

ASSIGNMENT-4.2

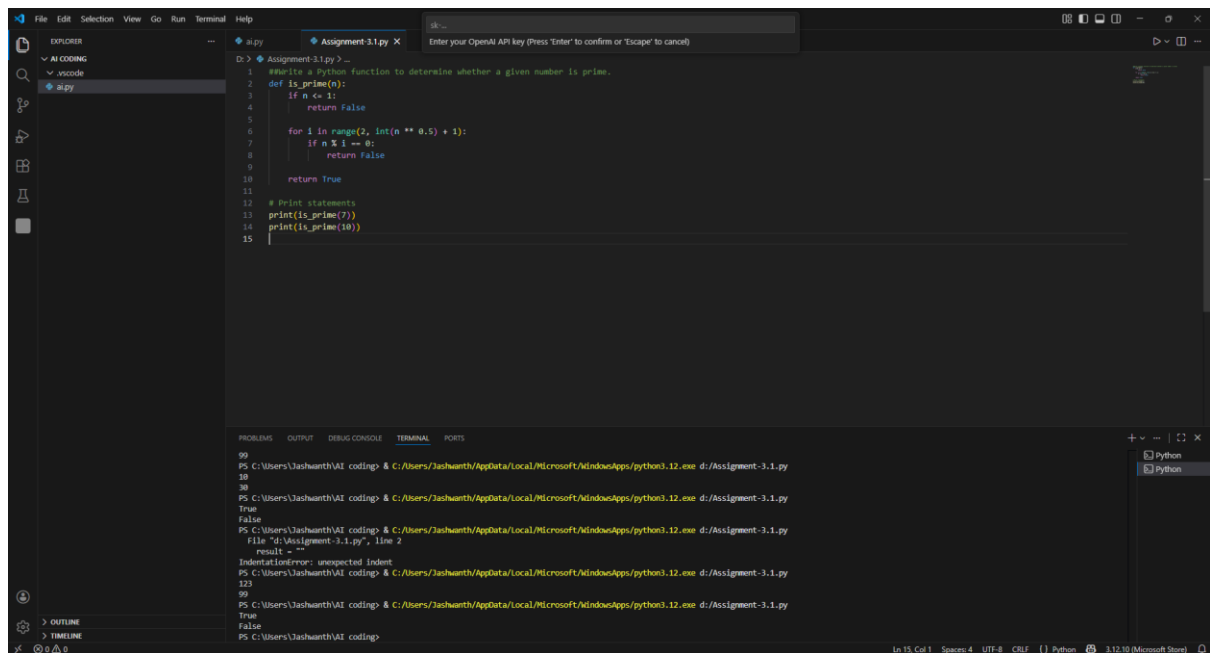
NAME-SRAVANI

ROLLNO:2303A510G7

TASK-1: ZERO-SHOT PROMPTING

PROMPT: Write a Python function to determine whether a given number is prime.

CODE:



```
D:\> cd Assignment-3.1.py >
1 #Write a Python function to determine whether a given number is prime.
2 def is_prime(n):
3     if n <= 1:
4         return False
5
6     for i in range(2, int(n ** 0.5) + 1):
7         if n % i == 0:
8             return False
9
10    return True
11
12 # Print statements
13 print(is_prime(7))
14 print(is_prime(10))
15
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
99 PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:\Assignment-3.1.py
10 10
30 PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:\Assignment-3.1.py
True
False
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:\Assignment-3.1.py
File "d:\Assignment-3.1.py", line 2
    result =
IndentationError: unexpected indent
123 PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:\Assignment-3.1.py
99 True
False
PS C:\Users\Jashwanth\AI coding>
```

OBSERVATION:

- AI model understands the concept of a prime number without being given any examples or additional guidance -It applies correct mathematical reasoning purely from the instruction -The model generates syntactically correct and logically sound Python code

