

AI Assisted Problem Solving Using Python

AcademicYear:2025-2026

| Assignment Type: Lab  Assig\_No. 2 |
| --- |

Submitted by:

Name: **Elijah M. Flomo**

Enrol No. **2506B09602**

Course: Msc Computer Science

Submitted to:

Name: Venkataramana Veeramsetty

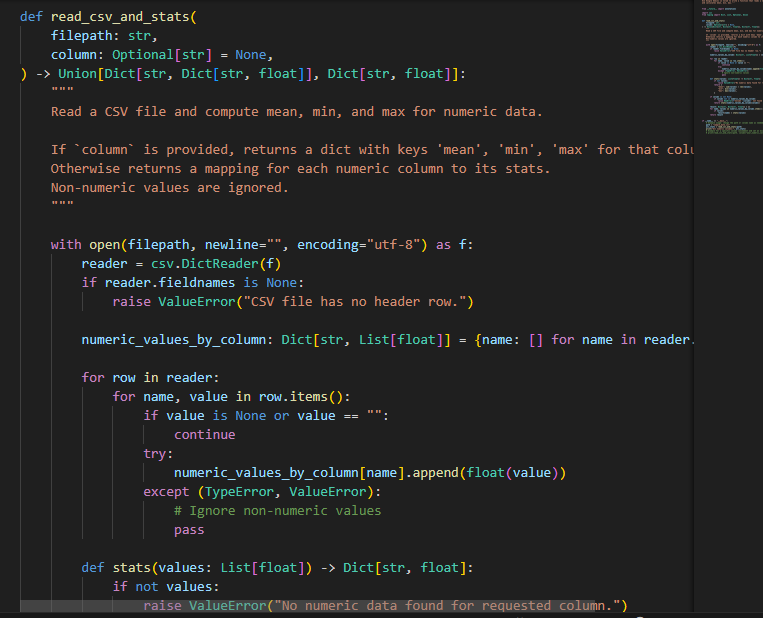
Title: Course Instructor

Date: 28 Oct 2025

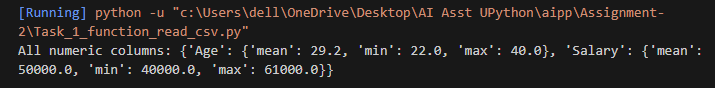
Q 1: **Expected Output#1**

Functional code with output and screenshot

Functional code:



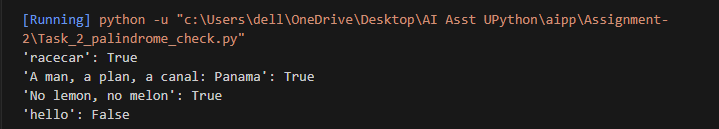
Output screenshot:



