AI ASSISTED CODING

LAB ASSIGNMENT-4.1

Name: P.AJAY

Roll no: 2403A510B4

Batch: 05

Department: CSE

Task #1 – Zero-Shot Prompting with Conditional Validation

Objective

Use zero-shot prompting to instruct an AI tool to generate a function that validates an Indian mobile number.

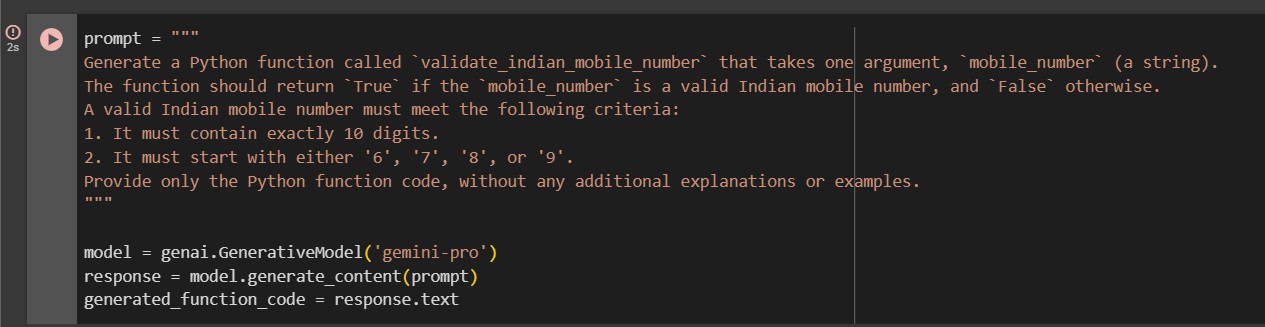
