Assignment 4.3

2303A51250

l.Indu

Batch – 18

Task 1: Zero-Shot Prompting – Leap Year Check

Scenario

Zero-shot prompting involves giving instructions without providing examples.

Task Description

Use zero-shot prompting to instruct an AI tool to generate a Python function that:

• Accepts a year as input

• Checks whether the given year is a leap year

• Returns an appropriate result

Note: No input-output examples should be provided in the prompt.

Expected Output

• AI-generated leap year checking function

• Correct logical conditions

• Sample input and output

• Screenshot of AI-generated response (if required)

Prompt :

Write a Python function that accepts a year as input and checks whether it is a leap year.

taking input from user

Code :

```python
# Write a Python function that accepts a year as input and checks whether it is a leap year.

def is_leap_year(year):
    if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
        return "Leap Year"
    else:
        return "Not a Leap Year"

# taking input from user
year = int(input("Enter a year: "))
print(is_leap_year(year))

```

output

```
PS C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe c:/U
sers/user1/Desktop/python/leapyear.py
Enter a year: 2004
Leap Year
PS C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe c:/U
sers/user1/Desktop/python/leapyear.py
Enter a year: 2025
```

Analysis

1.I used zero-shot prompting by giving only the task instruction without examples.

2.GitHub Copilot generated a correct Python function using proper leap year conditions.

3.The output was logically correct

Task 2: One-Shot Prompting – Centimeters to Inches Conversion

Scenario

One-shot prompting guides AI using a single example.

Task Description

Use one-shot prompting by providing one input-output example to generate a Python

function that:

• Converts centimeters to inches

• Uses the correct mathematical formula

Example provided in prompt:

Input: 10 cm → Output: 3.94 inches

Expected Output

• Python function with correct conversion logic

• Accurate calculation

• Sample test cases and outputs

Prompt :

#Write a Python function to convert centimeters to inches.

Example:

Input: 10 cm

Output: 3.94 inches

taking input from user

Code :

centimentstoinch.py > ...

```python
# Write a Python function to convert centimeters to inches.
# Example:
# Input: 10 cm
# Output: 3.94 inches
def cm_to_inches(cm):
    inches = cm / 2.54
    return round(inches, 2)
# taking input from user
cm = float(input("Enter length in centimeters: "))
print(f"{cm} cm is equal to {cm_to_inches(cm)} inches")

```

Output:

```
S C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe
ers/user1/Desktop/python/centimentstoinch.py
nter length in centimeters: 10
0.0 cm is equal to 3.94 inches
S C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe
ers/user1/Desktop/python/centimentstoinch.py
nter length in centimeters: 25.4
5.4 cm is equal to 10.0 inches
S C:\Users\user1\Desktop\python>
```

Analysis

1.I provided a single input-output example in the prompt to guide the AI.

2.The example helped GitHub Copilot understand the correct conversion formula.

3.The generated code was clear, accurate, and followed the expected logic.

Task 3: Few-Shot Prompting – Name Formatting

Scenario

Few-shot prompting improves accuracy by providing multiple examples.

Task Description

Use few-shot prompting with 2–3 examples to generate a Python function that:

• Accepts a full name as input

• Formats it as “Last, First”

Example formats:

• "John Smith" → "Smith, John"

• "Anita Rao" → "Rao, Anita"

Expected Output

• Well-structured Python function

• Output strictly following example patterns

• Correct handling of names

• Sample inputs and outputs

Prompt :

#Write a Python function that accepts a full name and formats it as "Last, First".

Examples:

Input: John Smith

Output: Smith, John

Input: Anita Rao

Output: Rao, Anita

Code :

```python
reversename.py > format_name
# Write a Python function that accepts a full name and formats it as "Last, First"
# Examples:
# Input: John Smith
# Output: Smith, John
# Input: Anita Rao
# Output: Rao, Anita
def format_name(full_name):
    parts = full_name.strip().split()
    if len(parts) != 2:
        return "Invalid name format"
    first, last = parts
    return f"{last}, {first}"
name = input("Enter full name: ")
print(format_name(name))
```

Output:

```
sers/user1/Desktop/python/centimentstoinch.py
PS C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe c:
sers/user1/Desktop/python/reversename.py
Enter full name: john smith
smith, john
PS C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe c:
sers/user1/Desktop/python/reversename.py
Enter full name: anitha rao
rao, anitha
PS C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe c:
sers/user1/Desktop/python/reversename.py
Enter full name: rahul verma
verma, rahul
```

Analysis:

1.I provided multiple examples to clearly show the required input and output format.

