AI ASSISTANT CODING

ASSIGNMENT-9

Name: Kommu Madhupriya

Hallticket:2303A51583

Batch:22

Task Description -1 (Documentation – Function Summary Generation)

Task:

Use AI to generate concise functional summaries for each Python function in a given script.

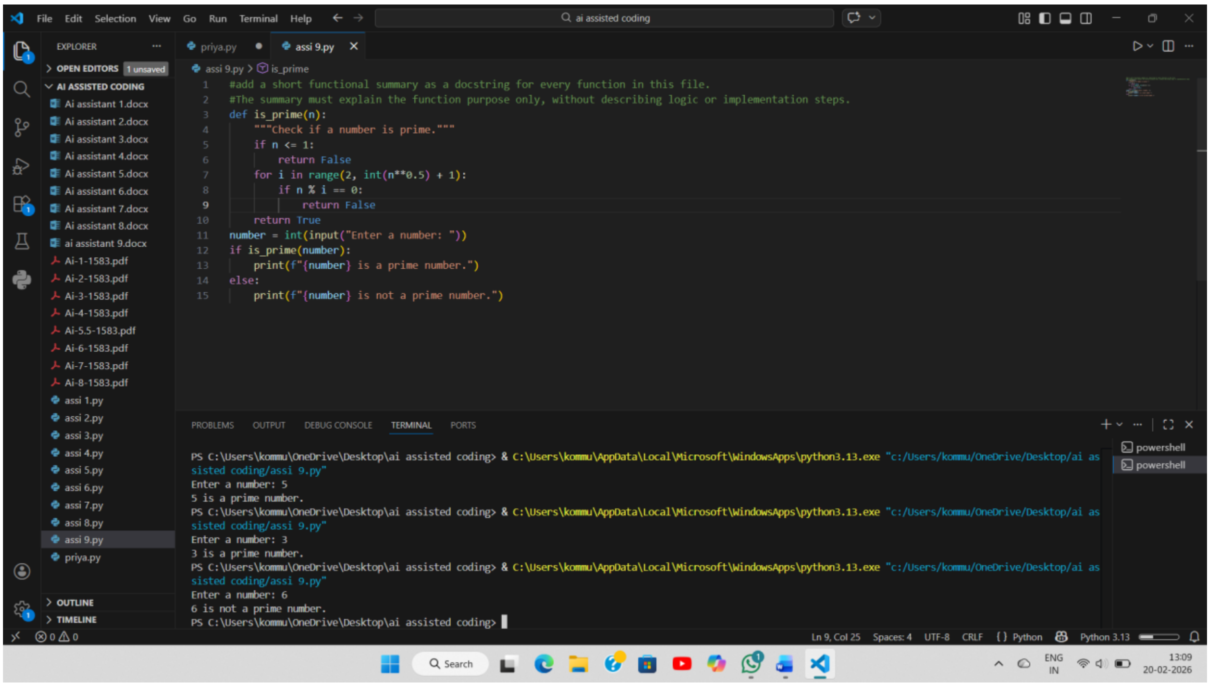
Instructions:

- Provide a Python script to the AI.
- Ask the AI to write a short summary describing the purpose of each function.
- Ensure summaries are brief and technically accurate.
- Do not include code implementation details.

Expected Output -1:

A Python script where each function contains a clear and concise summary explaining its purpose.

Code:



```python
#add a short functional summary as a docstring for every function in this file.
#The summary must explain the function purpose only, without describing logic or implementation steps.
def is_prime(n):
    """Check if a number is prime."""
    if n <= 1:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True
number = int(input("Enter a number: "))
if is_prime(number):
    print(f"{number} is a prime number.")
else:
    print(f"{number} is not a prime number.")
```

```
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai as
sisted coding/assi 9.py"
Enter a number: 5
5 is a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai as
sisted coding/assi 9.py"
Enter a number: 3
3 is a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai as
sisted coding/assi 9.py"
Enter a number: 6
6 is not a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding>
```

Observation:

AI-generated function summaries help improve code readability by clearly describing the purpose of each function. The summaries make it easier for developers to understand the program structure without analyzing the implementation details. This approach also supports faster maintenance and documentation consistency.