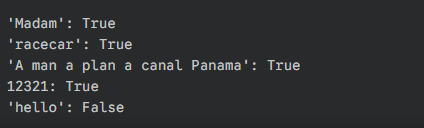
Q 2: **Expected Output#2**

Side-by-side comparison and observations

Copilot



Gemini

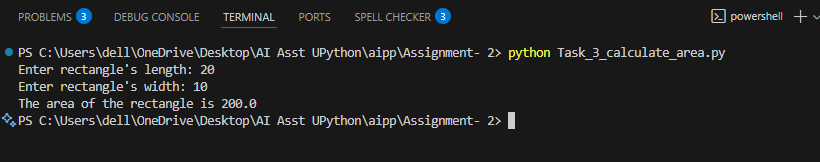


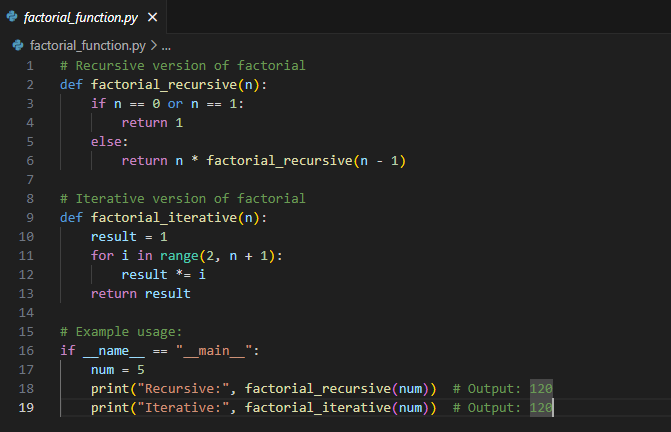
Q3: **Expected Output #3**

Detailed explanation

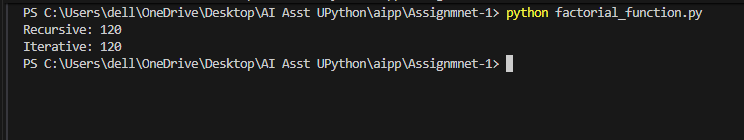
| **Line(s)** | **Code Snippet** | **Explanation** |
| --- | --- | --- |
| **1** | import math | **Imports the built-in math module**, which provides access to mathematical functions and constants, most notably math.pi (the value of pi), needed for circle calculations. |
| **3** | def calculate\_area(name): | **Defines a function** named calculate\_area that accepts one parameter: name (a string representing the shape's name). |
| **7** | name = name.lower() | **Converts the input name string to lowercase** to ensure the function is case-insensitive (e.g., "Rectangle" or "rectangle" both work). |
| **9** | if name == "rectangle": | **Starts a conditional block.** Checks if the lowercase shape name is "rectangle". |
| **10** | l = float(input(...)) | If it's a rectangle, **prompts the user** for the length and converts the input to a **floating-point number** (allowing for decimal values). |
| **11** | w = float(input(...)) | **Prompts the user** for the width, also converting the input to a float. |
| **12** | rect\_area = l \* w | **Calculates the area** of the rectangle (Area = Length $\times$ Width). |
| **13** | print(f"...") | **Prints the calculated area** using an f-string for formatted output. |
| **15** | elif name == "circle": | **Checks the next condition.** If the first if failed, it checks if the shape name is "circle". |
| **16** | r = float(input(...)) | If it's a circle, **prompts the user** for the radius and converts the input to a float. |
| **17** | circ\_area = math.pi \* r\*\*2 | **Calculates the area** of the circle (Area = $\pi \times r^2$), using math.pi and the exponent operator \*\*2. |
| **18** | print(f"...") | **Prints the calculated circle area.** |
| **20** | elif name == "triangle": | **Checks the next condition.** If the previous checks failed, it checks if the shape name is "triangle". |
| **21** | b = float(input(...)) | If it's a triangle, **prompts the user** for the base length and converts it to a float. |
| **22** | h = float(input(...)) | **Prompts the user** for the height, converting it to a float. |
| **23** | tri\_area = 0.5 \* b \* h | **Calculates the area** of the triangle (Area = $\frac{1}{2} \times \text{Base} \times \text{Height}$). |
| **24** | print(f"...") | **Prints the calculated triangle area.** |
| **26** | else: | **The final block**—runs if *none* of the preceding conditions (if or elif) were true. |
| **27** | print("...") | **Prints an error message** indicating the shape is not supported by the function. |

code snippet



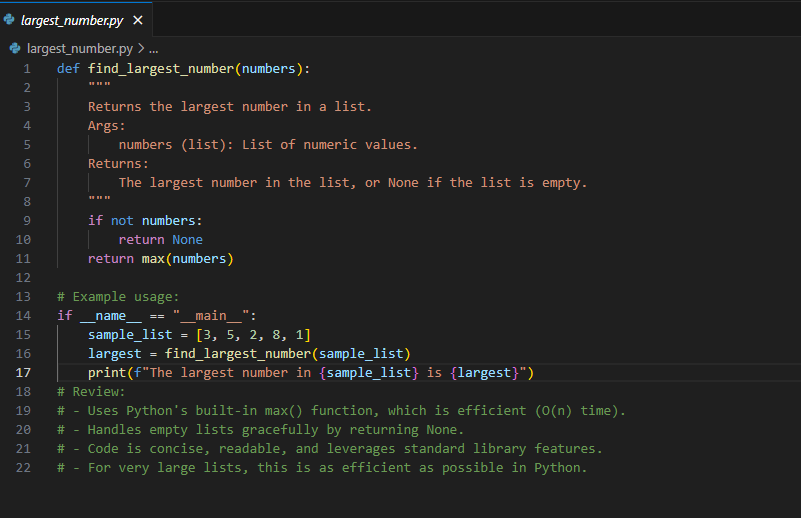


Output: Two working factorial implementations

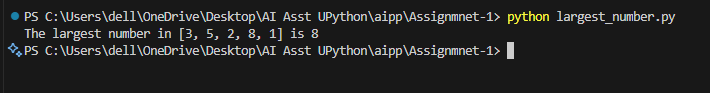


Q 5: Use Copilot to find the largest number in a list. Assess code quality and efficiency

