

AI ASSISTED CODING

LAB ASSIGNMENT-4

Name: Sampelly Suhas

HT.NO: 2303A5153

Batch: 21

```
#Task 1
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
n = int(input("Enter a number to check if it is prime: "))
if is_prime(n):
    print(f"{n} is a prime number")
else:
    print(f"{n} is not a prime number")
```

```
Enter a number to check if it is prime: 2
2 is a prime number
```

```
# Task 2
def calculate_sum(lst):
    total = 0
    for num in lst:
        total += num
    return total
# Example usage
result = calculate_sum([1, 2, 3, 4])
print(result) # Output: 10
```

```
Enter a number to check if it is prime: 21
21 is not a prime number
10
```

```

# Task 3
def extract_digits(s):
    digits = ""
    for char in s:
        if char.isdigit():
            digits += char
    return digits

# Test the function with the examples
print(extract_digits("abc123def"))      # Output: "123"
print(extract_digits("hello456world"))  # Output: "456"
print(extract_digits("test789xyz"))     # Output: "789"

```

```

Enter a number to check if it is prime: 32
32 is not a prime number
10
123
456
789

```

```

C: > Users > sampe > zero-short,oneshort,few-short[AIAC-Lab-4].py > ...
1  # Task 4
2  # Zero-shot output
3  def count_vowels(text):
4      vowels = "aeiouAEIOU"
5      count = 0
6      for char in text:
7          if char in vowels:
8              count += 1
9      return count
10 # Few-shot output
11 def count_vowels(s):
12     vowels = 'aeiou'
13     return sum(1 for char in s.lower() if char in vowels)

```

Output comparision:

Zero-shot Approach

Prompt:

"Write a Python function that counts the number of vowels in a string."

Generated Code:

```
python<br>def count_vowels(text):<br> vowels =  
"aeiouAEIOU"<br> count = 0<br> for char in text:<br> if  
char in vowels:<br> count += 1<br> return count<br>
```

Characteristics:

- Basic for-loop approach
- Explicitly includes both uppercase and lowercase vowels
- Straightforward but verbose
- No examples provided

Few-shot Approach

Prompt:

"Write a Python function that counts the number of vowels in a string."

Examples:

1. Input: 'hello', Output: 2 (e, o)
2. Input: 'python', Output: 1 (o)
3. Input: 'education', Output: 5 (e, u, a, i, o)

Generated Code:

```
python<br>def count_vowels(s):<br> vowels =  
'aeiou'<br> return sum(1 for char in  
s.lower() if char in vowels)<br>
```

Characteristics:

- Uses Pythonic generator expression
- Converts to lowercase with .lower()
- More concise (single line return)
- Guided by 3 examples

Key Differences & How Examples Helped:

1. **Code Efficiency:** Few-shot produced more concise code using sum() with a generator expression.
2. **Case Handling:** Zero-shot explicitly lists both cases; Few-shot uses .lower() for simplification.
3. **Readability:** Few-shot version is more Pythonic and readable.
4. **Learning from Examples:** The AI recognized from the examples that:
 - Only lowercase vowels needed counting
 - A counting approach was required
 - The output should be a simple integer

C: > Users > sampe > zero-short,oneshort,few-short[AIAC-Lab-4].py > ...

```
1  # Task5
2  def find_minimum(a, b, c):
3      if a <= b and a <= c:
4          return a
5      elif b <= a and b <= c:
6          return b
7      else:
8          return c
9
10 # Test with the examples
11 print(find_minimum(5, 2, 8))    # Output: 2
12 print(find_minimum(10, 15, 3)) # Output: 3
13 print(find_minimum(7, 7, 1))   # Output: 1
```

PS C:\Users\sampe> & 'c:\Users\sampe\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\sampe\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '54750' '--' 'C:\Users\sampe\zero-short,oneshort,few-short[AIAC-Lab-4].py'

2
3
1