Assignment-4.2

Perumalla Sushwanth

batch 29

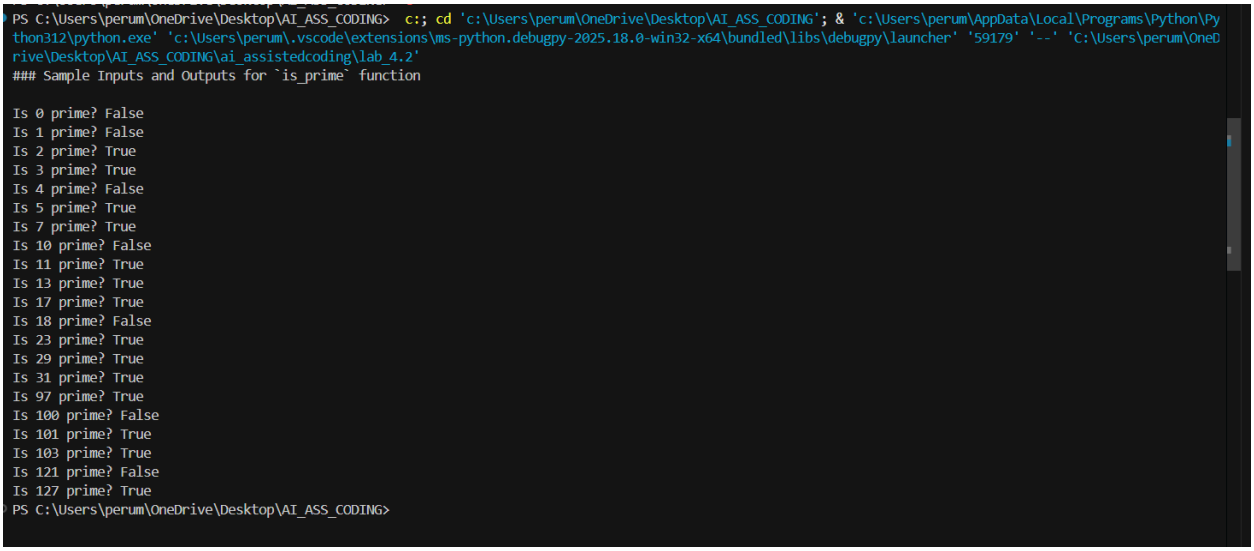
2303a51567

Task Description-1
• Zero-shot: Prompt AI with only the instruction. Write a Python function to determine
whether a given number is prime
Expected Output-1
• A basic Python function to check if a number is prime, demonstrating correct logical
conditions without relying on examples or additional context

CODE:

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

def is_prime(n):
    if n < 2:
        return False
    for i in range(2, int(n ** 0.5) + 1):
        if n % i == 0:
            return False
    return True

print("### Sample Inputs and Outputs for `is_prime` function\n")

test_numbers = [0, 1, 2, 3, 4, 5, 7, 10, 11, 13, 17, 18, 23, 29, 31, 97, 100, 101, 103, 121, 127]

for number in test_numbers:
    result = is_prime(number)
    print(f"Is {number} prime? {result}")
```

OUTPUT:



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.
Expected Output-2
• A correct conversion function guided by the single example

CODE:

```
    #Generate a function to calculate the sum of elements in a list.
def sum_of_list(lst):

    return sum(lst)

# Example usage
print(sum_of_list([1, 2, 3, 4]))
```

OUTPUT:10

```
PS C:\Users\perum\OneDrive\Desktop\AI_ASS_CODING>  c:; cd 'c:\Users\perum\OneDrive\Desktop\AI_ASS_CODING'; & 'c:\Users\perum\AppData\Local\Programs\Python\P
thon312\python.exe' 'c:\Users\perum\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55952' '--' 'C:\Users\perum\One
rive\Desktop\AI_ASS_CODING\ai_assistedcoding\lab_4.2'
10
```

Task Description-3

• Few-shot: Give 2–3 examples to create a function that extracts digits from an alphanumeric string.
Expected Output-3
• Accur
ate function that returns only the digits from alphanumeric string.

CODE:

```python
  #Extract digits from an alphanumeric string.
def extract_digits(s):

    return ''.join(filter(str.isdigit, s))

# Example usage
print(extract_digits("a1b2c3"))
```

OUTPUT:

```
PS C:\Users\perum\OneDrive\Desktop\AI_ASS_CODING>  c:; cd 'c:\Users\perum\OneDrive\Desktop\AI_ASS_CODING'; & 'c:\Users\perum\AppData\Local\Programs\Python\Py
thon312\python.exe' 'c:\Users\perum\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '56427' '--' 'C:\Users\perum\OneD
rive\Desktop\AI_ASS_CODING\ai_assistedcoding\lab_4.2'
123
```

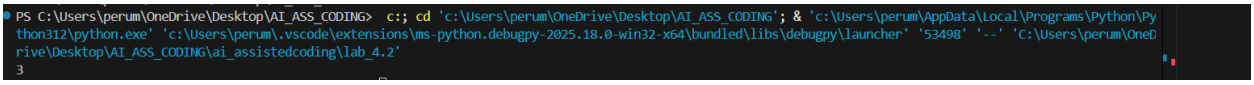
Task Description-4

• Compare zero-shot vs few-shot prompting for generating a function that counts the
number of vowels in a string.
Expected Output-4
• Output comparison + student explanation on how examples helped the model.

CODE:

```
35
36  #Count the number of vowels in a string.
37  def count_vowels(s):
38
39      vowels = "aeiouAEIOU"
40      return sum(1 for char in s if char in vowels)
41
42  # Example usage
43  print(count_vowels("Hello World"))
44
```

OUTPUT:



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.
Expected Output-5
• A function that handles all cases with correct logic based on example patterns.

CODE:

```python
#Determine the minimum of three numbers.
def min_of_three(a, b, c):

    if a <= b and a <= c:
        return a
    elif b <= a and b <= c:
        return b
    else:
        return c

# Example usage
print(min_of_three(3, 1, 2))
```

OUTPUT:

```
PS C:\Users\perum\OneDrive\Desktop\AI_ASS_CODING>  c:; cd 'c:\Users\perum\OneDrive\Desktop\AI_ASS_CODING'; & 'c:\Users\perum\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\perum\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '49931' '--' 'C:\Users\perum\OneDrive\Desktop\AI_ASS_CODING\ai_assistedcoding\lab_4.2'
1
```