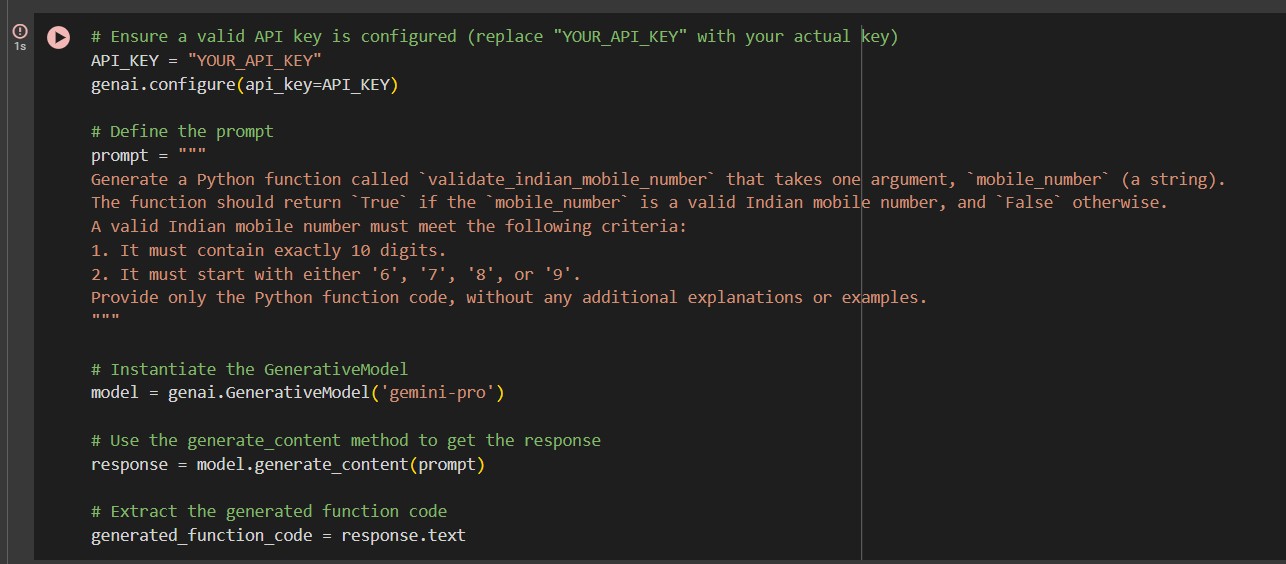
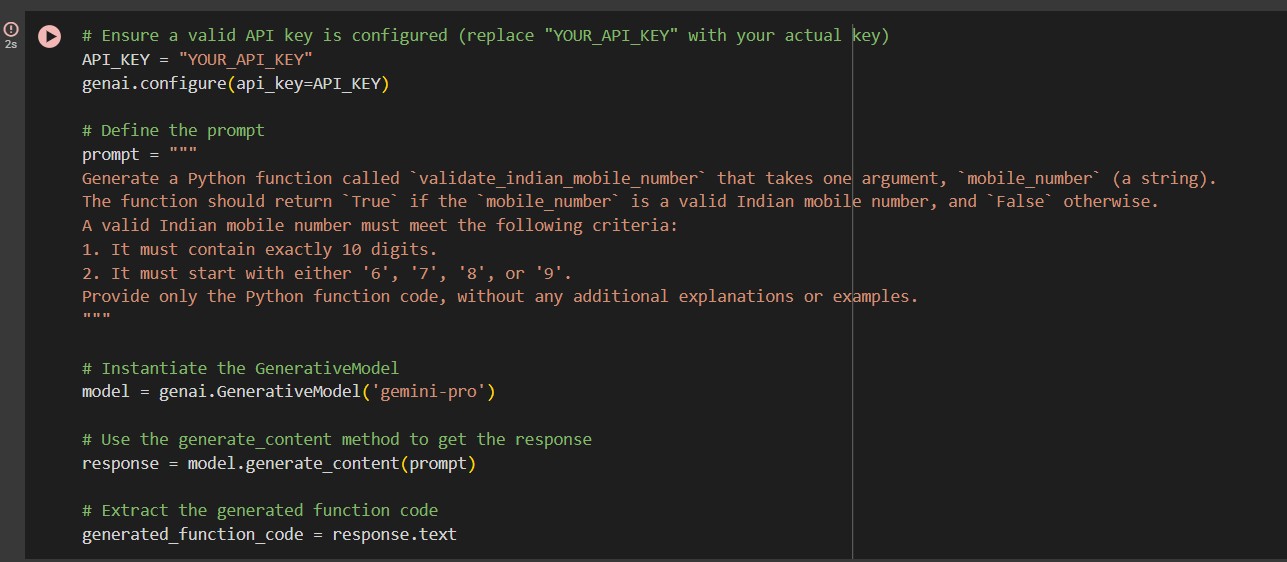
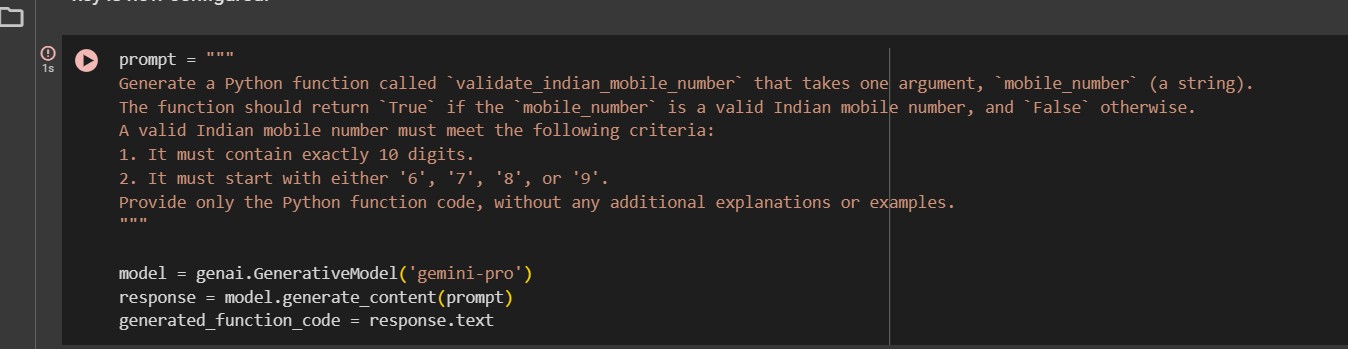
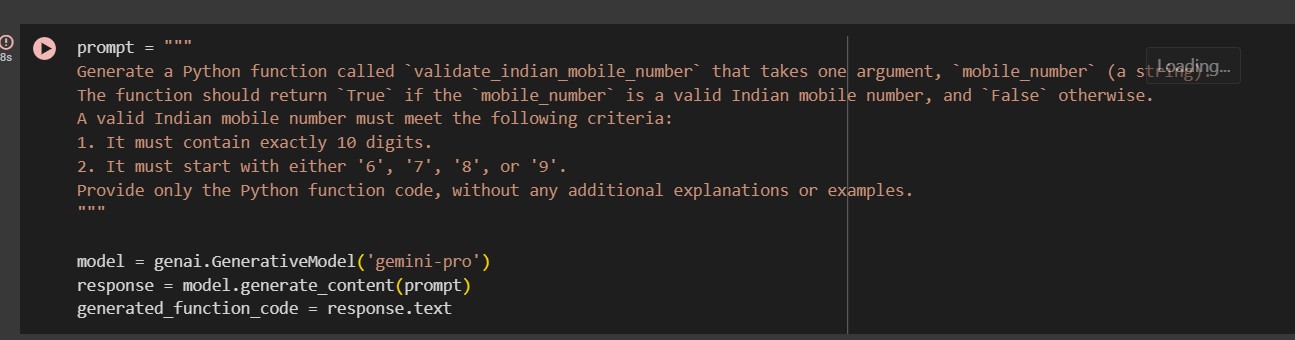
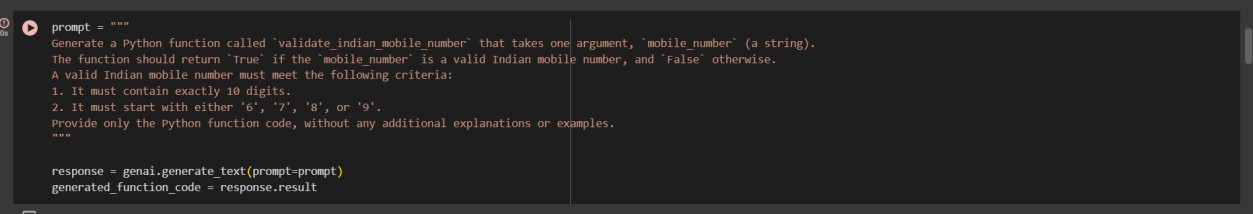
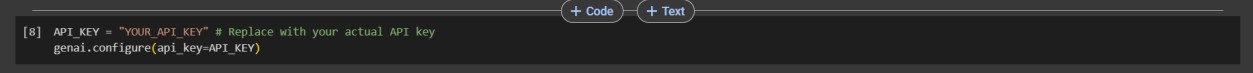
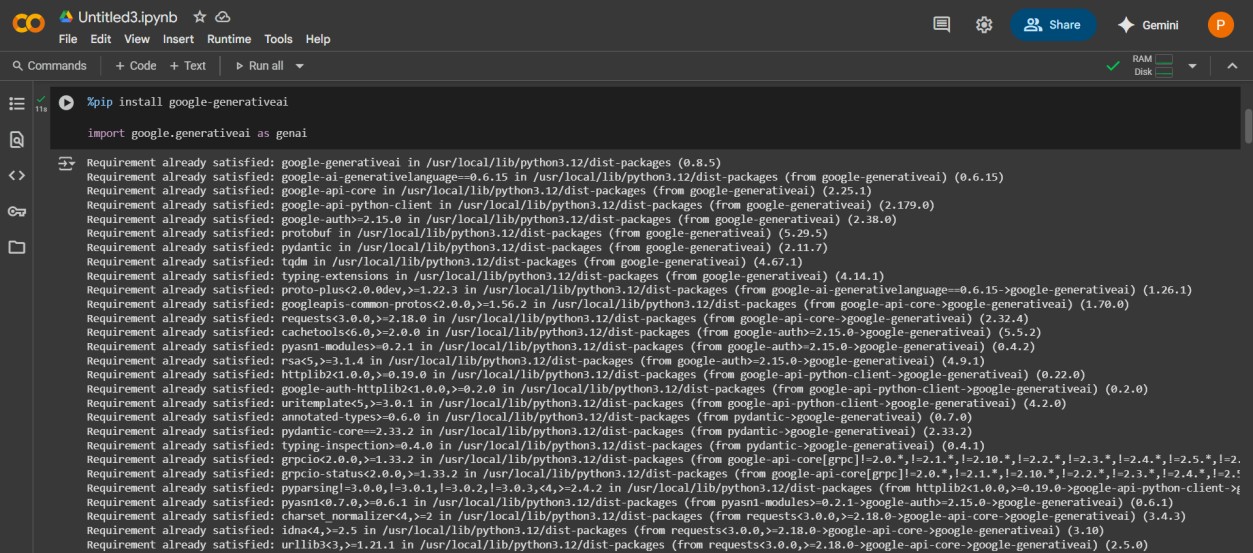
Requirements