Task Description -2 (Documentation – Logical Explanation for

Conditions and Loops)

Task:

Use AI to document the logic behind conditional statements and loops in

a Python program.

Instructions:

- Provide a Python program without comments.

- Instruct AI to explain only decision-making logic and loop

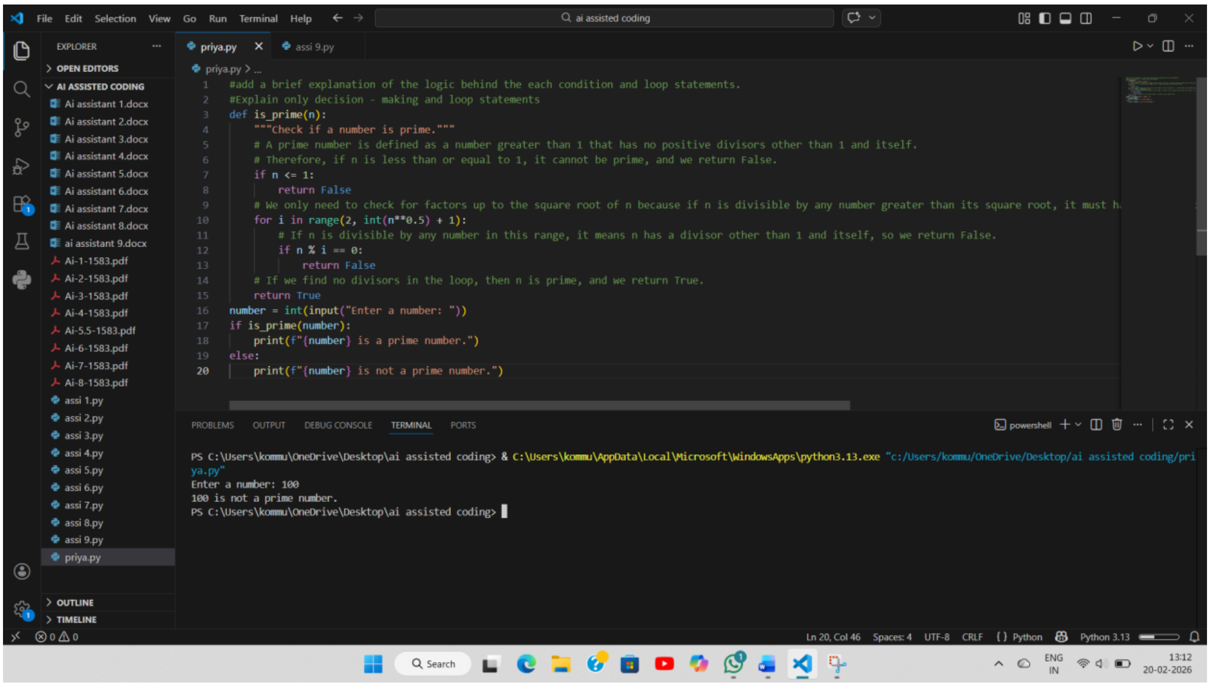
behavior.

- Skip basic syntax explanations.

Expected Output -2:

Python code with clear explanations describing the logic of conditions and loops.

code:



```python
#add a brief explanation of the logic behind the each condition and loop statements.
#Explain only decision - making and loop statements
def is_prime(n):
    """Check if a number is prime."""
    # A prime number is defined as a number greater than 1 that has no positive divisors other than 1 and itself.
    # Therefore, if n is less than or equal to 1, it cannot be prime, and we return False.
    if n <= 1:
        return False
    # We only need to check for factors up to the square root of n because if n is divisible by any number greater than its square root, it must h
    for i in range(2, int(n**0.5) + 1):
        # If n is divisible by any number in this range, it means n has a divisor other than 1 and itself, so we return False.
        if n % i == 0:
            return False
    # If we find no divisors in the loop, then n is prime, and we return True.
    return True
number = int(input("Enter a number: "))
if is_prime(number):
    print(f"{number} is a prime number.")
else:
    print(f"{number} is not a prime number.")
```

```
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: 100
100 is not a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding>
```

Observation:

AI-generated explanations for conditions and loops clarify the decision-making flow and iteration behavior within the program. These logical comments help users quickly understand how the program controls execution paths. As a result, debugging and code comprehension become more efficient.

