file path is 'C: > Users > slp13801 > Downloads > Untitled-1.py > ...'. The code is as follows:

```
1 def factorial(n):
2     if n == 0:
3         return 1
4     else:
5         return n * factorial(n - 1)
6
7 # Fix syntax error and handle negative inputs
```

3.2 Press enter for GitHub Copilot to suggest an improved version of the function

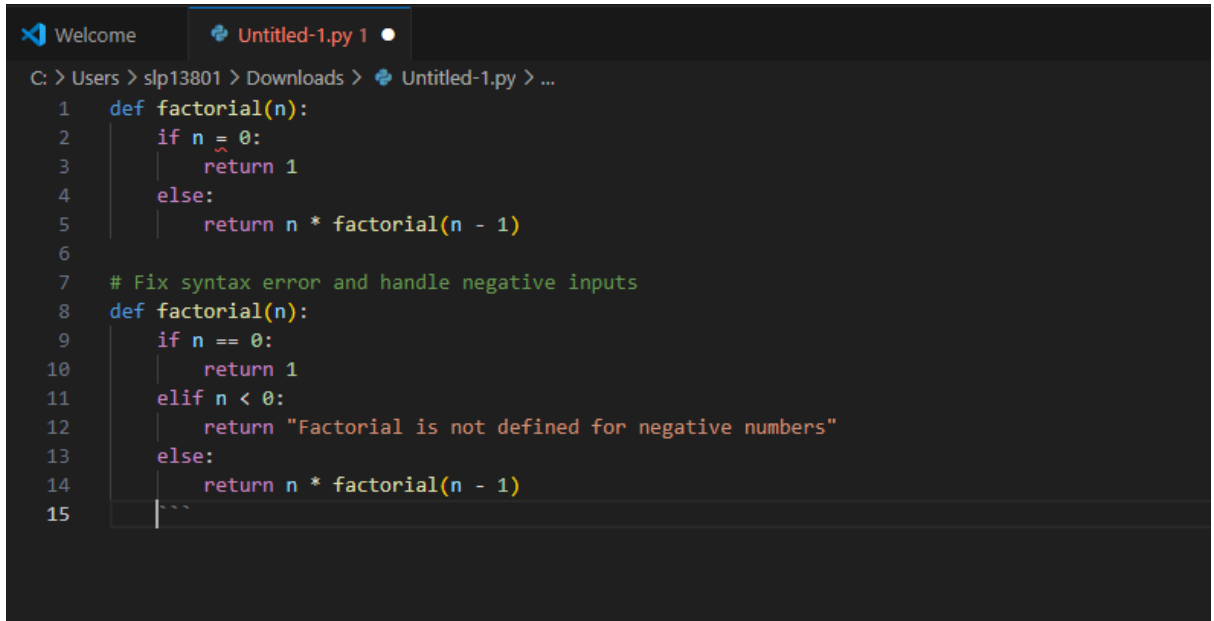


The screenshot shows the same code editor as before, but now with an improved version of the factorial function suggested by GitHub Copilot. The code is as follows:

```
1 def factorial(n):
2     if n == 0:
3         return 1
4     else:
5         return n * factorial(n - 1)
6
7 # Fix syntax error and handle negative inputs
8 def factorial(n):
9     if n == 0:
10        return 1
11    elif n < 0:
12        return "Factorial is not defined for negative numbers"
13    else:
14        return n * factorial(n - 1)
```

