

ASSIGNMENT – 4.2

NAME : M.SAI CHARANI

BATCH : 34

HALL TICKET NUMBER : 2303A52192

Task Description-1

- Zero-shot: Prompt AI with only the instruction. Write a Python function to determine whether a given number is prime

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows files in the "AI CODING" folder: assignment_12.py, assignment_32.py, assignment_42.py, assignment_45.py, iterative_factorial.py, and nb_email_classifier.pkl.
- Code Editor:** The active file is assignment_42.py, containing the following Python code:

```
def is_prime(n: int) -> bool:
    """Return True if n is prime, else False."""
    if not isinstance(n, int):
        return False
    if n <= 1:
        return False
    if n <= 3:
        return True
    if n % 2 == 0 or n % 3 == 0:
        return False
    i = 5
    while i * i <= n:
        if n % i == 0 or n % (i + 2) == 0:
            return False
        i += 6
    return True

# ----- Test / Output Section -----
if __name__ == "__main__":
    print("Prime Number Check Results:")
    print("2 ->", is_prime(2)) # True
    print("3 -> True")
    print("4 -> False")
    print("7 -> True")
    print("10 -> False")
    print("1 -> False")
```

- Terminal:** Shows command-line output from running the script:

```
PS C:\Users\LENOVO\OneDrive\Desktop>New Folder\ai_coding> & C:\Users\LENOVO\AppData\Local\Python\pythoncore-3.14-64\python.exe "C:/Users/LENOVO/OneDrive/Desktop/New Folder/ai_coding/assignment_4_2.py"
PS C:\Users\LENOVO\OneDrive\Desktop>New Folder\ai_coding> & C:\Users\LENOVO\AppData\Local\Python\pythoncore-3.14-64\python.exe "C:/Users/LENOVO/OneDrive/Desktop/New Folder/ai_coding/assignment_4_2.py"
Prime Number Check Results:
2 -> True
3 -> True
4 -> False
7 -> True
10 -> False
1 -> False
```

- Output Panel:** Displays the results of the prime number check.
- Bottom Status Bar:** Shows system information like weather (19°C, Sunny), battery level, and system time (10:10 AM, 1/20/2026).

The screenshot shows the Microsoft Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** AI assistance.
- Explorer:** Shows files ass-4.2.py, ass-4.2.py, ass-1.py, ass-3.2.py, ass-4.2.py, and ass-4.5.py.
- Editor:** Displays Python code for a prime number check function. The code includes imports, a prime check function, and a test section. Lines 129 through 160 show the prime check logic.
- Terminal:** Shows command-line output for running the script.
- Chat:** A sidebar titled "PYTHON FUNCTION TO SUM LIST ELEMENTS" contains AI-generated code for summing elements in a list. It includes a code snippet, usage notes, and a "Write a Python function that" prompt.
- Bottom Status Bar:** Includes file path (C:\Users\meeri\OneDrive\Desktop\AI assistance), line count (Ln 136), column count (Col 43), spaces (Spaces: 4), encoding (UTF-8), line endings (CRLF), Python icon, version (3.11.5), date (20-01-2026), and system status (10:11, ENG IN).

Task Description-2

- One-shot: Provide one example: Input: [1, 2, 3, 4], Output: 10 to help AI generate a function that calculates the sum of elements in a list.

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows files ass-4.2.py, ass-1.py, ass-3.2.py, ass-4.2.py, and ass-4.5.py.
- Code Editor:** Displays the following Python code:


```

41 def is_prime(n):
42     """Return True if n is a prime number, else False."""
43     if n < 1:
44         return False
45     for i in range(2, int(n ** 0.5) + 1):
46         if n % i == 0:
47             return False
48     return True
49
50
51 def sum_of_list(lst):
52     """Return the sum of elements in the list."""
53     return sum(lst)
54
55
56 def extract_digits(text):
57     """Return only the numeric characters from 'text' as a string."""
58     return ''.join(ch for ch in str(text) if ch.isdigit())
59
60
61 if __name__ == "__main__":
62     # Testing is_prime function
63

```
- Terminal:** Shows the command to run the file and the output of the Prime Check Results and Extract Digits Results.
- Chat Panel:** A sidebar titled "PYTHON FUNCTION TO SUM LIST ELEMENTS" contains the following text:

Write a Python function that extracts and returns only the digits from an alphanumeric string.

Examples:

 - Input: "abc123" → Output: "123"
 - Input: "a1b2c3" → Output: "123"
 - Input: "2024year" → Output: "2024"

```

def extract_digits(text):
    pass

```
- Bottom Status Bar:** Shows the file path (C:\Users\meeri\OneDrive\Desktop\AI assistance\ass-4.2.py), line count (Ln 79, Col 1), and date (20-01-2026).