• The function must ensure the mobile number:

o Starts with 6, 7, 8, or 9 o Contains exactly 10 digits

Expected Output

* A valid Python function that performs all required validations without using any input-output examples in the prompt.



Task #2 – One-Shot Prompting with Edge Case Handling

Objective

Use one-shot prompting to generate a Python function that calculates the factorial of a number.

Requirements

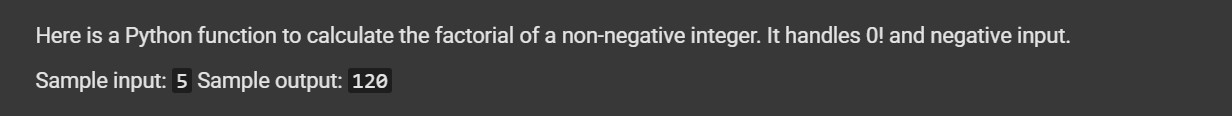
* Provide one sample input-output pair in the prompt to guide the

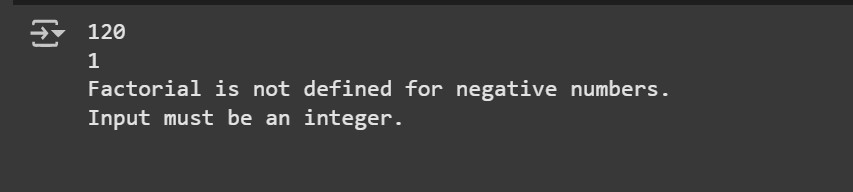
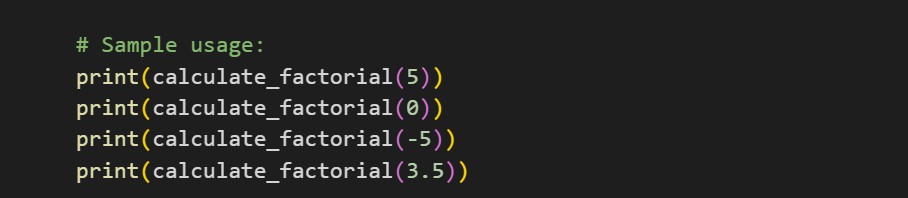
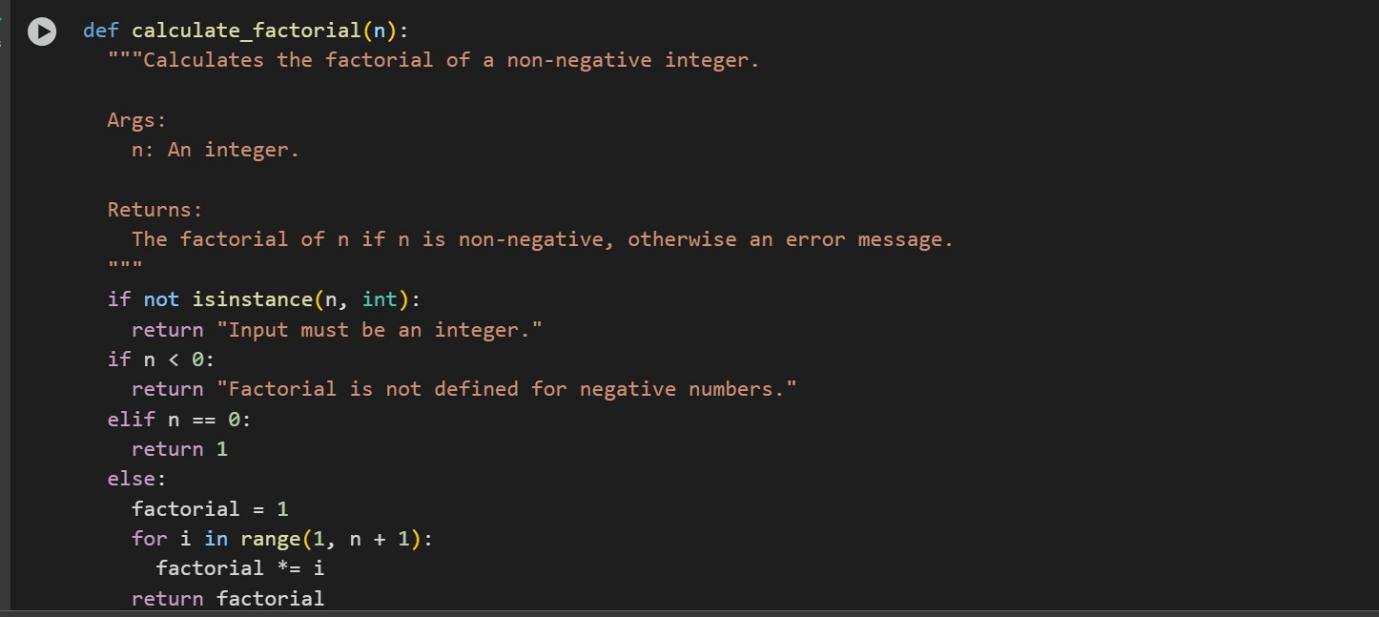
AI.

* The function should handle:
* 0! correctly
* Negative input by returning an appropriate message