Step 4: Apply the fixes and test the function

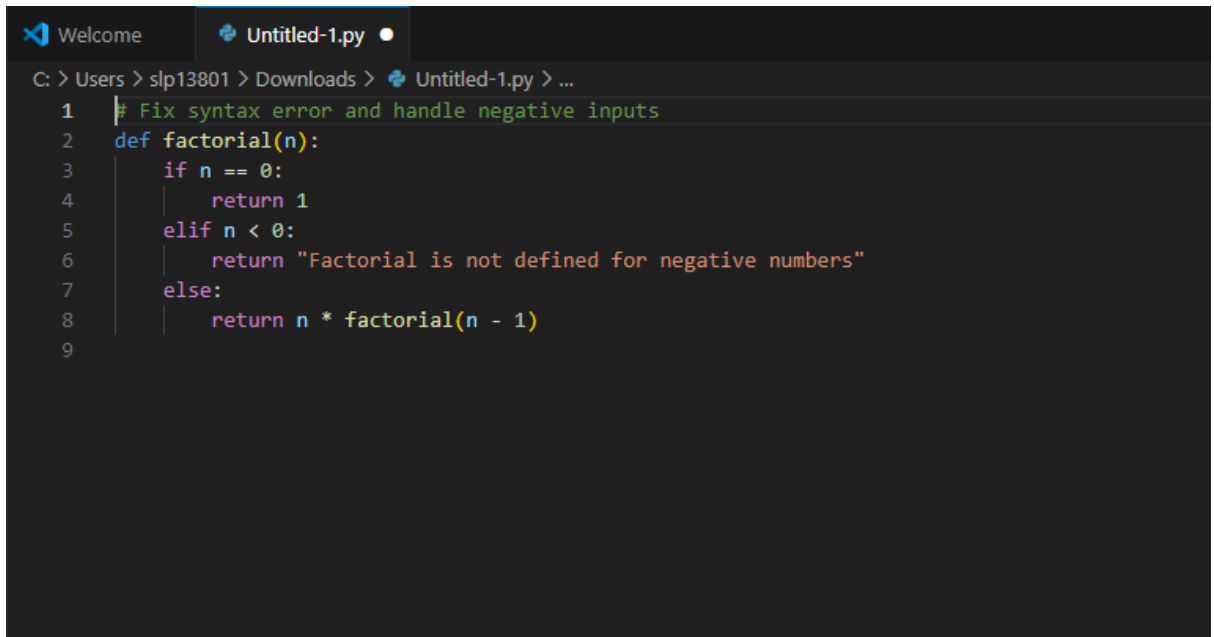
4.1 Press tab to accept Copilot's suggestion, or you can manually rectify the function



The screenshot shows a VS Code editor window with a file named 'Untitled-1.py'. The editor displays a Python function 'factorial(n)' with a Copilot suggestion. The suggestion is a green comment: '# Fix syntax error and handle negative inputs'. The function is defined as follows:

```
1 def factorial(n):
2     if n == 0:
3         return 1
4     else:
5         return n * factorial(n - 1)
6
7 # Fix syntax error and handle negative inputs
8 def factorial(n):
9     if n == 0:
10        return 1
11    elif n < 0:
12        return "Factorial is not defined for negative numbers"
13    else:
14        return n * factorial(n - 1)
15
```

4.2 Remove the erroneous function that you created earlier

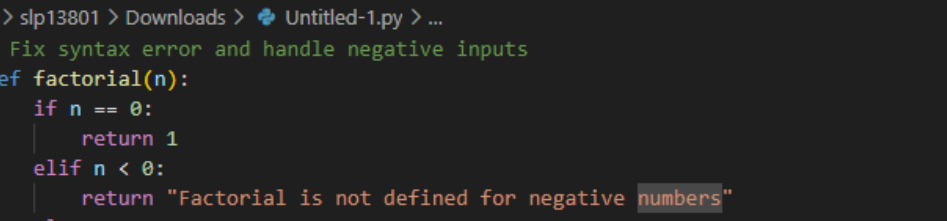


The screenshot shows the same VS Code editor window, but the function definition has been corrected. The erroneous function from the previous step has been removed, and the corrected function is now the only one in the file. The function is defined as follows:

```
1 # Fix syntax error and handle negative inputs
2 def factorial(n):
3     if n == 0:
4         return 1
5     elif n < 0:
6         return "Factorial is not defined for negative numbers"
7     else:
8         return n * factorial(n - 1)
9
```

