AI ASSISTED CODING

LAB ASSIGNMENT – 4

Student Name : **Sarayu**

Hall Ticket Number : **2303A51842**

Course : **B.Tech**

1. Zero-Shot Prompting – Leap Year Check

Prompt Used

Write a Python function to check whether a given year is a leap year.

Explanation

In zero-shot prompting, no examples are given. The AI uses its prior knowledge of leap year rules to generate the logic.

Steps Followed

Step 1: Read the year value.

Step 2: Check divisibility by 4.

Step 3: Eliminate years divisible by 100.

Step 4: Include years divisible by 400.

Step 5: Apply logical operators.

Step 6: Return the final decision.

Program Code

```
def is_leap_year(year):
  if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
    return True
  return False
```

Output

Input: 2024 → Output: True

Input: 1900 → Output: False

Input: 2000 → Output: True

Observation

The program works correctly for leap year identification.

Result

Thus, the given year is correctly classified.

2. One-Shot Prompting – Centimeters to Inches Conversion

Prompt Used

Convert centimeters to inches. Example: 10 cm → 3.94 inches.

Explanation

One example guides the AI to apply the correct conversion formula.

Steps Followed

Step 1: Accept centimeter value.

Step 2: Use conversion factor 2.54.

Step 3: Perform division.

Step 4: Store the value.

Step 5: Return result.

Step 6: Verify output.

Program Code

```
def cm_to_inches(cm):
  return cm / 2.54
```

Output

Input: 10 → Output: 3.94

Input: 25 → Output: 9.84

Observation

The conversion is accurate.

Result

Hence, the centimeter value is successfully converted.

3. Few-Shot Prompting – Name Formatting

Prompt Used

Format a full name as 'Last, First' using examples.

Explanation

Multiple examples help the AI understand the formatting pattern clearly.

Steps Followed

Step 1: Read full name.

Step 2: Split name into parts.

Step 3: Identify first and last name.

Step 4: Rearrange order.

Step 5: Add comma.

Step 6: Return formatted name.

Program Code

```
def format_name(name):
  first, last = name.split()
  return f"{last}, {first}"
```

Output

Input: John Smith → Output: Smith, John

Input: Anita Rao → Output: Rao, Anita

Observation

The program formats names correctly.

Result

Thus, the required name format is obtained.

4. Comparative Analysis – Vowel Count

Prompt Used

Count the number of vowels in a string using prompting techniques.

Explanation

This task compares the effectiveness of zero-shot and few-shot prompting.

Steps Followed

Step 1: Read input string.

Step 2: Define vowels.

Step 3: Initialize counter.

Step 4: Traverse string.

Step 5: Increment counter.

Step 6: Display result.

Program Code

```
def count_vowels(text):
  vowels = 'aeiouAEIOU'
  count = 0
  for c in text:
    if c in vowels:
      count += 1
  return count
```

Output

Input: hello → Output: 2

Input: education → Output: 5

Observation

Few-shot prompting improves clarity.

Result

Hence, vowel count is obtained correctly.

5. Few-Shot Prompting – File Handling

Prompt Used

Count the number of lines in a text file.

Explanation

Few-shot prompting helps in generating correct file handling logic.

Steps Followed

Step 1: Provide file name.

Step 2: Open file in read mode.

Step 3: Read contents.

Step 4: Count lines.

Step 5: Return count.

Step 6: Print output.

Program Code

```
def count_lines(filename):
  with open(filename, 'r') as f:
    return len(f.readlines())
```

Output

File with 3 lines → Output: 3

Empty file → Output: 0

Observation

The function works correctly for valid files.

Result

Thus, the number of lines is calculated successfully.