Task Description -3 (Documentation – File-Level Overview)

Task:

Use AI to generate a high-level overview describing the functionality of

an entire Python file.

Instructions:

- Provide the complete Python file to AI.

- Ask AI to write a brief overview summarizing the file's purpose

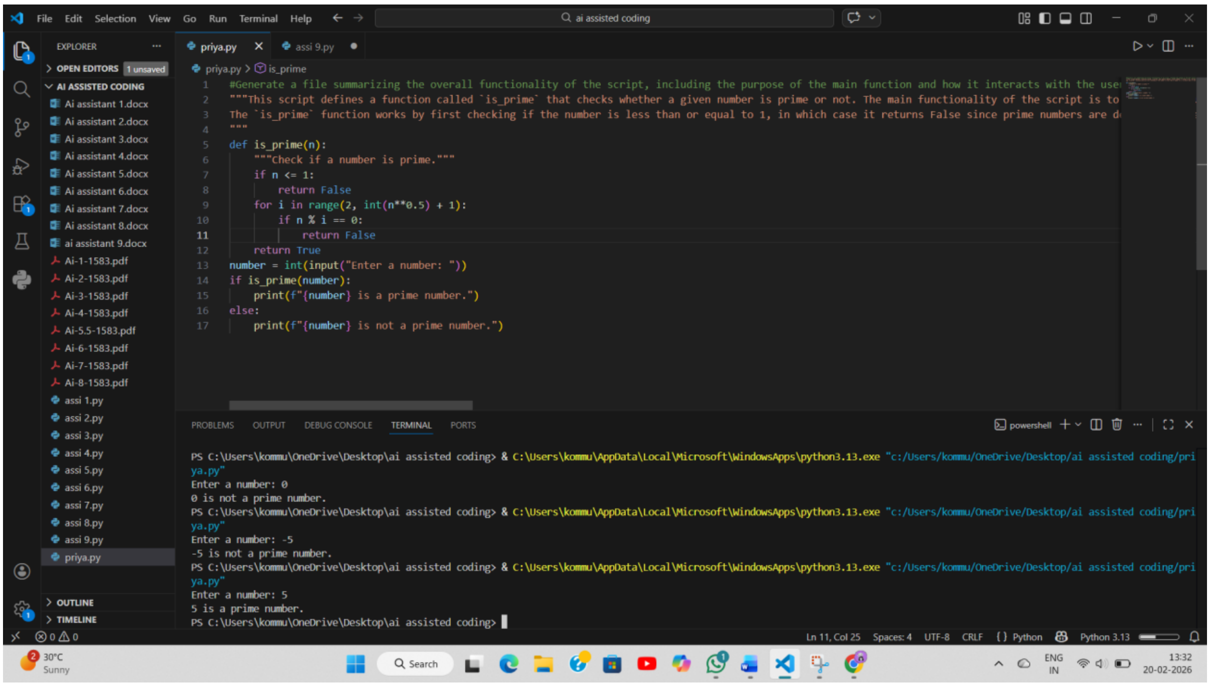
and functionality.

- Place the overview at the top of the file.

Expected Output -3:

A Python file with a clear and concise file-level overview at the beginning.

code:



```
#Generate a file summarizing the overall functionality of the script, including the purpose of the main function and how it interacts with the use
"""This script defines a function called `is_prime` that checks whether a given number is prime or not. The main functionality of the script is to
The `is_prime` function works by first checking if the number is less than or equal to 1, in which case it returns False since prime numbers are d
"""
def is_prime(n):
    """Check if a number is prime."""
    if n <= 1:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True
number = int(input("Enter a number: "))
if is_prime(number):
    print(f"{number} is a prime number.")
else:
    print(f"{number} is not a prime number.")
```

```
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: 0
0 is not a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: -5
-5 is not a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: 5
5 is a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding>
```

Script Summary:

This script defines a function called is_prime that checks whether a given number is a prime number or not.The main function prompts the user to input a number, then calls the is_prime function with the user's input.Based on the return value of the is_prime function, it prints out whether the number is a prime number or not.The script uses basic control flow statements to determine the primality of the number and provides feedback to the user accordingly.