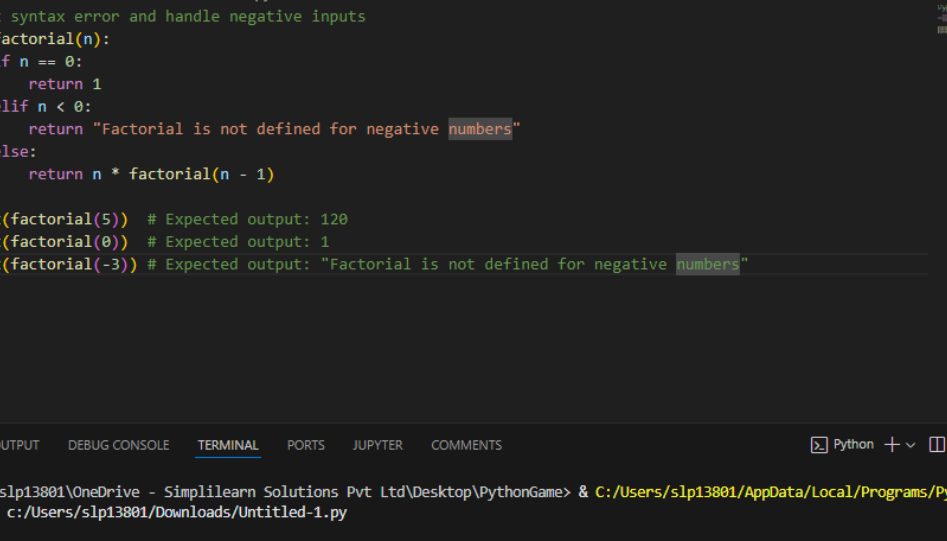
4.3 Add the following test cases to check different inputs:

```
print(factorial(5)) # Expected output: 120
print(factorial(0)) # Expected output: 1
print(factorial(-3)) # Expected output: "Factorial is not defined for negative
numbers"
```



```
1 # Fix syntax error and handle negative inputs
2 def factorial(n):
3     if n == 0:
4         return 1
5     elif n < 0:
6         return "Factorial is not defined for negative numbers"
7     else:
8         return n * factorial(n - 1)
9
10 print(factorial(5)) # Expected output: 120
11 print(factorial(0)) # Expected output: 1
12 print(factorial(-3)) # Expected output: "Factorial is not defined for negative numbers"
13
```

4.4 Click the play button to run the Python file in VS Code and observe the output



The image shows a VS Code editor window with a file named 'Untitled-1.py'. The code defines a 'factorial' function that handles negative inputs by returning a message. The terminal shows the command to run the script and the resulting output, which includes the message for a negative input.

```

C: > Users > slp13801 > Downloads > Untitled-1.py > ...
1  # Fix syntax error and handle negative inputs
2  def factorial(n):
3      if n == 0:
4          return 1
5      elif n < 0:
6          return "Factorial is not defined for negative numbers"
7      else:
8          return n * factorial(n - 1)
9
10 print(factorial(5)) # Expected output: 120
11 print(factorial(0)) # Expected output: 1
12 print(factorial(-3)) # Expected output: "Factorial is not defined for negative numbers"
13

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER COMMENTS

```

PS C:\Users\slp13801\OneDrive - Simplilearn Solutions Pvt Ltd\Desktop\PythonGame> & C:/Users/slp13801/AppData/Local/Programs/Python/Python31
3/python.exe c:/Users/slp13801/Downloads/Untitled-1.py
120
1
Factorial is not defined for negative numbers
PS C:\Users\slp13801\OneDrive - Simplilearn Solutions Pvt Ltd\Desktop\PythonGame>

```

Note: The output in the terminal window should match the expected result.

By following these steps, you have successfully identified, debugged, and optimized an AI-generated function using GitHub Copilot. You have learned to analyze syntax and logical errors, leverage Copilot for automated fixes, and validate solutions with test cases. This structured debugging approach ensures that AI-assisted coding produces reliable and optimized results for real-world applications.