Expected Output

• A Python function with correct factorial logic and edge case handling, generated from a single example.





# Task #3 – Few-Shot Prompting for Nested Dictionary Extraction

Objective

Use few-shot prompting (2–3 examples) to instruct the AI to create a function that parses a nested dictionary representing student information.

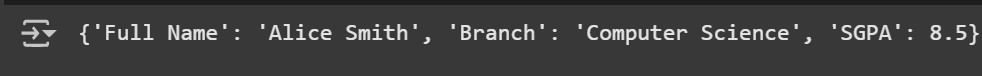
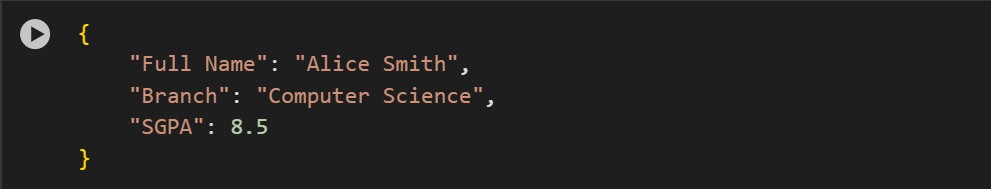
Requirements

* The function should extract and return:

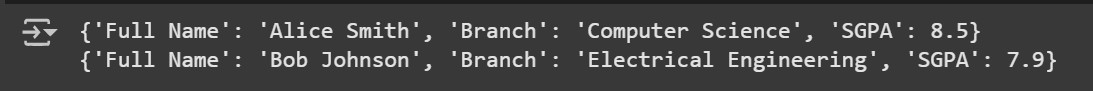
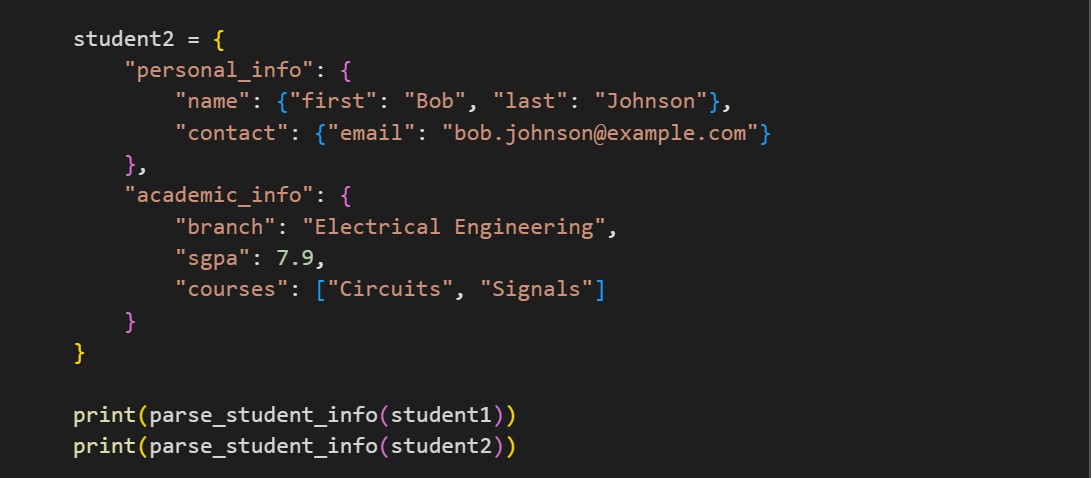
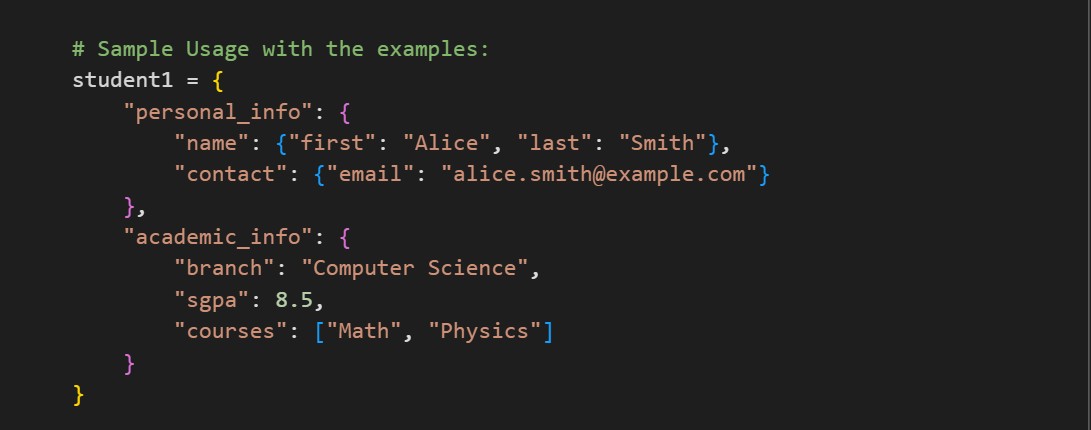
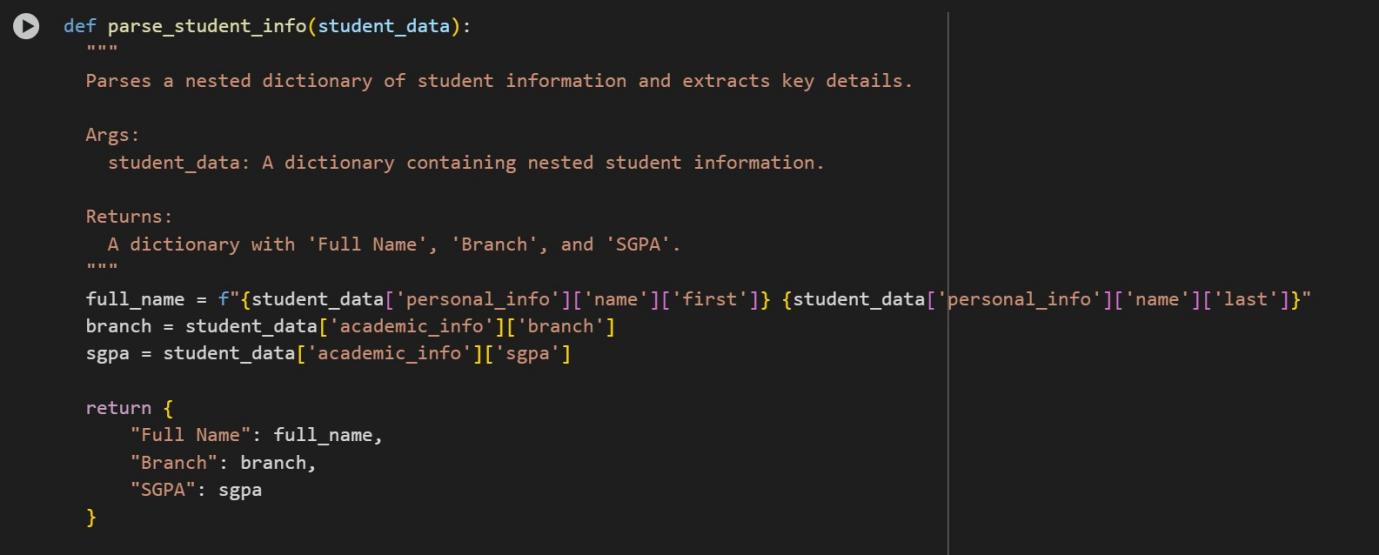
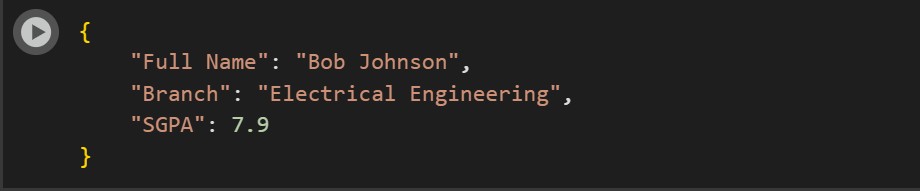
o Full Name o Branch o SGPA Expected Output