Observation: The AI successfully generated a concise file-level overview summarizing the overall purpose and functionality of the Python program. The overview was placed at the beginning of the file and described the program at a conceptuallevel without including implementation details. This improved the readability of the code and helped users quickly understand the program’s objective.

Task Description -4 (Documentation – Refine Existing Documentation)

Task:

Use AI to improve clarity and consistency of existing documentation in Python code.

Instructions:

- Provide Python code containing basic or unclear comments.
- Ask AI to rewrite the documentation to improve clarity and

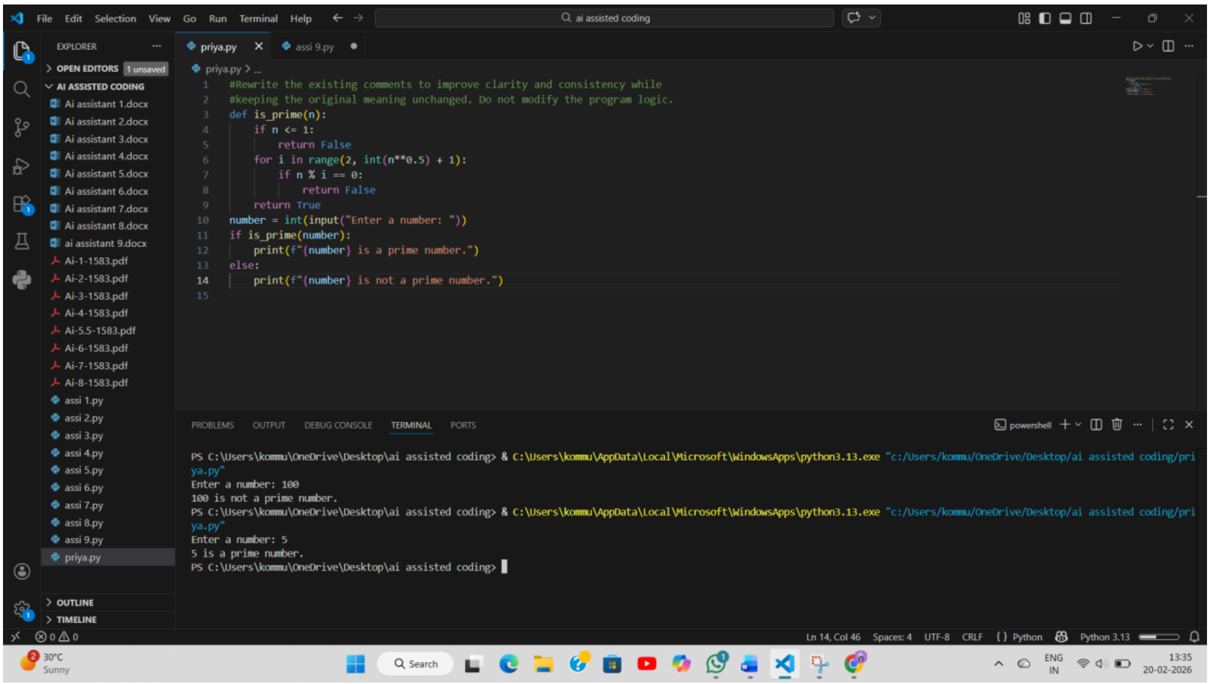
consistency.

- Ensure technical meaning remains unchanged.

Expected Output -4:

Python code with refined and improved documentation that is clear and Consistent.