Code:

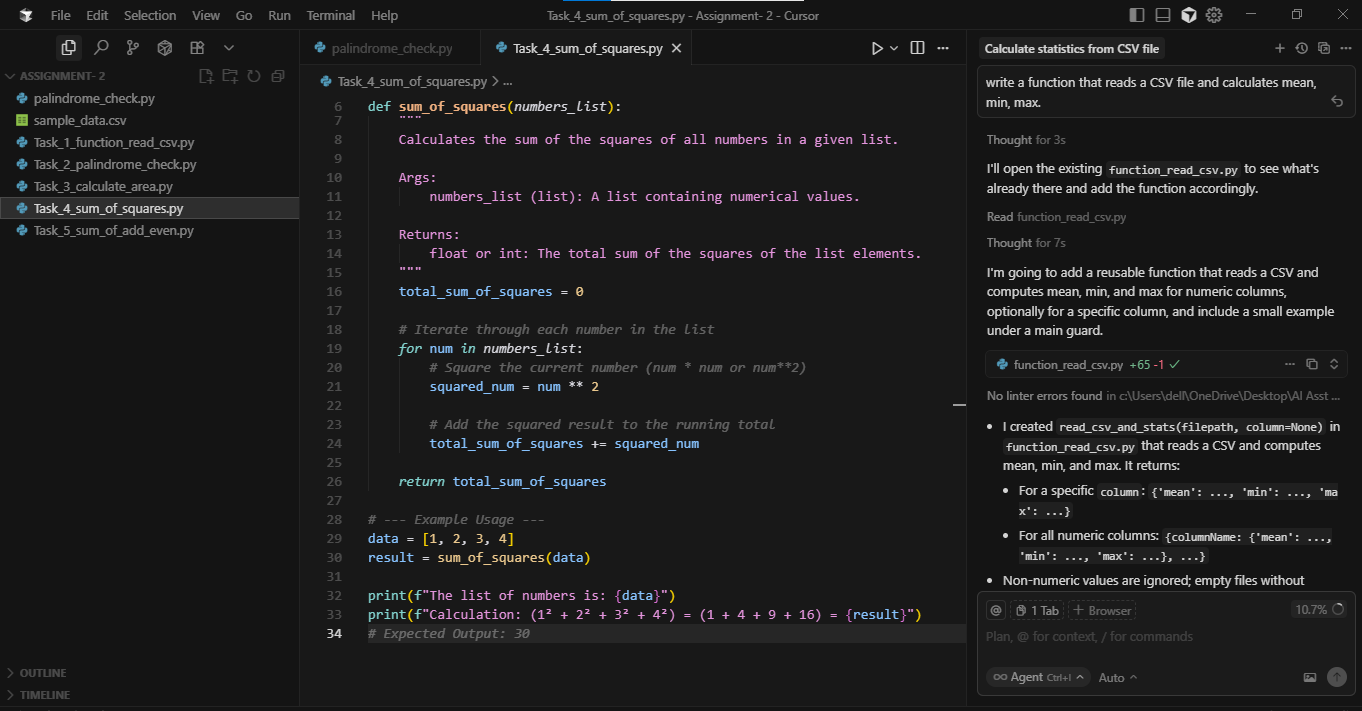


Output: A valid function with your review



Q: **Expected Output #4**

Screenshots of working environments with few prompts to generate python code



Q 5: **Expected Output#5**

Refactored code written by student with improved logic

def calculate\_odd\_even\_sum(numbers\_list):

odd\_sum = 0

even\_sum = 0

for num in numbers\_list:

if num % 2 == 0:

even\_sum += num

else:

odd\_sum += num

return odd\_sum, even\_sum

list\_num = [2, 3, 4, 5, 6, 7, 8, 9]

odd\_sum, even\_sum = calculate\_odd\_even\_sum(list\_num)

print(f"List: {list\_num}")

print(f"The sum of odd numbers is {odd\_sum}")

print(f"The sum of even numbers is {even\_sum}")