Htno:2403A51327

Name: Sanjana Karnakanti

Batch no: 24BTCAICSB13

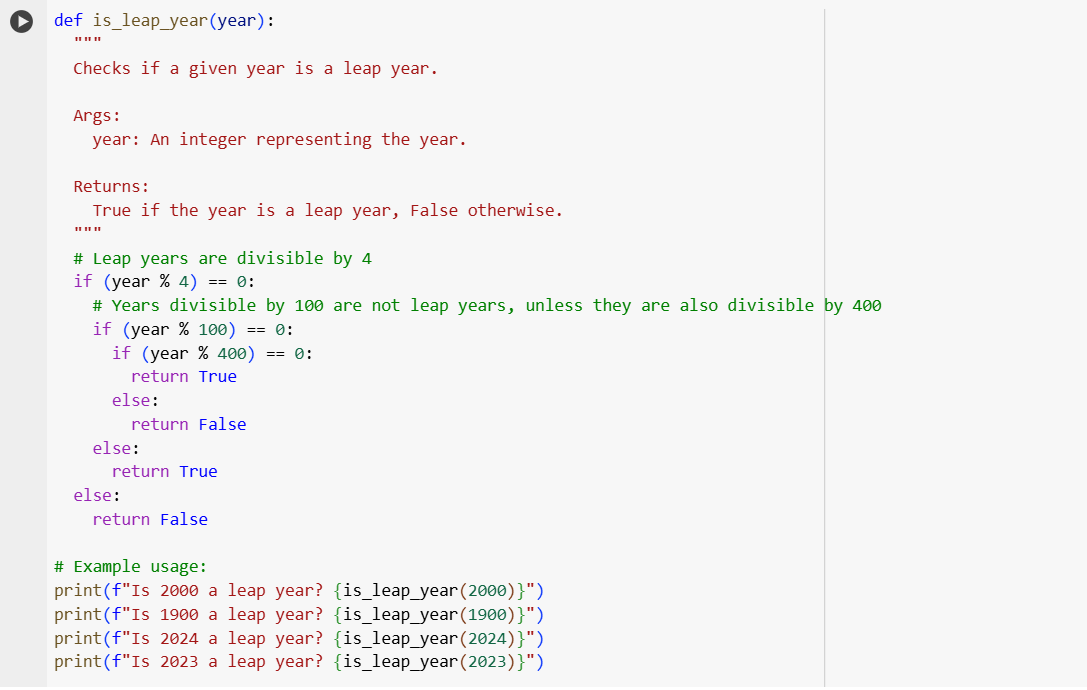
Date:20-08-2025

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

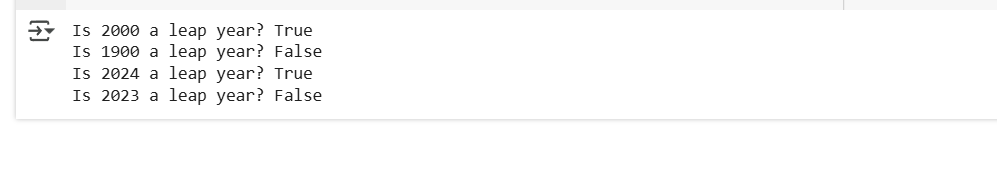
**Task 1:**

**Prompt:** write a python function that checks whether a given year is a leap year or not with all possibilities.

**Code:**

****

**OUTPUT:**

****

**Observation:**

This code defines a function is\_leap\_year that checks if a given year is a leap year. It follows the rules: divisible by 4, but not by 100 unless also divisible by 400. The function returns True if it's a leap year and False otherwise. The example usage demonstrates checking a few different years.

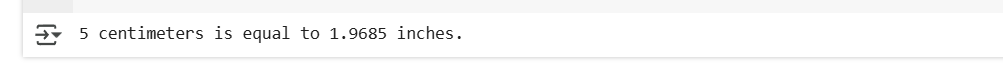
**TASK 2:  
Prompt:** write a python function that converts centimeters to inche

example: 5cm=1.9685inches

**Code:**

****

**Output:**

****

**Observation:**

This code defines a function cm\_to\_inches that takes a value in centimeters as input. Inside the function, it converts the centimeters to inches by dividing by 2.54 (since 1 inch equals 2.54 cm). Finally, it returns the calculated value in inches. The example usage shows how to call the function and print the result.

**TASK 3:**

**Prompt:**

Write a python function to Format full names as “Last, First”

Examples:  
Input: John Smith  
Output: Smith, John

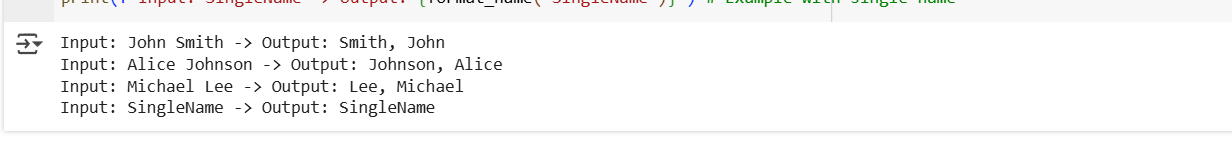
Input: Alice Johnson  
Output: Johnson, Alice