Code:



```
#Rewrite the existing comments to improve clarity and consistency while
#keeping the original meaning unchanged. Do not modify the program logic.
def is_prime(n):
    if n <= 1:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True
number = int(input("Enter a number: "))
if is_prime(number):
    print(f"{number} is a prime number.")
else:
    print(f"{number} is not a prime number.")
```

```
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/priya.py"
Enter a number: 100
100 is not a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/priya.py"
Enter a number: 5
5 is a prime number.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding>
```

Observation: The AI improved the clarity and consistency of the existing comments without altering the program's functionality. The refined documentation provided clearer explanations while preserving the original technical meaning. This enhancement made the code easier to read, understand, and maintain.

Task Description -5 (Documentation – Prompt Detail Impact Study)

Task:

Study the impact of prompt detail on AI-generated documentation

quality.

Instructions:

Create two prompts: one brief and one detailed.

- Use both prompts to document the same Python function.

- Compare the generated outputs.

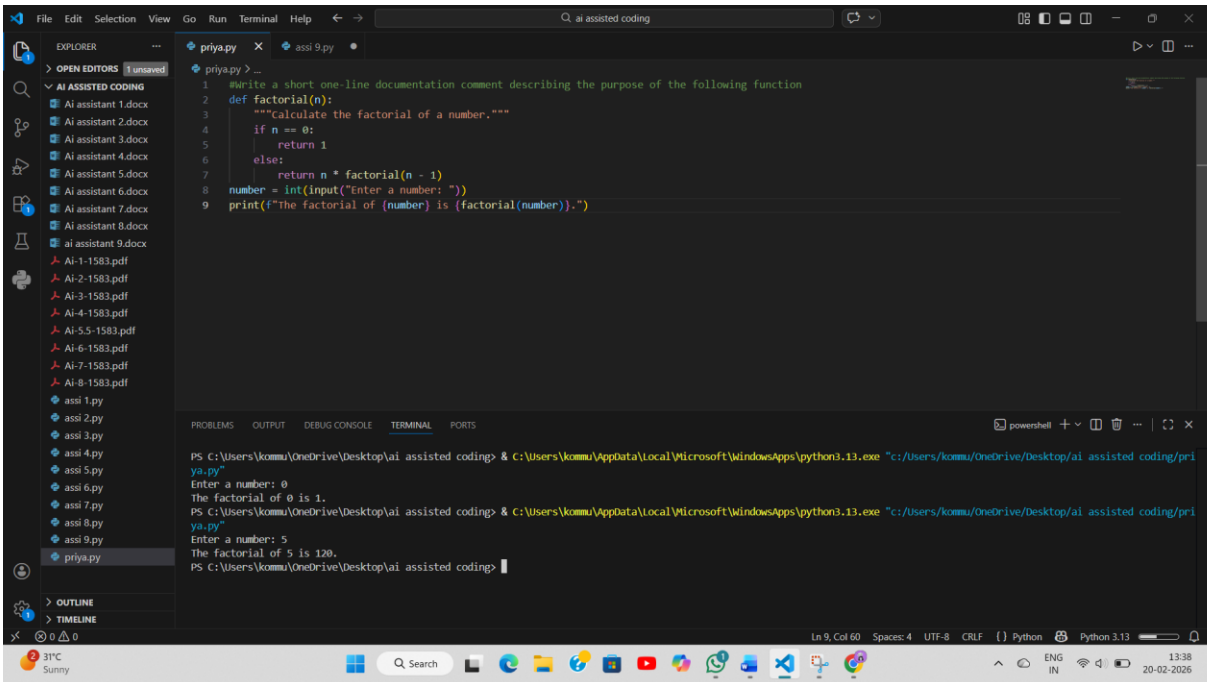
Expected Output -5:

A comparison table highlighting differences in completeness, clarity, and accuracy of documentation.

Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots.