* A reusable Python function that correctly navigates and extracts values from nested dictionaries based on the provided examples.

Example 1:



Example 2:



# Task #4 – Comparing Prompting Styles for File Analysis

Objective

Experiment with zero-shot, one-shot, and few-shot prompting to generate functions for CSV file analysis.

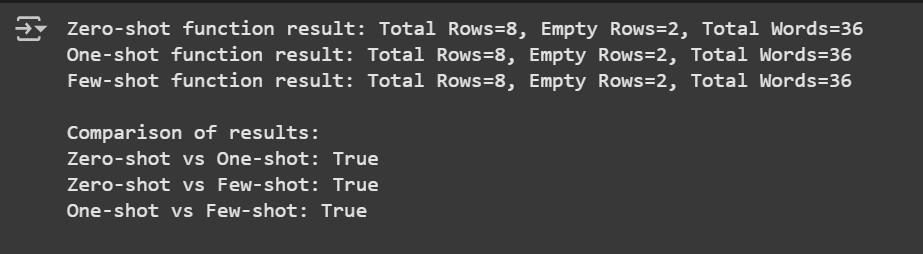
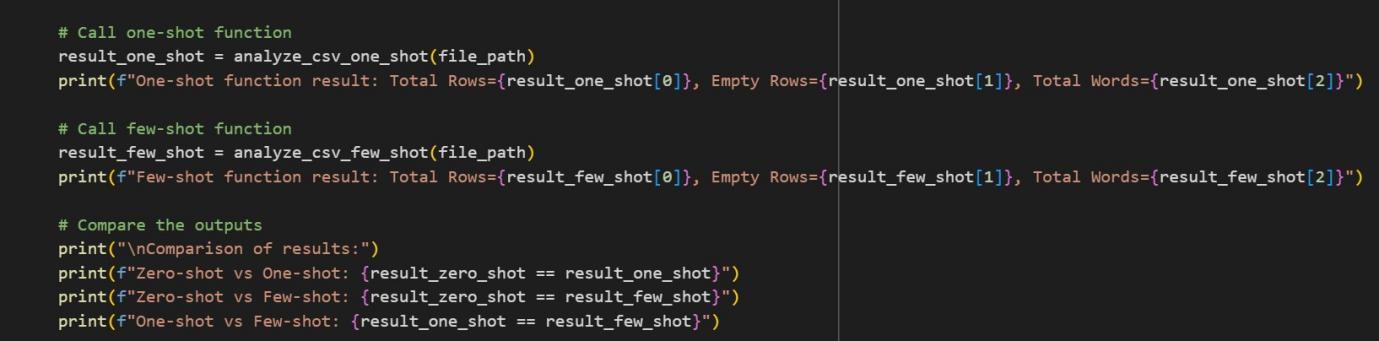
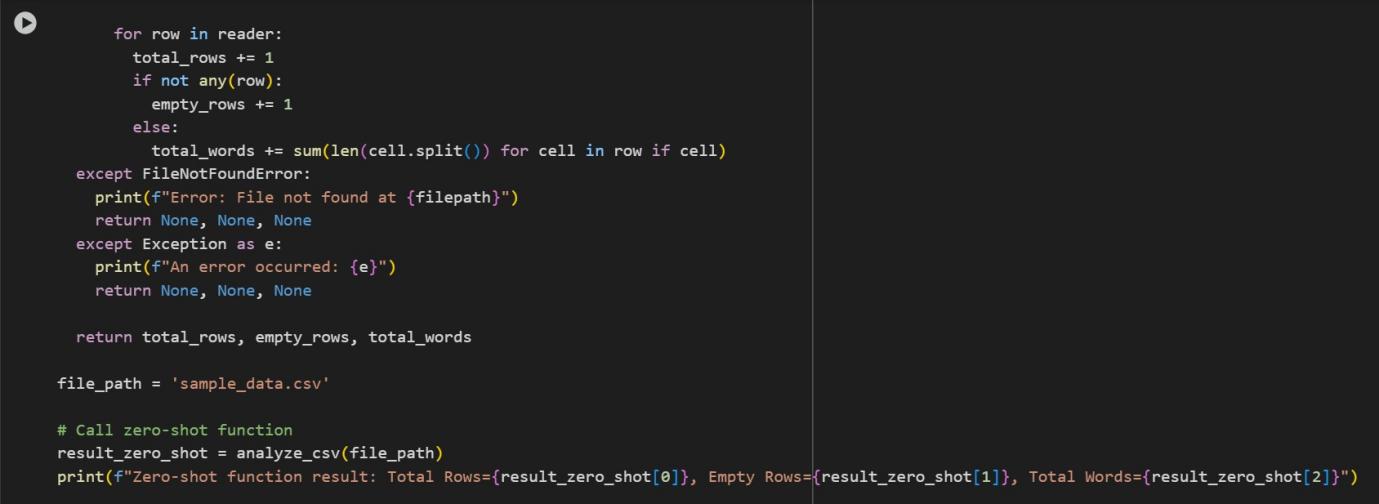
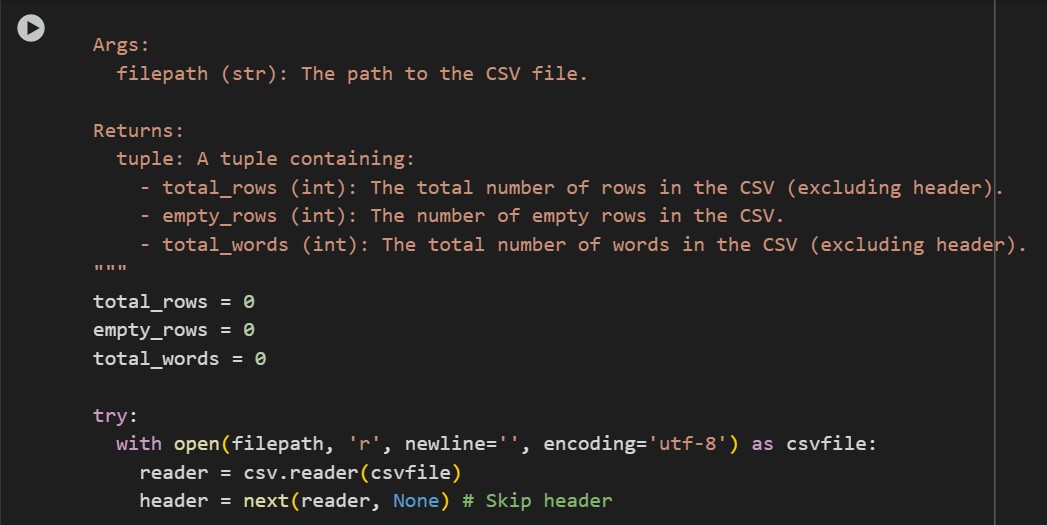
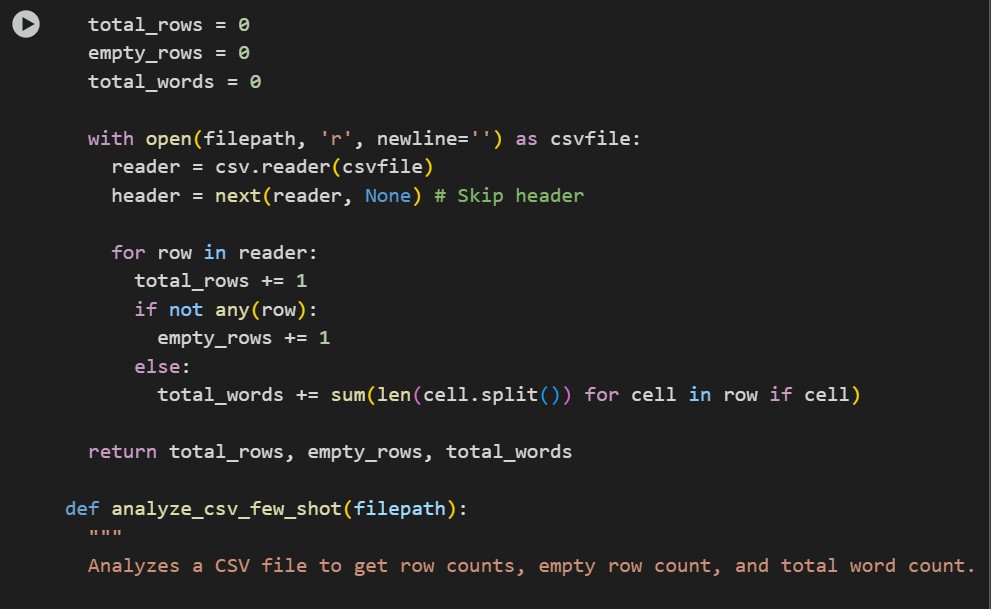
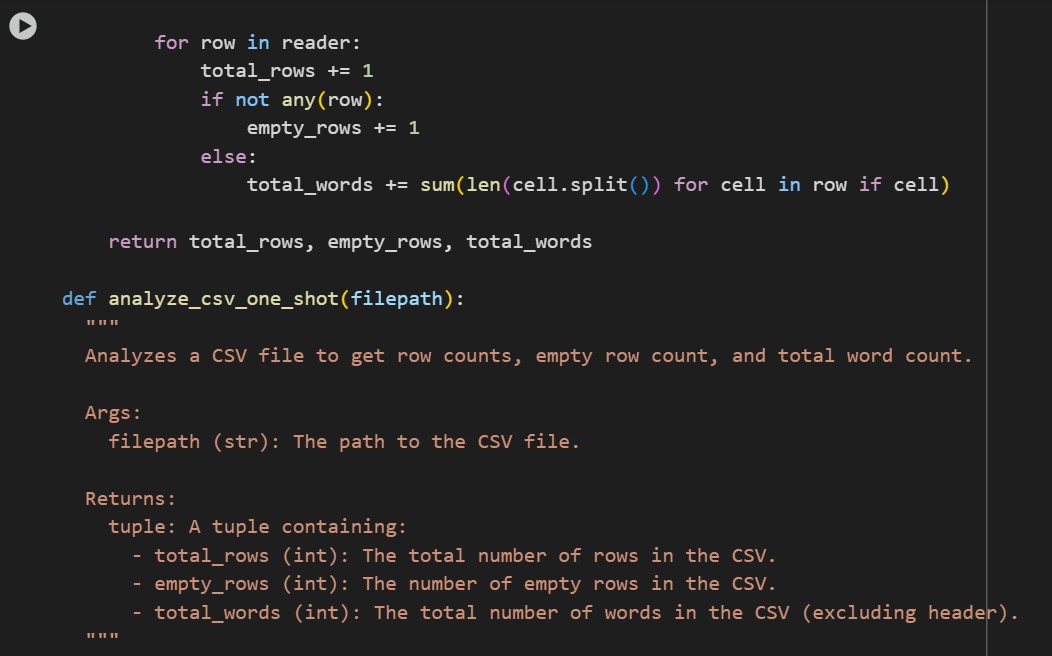
