Vyshnavi Parisha

2403A51L34

Batch : 52

Lab 4: Advanced Prompt Engineering – Zero-shot, One-shot, and Few-shot Techniques

Task 1: Zero-shot Prompting

Prompt:

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

Code :

```python
# AI-4.2.py
# AI-4.2.py > is_prime
1  #Write a Python function to determine whether a given number is prime.
2  def is_prime(n):
3      """Check if a number is prime.
4      Args:
5          n (int): The number to check.
6      Returns:
7          bool: True if n is prime, False otherwise.
8      """
9      if n <= 1:
10         return False
11     for i in range(2, int(n**0.5) + 1):
12         if n % i == 0:
13             return False
14     return True
15  # Example usage:
16  print(is_prime(17))  # Output: True
17  print(is_prime(15))  # Output: False
```

Explanation

- Uses only instruction, no examples. □   Checks divisibility up to √n.
- Correctly handles non-prime cases.

Output :

Task 2: One-shot Prompting

Prompt:
Generate a function to calculate the sum of elements in a list.

Code :


```python
#Generate a function to calculate the sum of elements in a list.
def sum_of_elements(input_list):
    return sum(input_list)
# Example usage:
example_list = [1, 2, 3, 4, 5]
result = sum_of_elements(example_list)
print("The sum of elements in the list is:", result)
# Output: The sum of elements in the list is: 15
```

Explanation

- Single example guides the logic.
- Iterates through list and adds elements.

Output :

Task 3: Few-shot Prompting Prompt

:

Write a Python function that takes an alphanumeric string and returns only the digits.
Code :

```
AI-4.2.py  X
AI-4.2.py > ...
1    #write a Python function that takes an alphanumeric string and returns only the digits.
2    def extract_digits(alphanumeric_string):
3        digits = ''.join(filter(str.isdigit, alphanumeric_string))
4        return digits
5    # Example usage:
6    input_string = "a1b2c3d4e5"
7    result = extract_digits(input_string)
8    print(result)  # Output: 12345
9
10
```

Explanation

- Multiple examples clarify pattern.
- Extracts only numeric characters.

Output :

```
ms\Python\Python313\python.exe' 'c:\Users\deept\.vscode\extensions\ms-python.debugpy-2025.19.2026011901-win32-x64\bun
s\debugpy\launcher' '51269' '--' 'C:\Users\deept\Downloads\DevOps.3-2\AI-4.2.py'
12345
```

Task 4: Zero-shot vs Few-shot Comparison (Vowel Count)

Prompt :

Write a Python function to count the number of vowels in a given string.

Code :

```
AI-4.2.py  X
AI-4.2.py > ...
1    #Write a Python function to count the number of vowels in a given string.
2    def count_vowels(input_string):
3        vowels = "aeiouAEIOU"
4        count = 0
5        for char in input_string:
6            if char in vowels:
7                count += 1
8        return count
9    # Example usage:
10   input_str = "Hello, World!"
11   print(f"Number of vowels in '{input_str}': {count_vowels(input_str)}")
12   # Output: Number of vowels in 'Hello, World!': 3
13
```

Comparison Explanation

- **Zero-shot**: Basic and longer logic.
- **Few-shot**: More optimized and concise.
- Examples help AI improve clarity and efficiency.

Output :

```
ms\Python\Python313\python.exe   c:\Users\deept\.vscode\extensions\ms-python.debugpy-2025.19.2026011901-win32-x
s\debugpy\launcher' '61501' '--' 'C:\Users\deept\Downloads\DevOps.3-2\AI-4.2.py'
Number of vowels in 'Hello, World!': 3
```

Task 5: Few-shot Prompting (Minimum of Three Numbers) Prompt

:

write a Python function that takes three numbers and returns the minimum value without using min().

Code :

```python
#write a Python function that takes three numbers and returns the minimum value without using min().
def find_minimum(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(find_minimum(3, 1, 2))  # Output: 1
print(find_minimum(-5, -10, -3))  # Output: -10
```

Explanation

- Uses conditional logic.
- Does not use built-in `min()`.
- Covers all comparison cases.

Output :

```
PS C:\Users\deept\Downloads\DevOps                    ept\Downloads\DevOps.3-2'; & 'C:\Users\deept\App
ms\Python\Python313\python.exe' 'c   Open file in editor (ctrl + click)  sions\ms-python.debugpy-2025.19.2026011901-win3
s\debugpy\launcher' '61379' '--' 'C:\Users\deept\Downloads\DevOps.3-2\AI-4.2.py'
1
-10
```