Code:

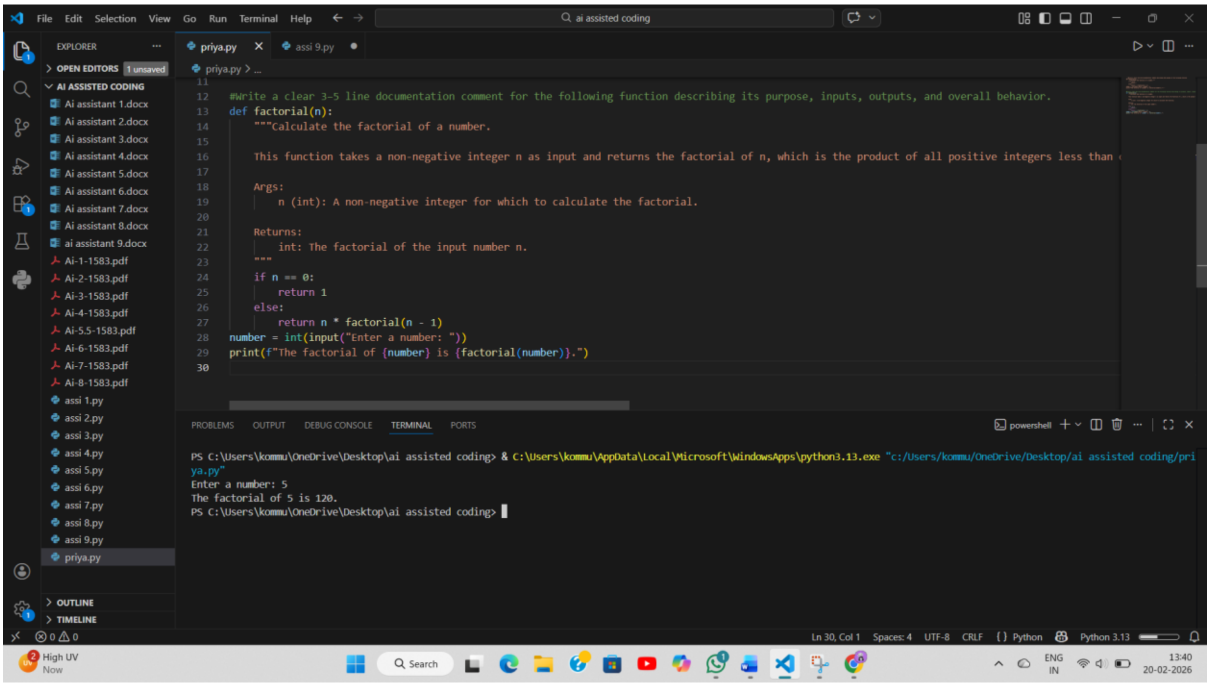
Brief code:



```python
#Write a short one-line documentation comment describing the purpose of the following function
def factorial(n):
    """Calculate the factorial of a number."""
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)
number = int(input("Enter a number: "))
print(f"The factorial of {number} is {factorial(number)}.")
```

```
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: 0
The factorial of 0 is 1.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: 5
The factorial of 5 is 120.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding>
```

Detail code:



```python
#Write a clear 3-5 line documentation comment for the following function describing its purpose, inputs, outputs, and overall behavior.
def factorial(n):
    """Calculate the factorial of a number.

    This function takes a non-negative integer n as input and returns the factorial of n, which is the product of all positive integers less than

    Args:
        n (int): A non-negative integer for which to calculate the factorial.

    Returns:
        int: The factorial of the input number n.
    """
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)
number = int(input("Enter a number: "))
print(f"The factorial of {number} is {factorial(number)}.")
```

```
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding> & C:\Users\kommu\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/Users/kommu/OneDrive/Desktop/ai assisted coding/pri
ya.py"
Enter a number: 5
The factorial of 5 is 120.
PS C:\Users\kommu\OneDrive\Desktop\ai assisted coding>
```

Observation:

Criteria	Brief Prompt Output (One-line Comment)	Detailed Prompt Output (3–5 line Docstring)
Completeness	Provides only the basic purpose of the function.	Includes purpose, input description, output, and behavior details.
Clarity	Clear but very minimal explanation.	More descriptive, structured, and easier to understand.
Accuracy	Correct but limited technical information.	Highly accurate with full technical context and parameter details.