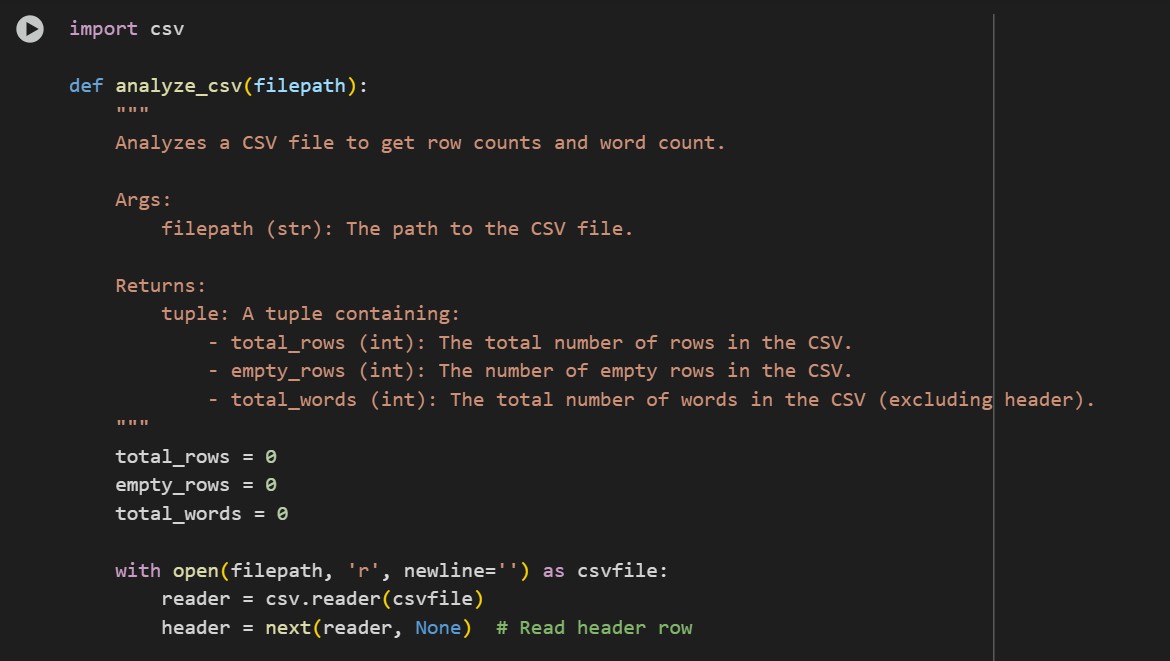
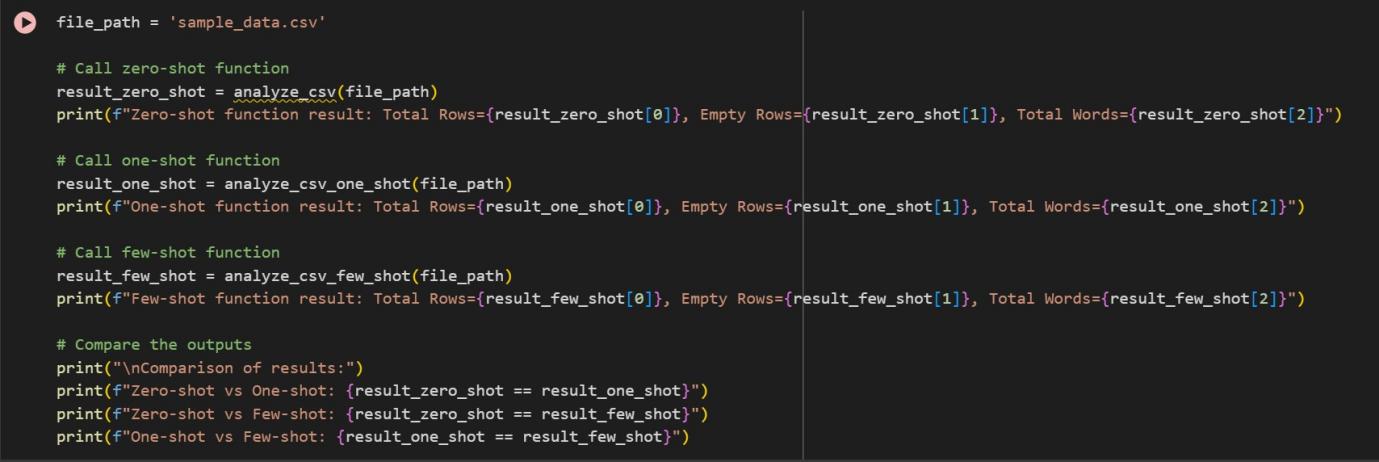
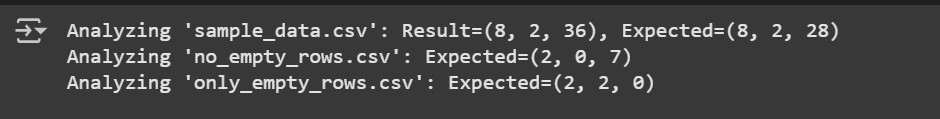
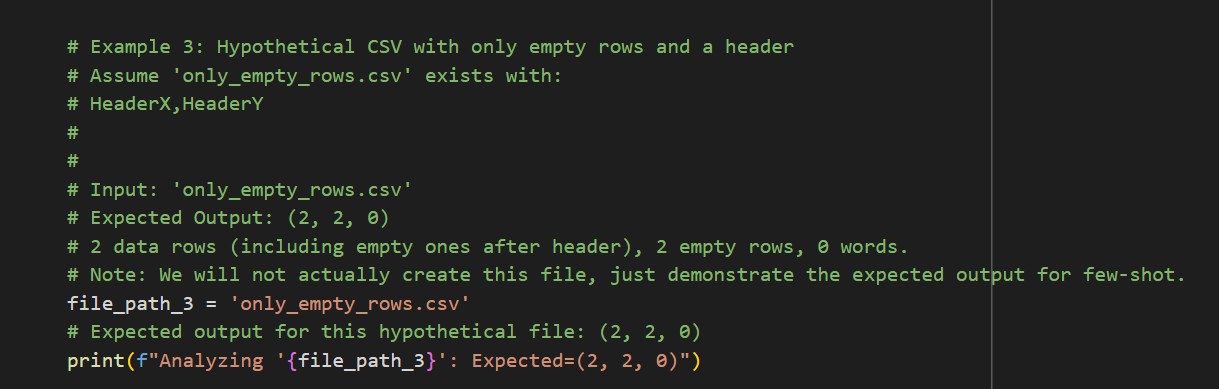
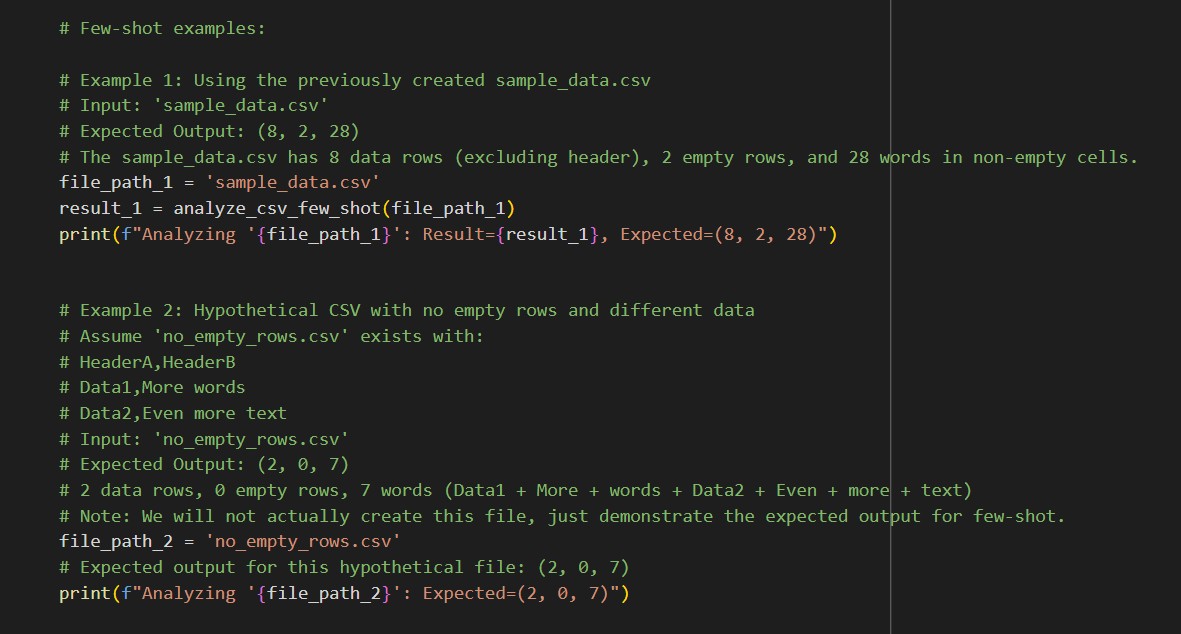
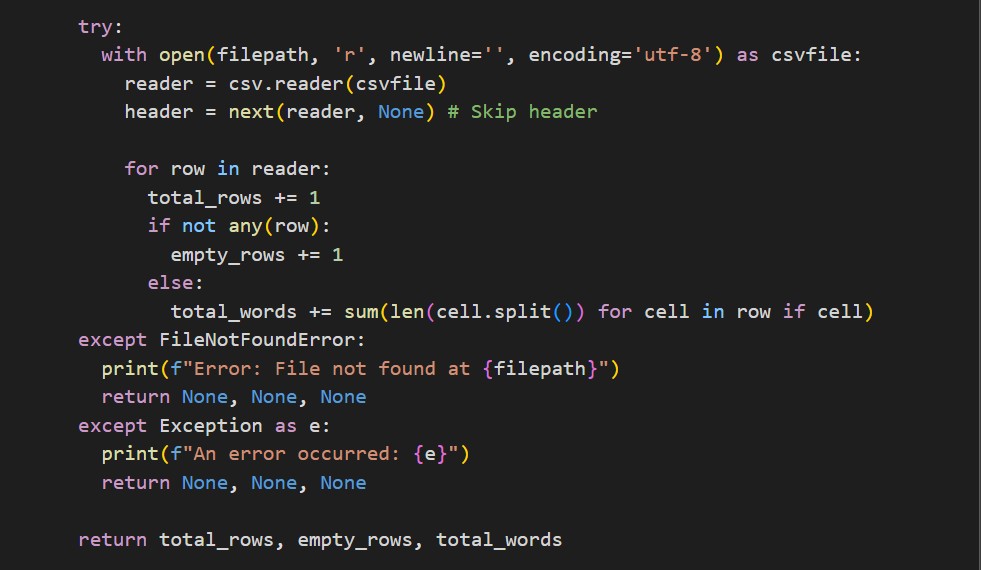
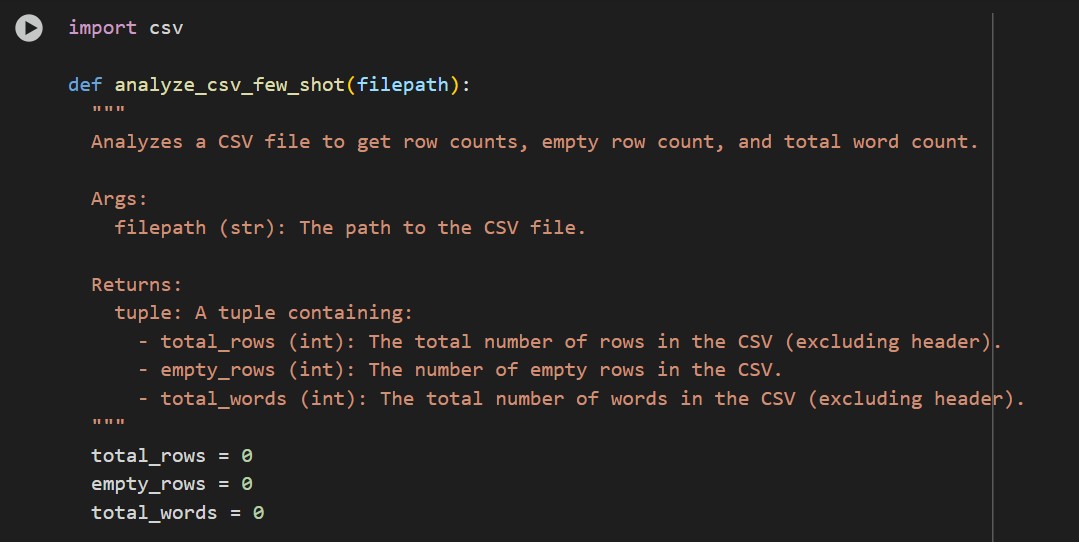