Input: Michael Lee  
Output: Lee, Michael

**Code:**

****

**Output:**

****

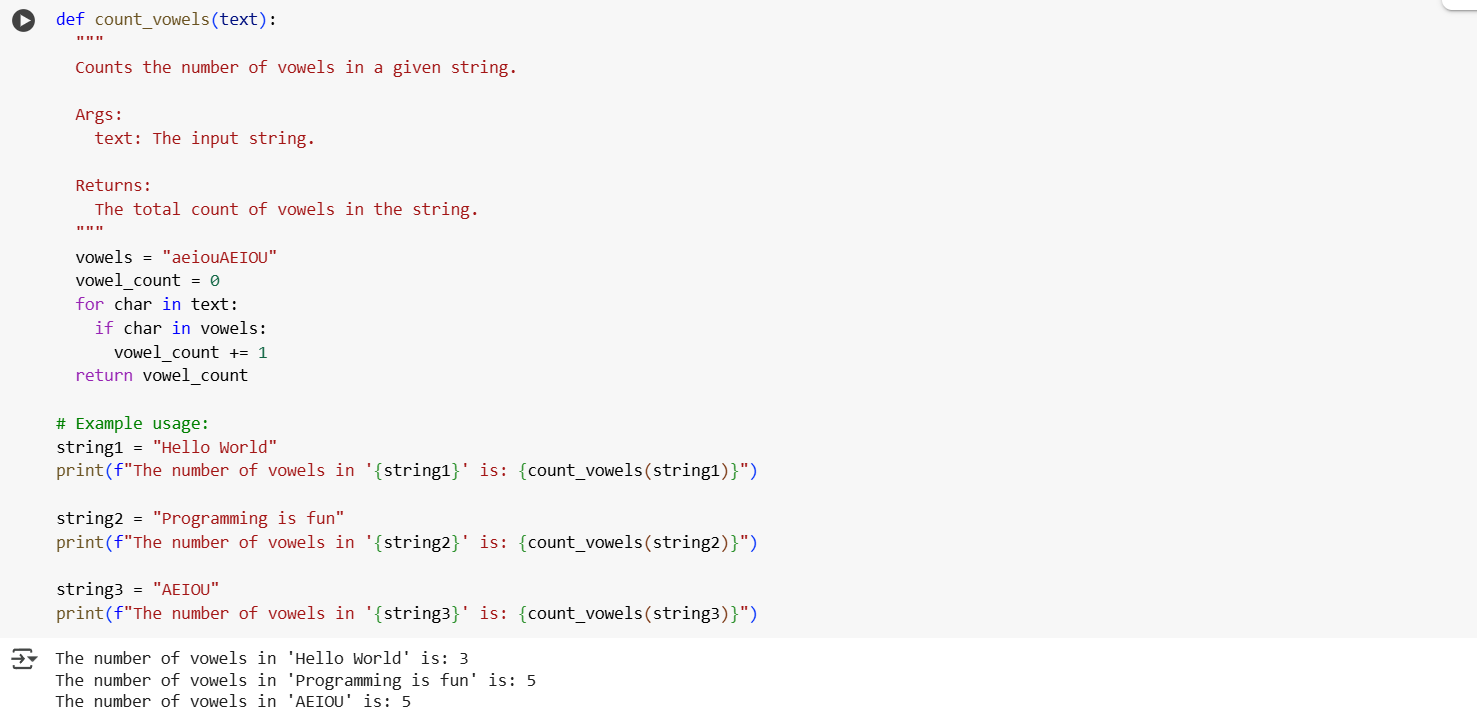
**Observation:**

This code defines a function format\_name that takes a full name as input. It splits the name into parts based on spaces. If there are two or more parts, it rearranges them to put the last part first, followed by a comma and the rest of the parts. If there's only one part, it returns the original name.

**TASK 4 :**

**Zero-shot Prompt:** Write a function that counts the number of vowels in a string.

**Code and Output:**

****

**Few-shot Prompt:**

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

Examples:  
Input: "hello"  
Output: 2

Input: "sky"  
Output: 0

Input: "Education"  
Output: 5

**Code and Output:**

****

**Observation Table:**

| **Aspect** | **Zero-Shot** | **Few-Shot** |
| --- | --- | --- |
| Input examples | None | 1 or more related examples |
| Flexibility | High, but less predictable style | More control over style & structure |
| Consistency | May vary in formatting | More consistent with examples |
| Ease of setup | Simple, fast to create | Requires relevant examples |
| Best use case | Standard/common tasks | Tasks needing consistent style or logic patterns |

**TASK 5:**

**Prompt:** Write a function that reads a .txt file and returns the number of lines.

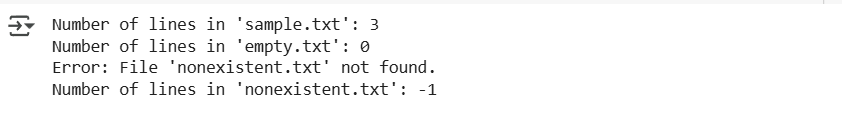
Example 1:  
Input: A file named sample.txt containing:  
Hello world  
This is a test file  
It has three lines  
Output: 3

Example 2:  
Input: A file named empty.txt containing no content  
Output: 0

**Code:**

****

**Output:**

****

**Observation:**

The count\_lines\_in\_file function opens a specified file using a with statement, which ensures the file is closed automatically. It then reads all the lines of the file into a list using readlines(). The number of lines is determined by getting the length of this list using len(). A try...except FileNotFoundError block is included to catch errors if the file does not exist.