2.The examples helped GitHub Copilot generate a well-structured and accurate function.

3.The output follows the “Last, First” pattern as shown in the examples.

Task 4: Comparative Analysis – Zero-Shot vs Few-Shot

Scenario

Different prompt strategies may produce different code quality.

Task Description

• Use zero-shot prompting to generate a function that counts vowels in a string

• Use few-shot prompting for the same problem

• Compare both outputs based on:

o Accuracy

o Readability

o Logical clarity

Expected Output

• Two vowel-counting functions

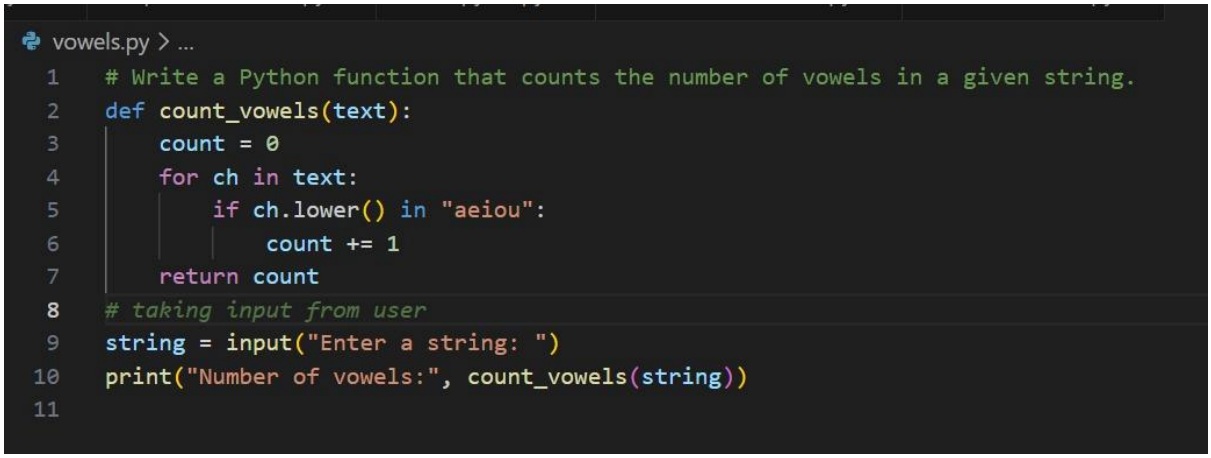
• Comparison table or short reflection paragraph

• Conclusion on prompt effectiveness

Prompt :

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

Code :



Output

Analysis :

1.In zero-shot prompting, the AI generated a basic and correct solution but with minimal structure.

2.When examples were provided, the AI produced clearer and more readable code.

3.Few-example prompting helped the AI better understand expected output.

Task 5: Few-Shot Prompting – File Handling

Scenario

File processing requires clear logical understanding.

Task Description

Use few-shot prompting to generate a Python function that:

• Reads a .txt file

• Counts the number of lines in the file

• Returns the line count

Expected Output

• Working Python file-processing function

• Correct line count

• Sample .txt input and output

• AI-assisted logic explanation

Prompt :

Write a Python function that reads a text file and counts the number of lines.

Examples:

File content:

Hello

World

Output: 2

File content:

Line one

Line two

Line three

Output: 3

Code:

```
linecount.py > ...
1   # Write a Python function that reads a text file and counts the number of lines.
2   # Examples:
3   # File content:
4   # Hello
5   # World
6   # Output: 2
7   # File content:
8   # Line one
9   # Line two
10  # Line three
11  # Output: 3
12  def count_lines(filename):
13      with open(filename, 'r') as file:
14          return len(file.readlines())
15
16
17  file_name = input("Enter file name: ")
18  print("Number of lines:", count_lines(file_name))
19
```

Output:

```
PS C:\Users\user1\Desktop\python> & C:/Users/user1/AppData/Local/Programs/Python/Python313/python.exe c:/Users/user1/Desktop/python/linecount.p
y
Enter file name: linecount.py
Number of lines: 19
```

Analysis:

1.I provided multiple examples showing file content and expected line count.

2.These examples helped GitHub Copilot understand the required file-handling logic.

3.The generated function was simple, readable, and correctly counted lines.