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows files ass-4.2.py, ass-1.py, ass-3.2.py, ass-4.2.py, and ass-4.5.py.
- Code Editor:** Displays the following Python code:


```

20
21 def sum_of_list(numbers):
22     """Return the sum of all elements in the list `numbers`."""
23     total = 0
24     for value in numbers:
25         total += value
26     return total
27
28
29 # ----- Test / Output Section -----
30 if __name__ == "__main__":
31     # Testing is_prime function
32     print("Prime Check Results:")
33     print("7 ->, is_prime(7)) # True
34     print("10 ->, is_prime(10)) # False
35     print("1 ->, is_prime(1)) # False
36
37     # Testing sum_of_list function
38     print("\nSum of List Results:")
39     print("[1, 2, 3, 4] ->, sum_of_list([1, 2, 3, 4])) # 10
40     print("[5, 10, 15] ->, sum_of_list([5, 10, 15])) # 30

```
- Terminal:** Shows the command to run the file and the output of the Prime Check Results and Sum of List Results.
- Chat Panel:** A sidebar titled "PYTHON FUNCTION TO SUM LIST ELEMENTS" contains the following text:

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

Example:

Input: [1, 2, 3, 4]

Output: 10

```

def sum_of_list(numbers):
    pass

```
- Bottom Status Bar:** Shows the file path (C:\Users\meeri\OneDrive\Desktop\AI assistance\ass-4.2.py), line count (Ln 79, Col 61), and date (20-01-2026).

Task Description-3

- Few-shot: Give 2–3 examples to create a function that extracts digits from an alphanumeric string.

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows files ass-4.2.py, ass-1.py, ass-3.2.py, ass-4.2.py, and ass-4.5.py.
- Editor:** Displays Python code for functions is_prime, sum_of_list, and extract_digits.
- Terminal:** Shows command-line output for prime check results and extract digits results.
- AI Assistance:** A sidebar titled "PYTHON FUNCTION TO SUM LIST ELEMENTS" contains the following text:

Write a Python function that extracts and returns only the digits from an alphanumeric string.

Examples:

 - Input: "abc123" → Output: "123"
 - Input: "a1b2c3" → Output: "123"
 - Input: "2024year" → Output: "2024"

```
def extract_digits(text):
    pass
```

Used 1 reference
Creating function code
Updated file with an extract_digits function and tests.
- Bottom Status Bar:** Shows file path, line number (Ln 79, Col 1), spaces (Spaces: 4), encoding (UTF-8), Python, and timestamp (20-01-2026 09:57).

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows files ass-4.2.py, ass-1.py, ass-3.2.py, ass-4.2.py, and ass-4.5.py.
- Editor:** Displays Python code for functions is_prime, sum_of_list, and extract_digits.
- Terminal:** Shows command-line output for prime check results and extract digits results.
- AI Assistance:** A sidebar titled "PYTHON FUNCTION TO SUM LIST ELEMENTS" contains the following text:

Write a Python function that extracts and returns only the digits from an alphanumeric string.

Examples:

 - Input: "abc123" → Output: "123"
 - Input: "a1b2c3" → Output: "123"
 - Input: "2024year" → Output: "2024"

```
def extract_digits(text):
    pass
```

Used 1 reference
Creating function code
Updated file with an extract_digits function and tests.
- Bottom Status Bar:** Shows file path, line number (Ln 79, Col 1), spaces (Spaces: 4), encoding (UTF-8), Python, and timestamp (20-01-2026 09:58).

Task Description-4

- Compare zero-shot vs few-shot prompting for generating a function that counts the number of vowels in a string.

The screenshot shows the VS Code interface with the following details:

- EXPLORER**: Shows files ass-4.2.py, ass-4.2.py > ..., ass-1.py, ass-3.2.py, ass-4.2.py, and ass-4.5.py.
- CHAT**: A sidebar titled "PYTHON FUNCTION TO SUM LIST ELEMENTS" contains:
 - Used 1 reference
 - Preparing code for summation
 - # ...existing code...
 - # Write a Python function that calculates the #
 - # Example:
 - # Input: [1, 2, 3, 4]
 - # Output: 10
 - def sum_of_list(numbers):
 - """Return the sum of all numeric elements
 - total = 0
 - for n in numbers:
 - total += n
 - return total
 - # ...existing code...
- TERMINAL**: Shows command-line history and output for the script ass-4.2.py, which prints vowel counts for various strings.
- STATUS BAR**: Shows file path C:\Users\meeri\OneDrive\Desktop\AI assistance, line Ln 92, column Col 57, spaces Spaces: 4, encoding UTF-8, CRLF, Python, 3.11.5, and timestamp 20-01-2026 10:09.

Task Description-5

- Use few-shot prompting with 3 sample inputs to generate a function that determines the minimum of three numbers without using the built-in min() function.

The screenshot shows the Visual Studio Code (VS Code) interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** Q AI assistance.
- Left Sidebar (EXPLORER):** OPEN EDITORS (ass-4.2.py), AI ASSISTANCE (AI.Assisted_Coding.R.., ass-1.py, ass-3.2.py, ass-4.2.py, ass-4.5.py).
- Editor Area:** Code editor showing Python script `ass-4.2.py`. The code defines a function `find_min(a, b, c)` to find the minimum of three numbers without using `min()`. It also prints results for various test cases.
- Terminal:** Shows command-line output of the script execution.
- Right Panel (CHAT):**
 - PYTHON FUNCTION TO SUM LIST ELEMENTS:** A snippet of code for a sum function.
 - Used 1 reference:**
 - Preparing code for summation:** A box containing existing code and instructions to calculate the sum of numeric elements.
 - Write a Python function that:** A box for exploring and understanding the code.
- Bottom Status Bar:** Ln 106, Col 65, Spaces: 4, UTF-8, CRLF, Python, 3.11.5, 10:09, ENG IN, 20-01-2026.