TASK-2

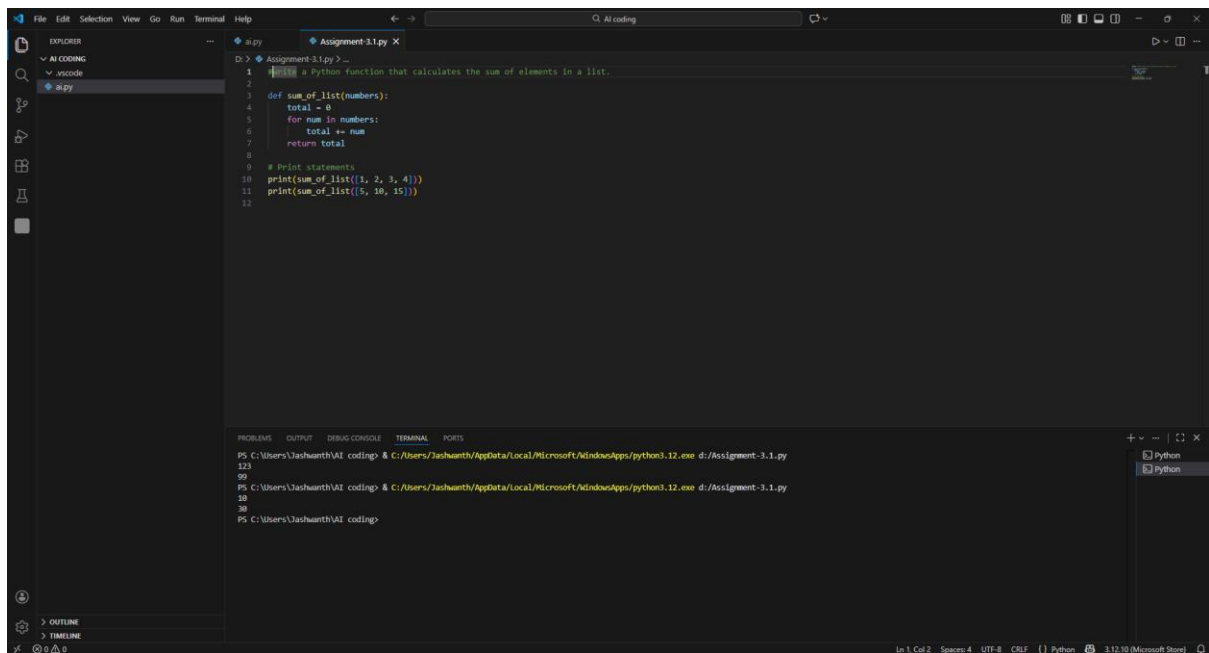
PROMPT: Write a Python function that calculates the sum of elements in a list.

Example:

Input: [1, 2, 3, 4]

Output: 10

CODE:



The screenshot shows a VS Code editor window with a file named 'Assignment-3.1.py'. The code defines a function 'sum_of_list' that calculates the sum of elements in a list. It includes comments and test cases. The terminal at the bottom shows the command 'python d:/Assignment-3.1.py' being executed, resulting in the output '123', '98', '18', and '38' on separate lines.

```
1 # Write a Python function that calculates the sum of elements in a list.
2
3 def sum_of_list(numbers):
4     total = 0
5     for num in numbers:
6         total += num
7     return total
8
9 # Print statements
10 print(sum_of_list([1, 2, 3, 4]))
11 print(sum_of_list([5, 10, 15]))
12
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Jashwanth\AI coding> python d:/Assignment-3.1.py

123

PS C:\Users\Jashwanth\AI coding> python d:/Assignment-3.1.py

98

PS C:\Users\Jashwanth\AI coding> python d:/Assignment-3.1.py

18

PS C:\Users\Jashwanth\AI coding> python d:/Assignment-3.1.py

38

OBSERVATION:

The single example guides the AI model to understand the expected input and output relationship The model correctly generalizes the pattern from the example to any list of numbers

TASK-3

PROMPT: Write a Python function that extracts only digits from an alphanumeric string.

Examples:

Input: "a1b2c3"

Output: "123"

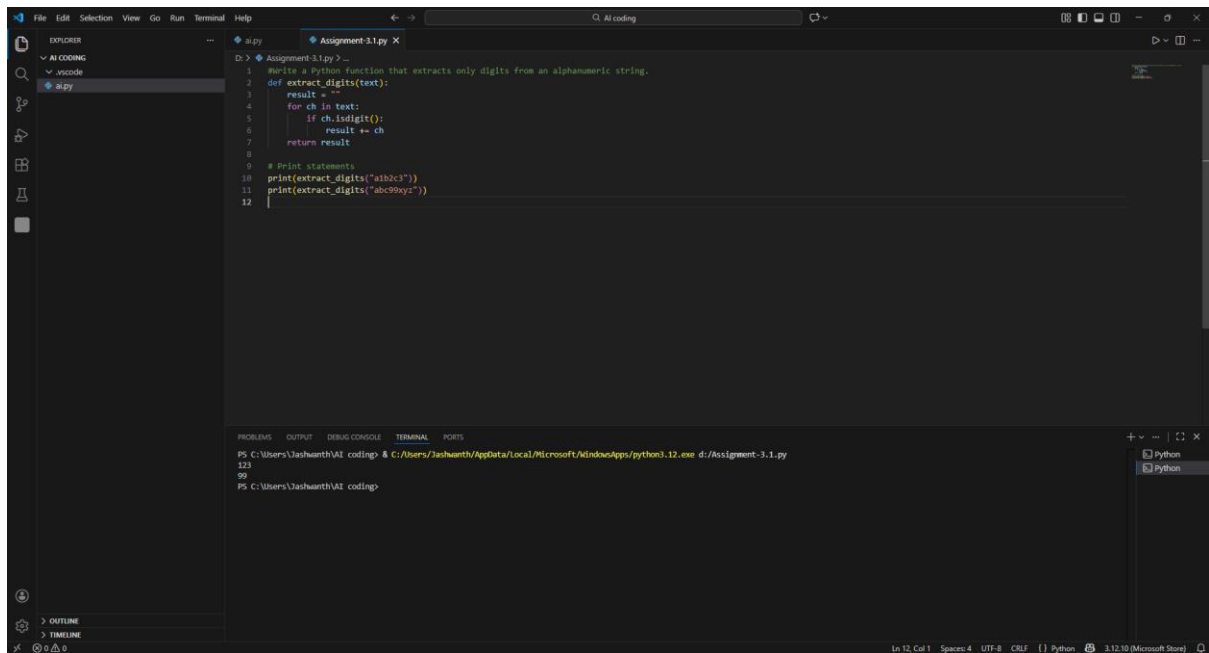
Input: "x9y8z7"

Output: "987"

Input: "abc123def"

Output: "123"

CODE:



OBSERVATION:

- Multiple examples help the AI model clearly identify the pattern to be learned -The model focuses only on digit characters and ignores alphabetic content
- The AI demonstrates improved confidence and reduced ambiguity compared to zero shot and one shot prompting

TASK-4

PROMPT: ZERO-SHOT: Write a Python function that counts the number of vowels in a string.

FEW-SHOT: Write a Python function that counts the number of vowels in a string.

Examples:

Input: "hello"

Output: 2

Input: "AEIOU"

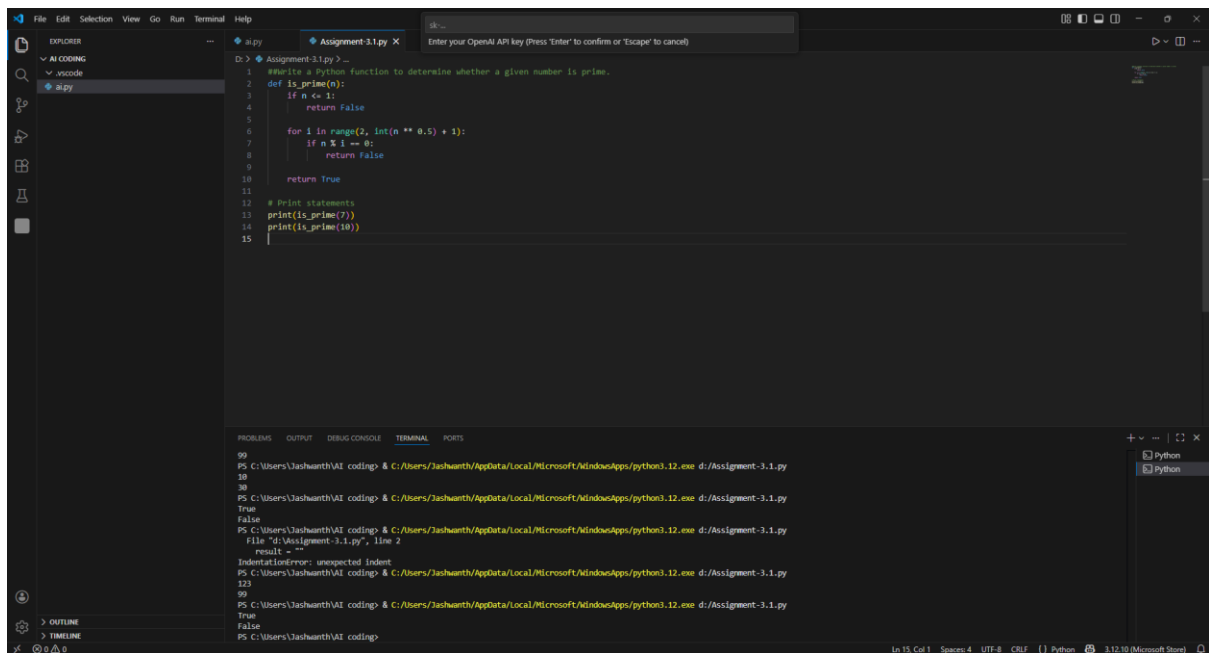
Output: 5

Input: "chatgpt"

Output: 2

CODE:

ZERO-SHOT:

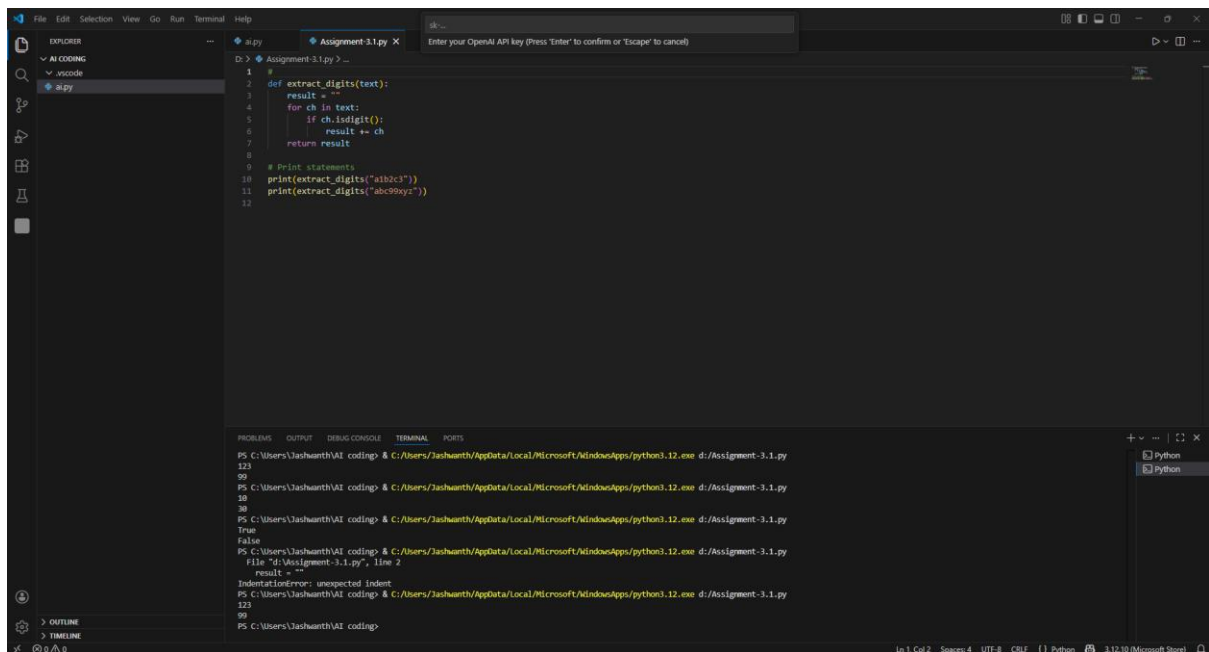


```
D:\> Assignment-3.1.py > ...
1 # Write a Python function to determine whether a given number is prime.
2 def is_prime(n):
3     if n <= 1:
4         return False
5
6     for i in range(2, int(n ** 0.5) + 1):
7         if n % i == 0:
8             return False
9
10    return True
11
12 # Print statements
13 print(is_prime(7))
14 print(is_prime(10))
15
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
10
30
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
True
False
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
File "d:\Assignment-3.1.py", line 2
    result = ...
    ^
IndentationError: unexpected indent
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
99
True
False
PS C:\Users\Jashwanth\AI coding>
```

FEW-SHOT:



```
D:\> Assignment-3.1.py > ...
1 #
2 def extract_digits(text):
3     result = ""
4     for ch in text:
5         if ch.isdigit():
6             result += ch
7     return result
8
9 # Print statements
10 print(extract_digits("a1b2c3"))
11 print(extract_digits("abc9xyz"))
12
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
123
99
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
30
30
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
True
False
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
File "d:\Assignment-3.1.py", line 2
    result = ...
    ^
IndentationError: unexpected indent
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
123
99
PS C:\Users\Jashwanth\AI coding>
```

def count_vowels(text):

vowels = "aeiouAEIOU"

count = 0

for ch in text:

if ch in vowels:

count += 1

return count

OBSERVATION:

FEW-SHOT OBSERVATION

The provided examples clearly define what characters should be counted as vowels. The model confidently includes both uppercase and lowercase vowels due to examples.

ZERO SHOT:

zero shot prompting the AI guesses the intent based on general knowledge which may vary for ambiguous tasks.

TASK-5

PROMPT:

Write a Python function that determines the minimum of three numbers without using the built-in min() function.

Examples:

Input: 3, 7, 5

Output: 3

Input: 10, 2, 8

Output: 2

Input: 4, 4, 9

Output: 4

CODE:

The screenshot shows a code editor with a Python file named `Assignment 3.1.py`. The code defines a function `find_min(a, b, c)` that returns the minimum of three numbers using conditional logic. Below the code, the terminal output shows the function being called with various arguments, and the results are printed. The output includes an indentation error on line 123.

```
D:\> D:\Assignment 3.1.py
1 def find_min(a, b, c):
2     if a <= b and a <= c:
3         return a
4     elif b <= a and b <= c:
5         return b
6     else:
7         return c
8
9 # Print statements
10 print(find_min(3, 7, 5))
11 print(find_min(10, 2, 8))
12 print(find_min(4, 4, 9))
13
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
File "d:/Assignment-3.1.py", line 2
    result =
    ^
IndentationError: unexpected indent
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
30
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
True
False
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
True
False
PS C:\Users\Jashwanth\AI coding> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Assignment-3.1.py
3
4
PS C:\Users\Jashwanth\AI coding>
```

OBSERVATION:

The examples clearly establish the comparison pattern needed to identify the smallest value. The AI model infers the requirement to handle equality cases correctly. Conditional logic is generated without relying on built-in functions.