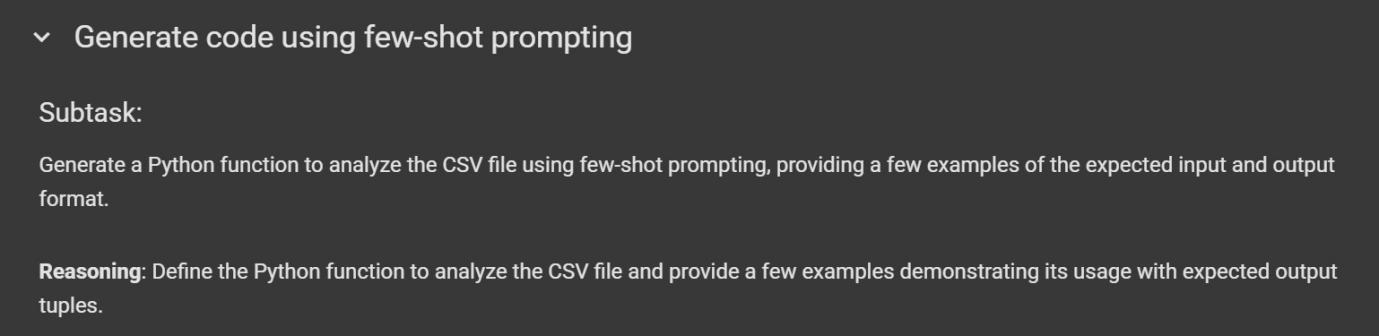
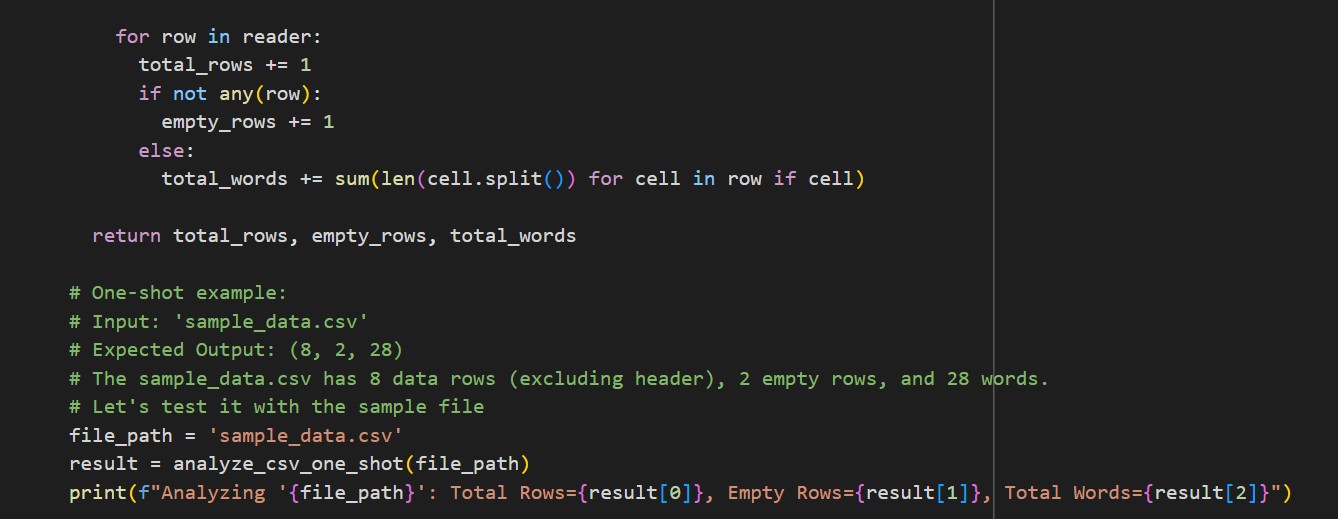
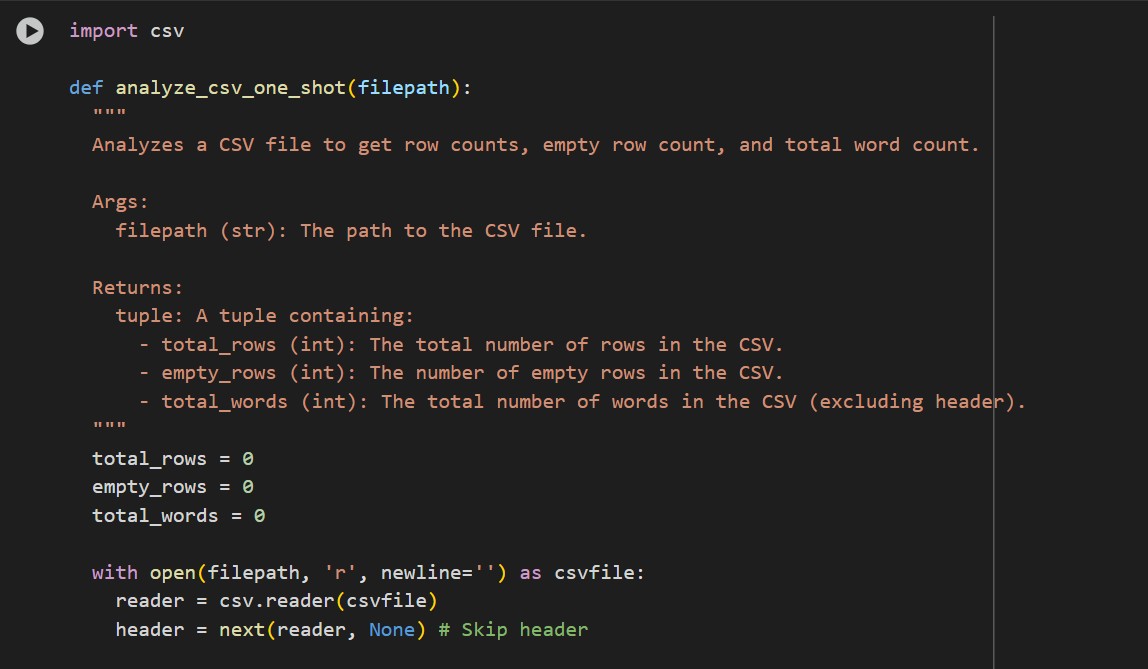
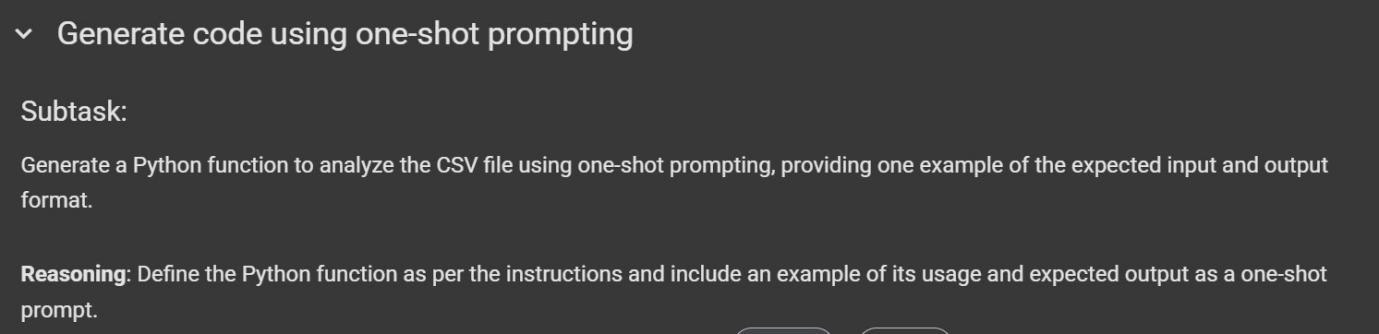
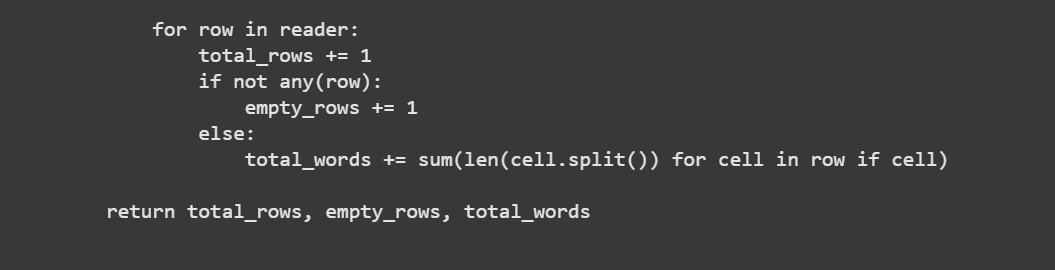
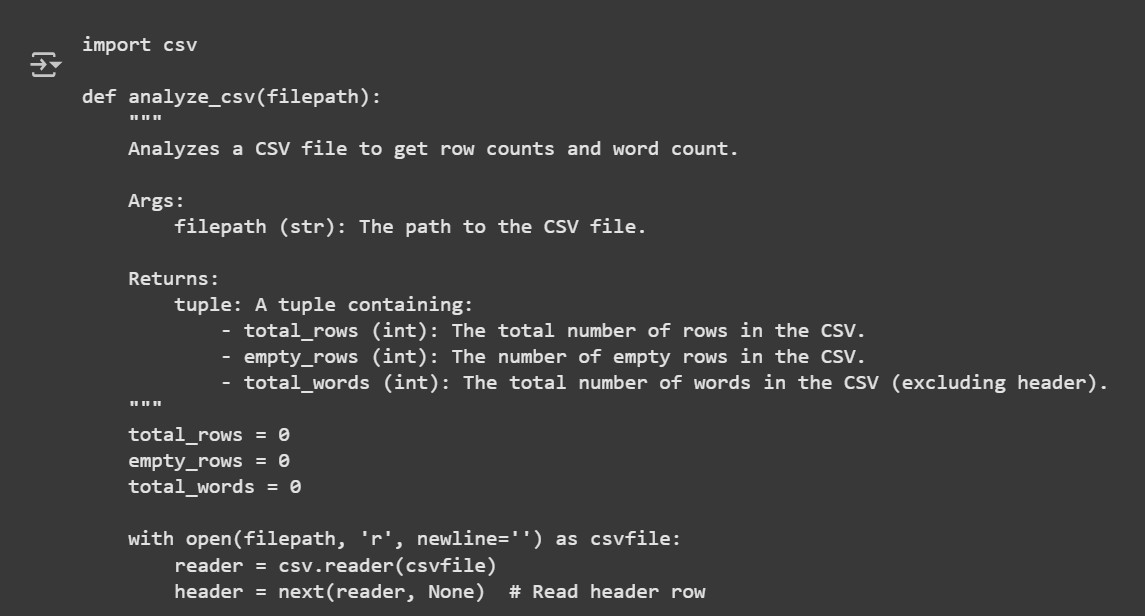
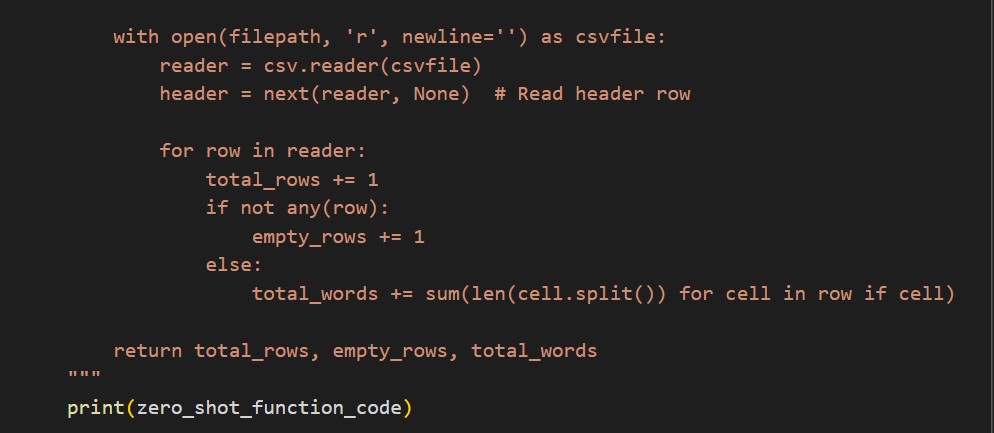
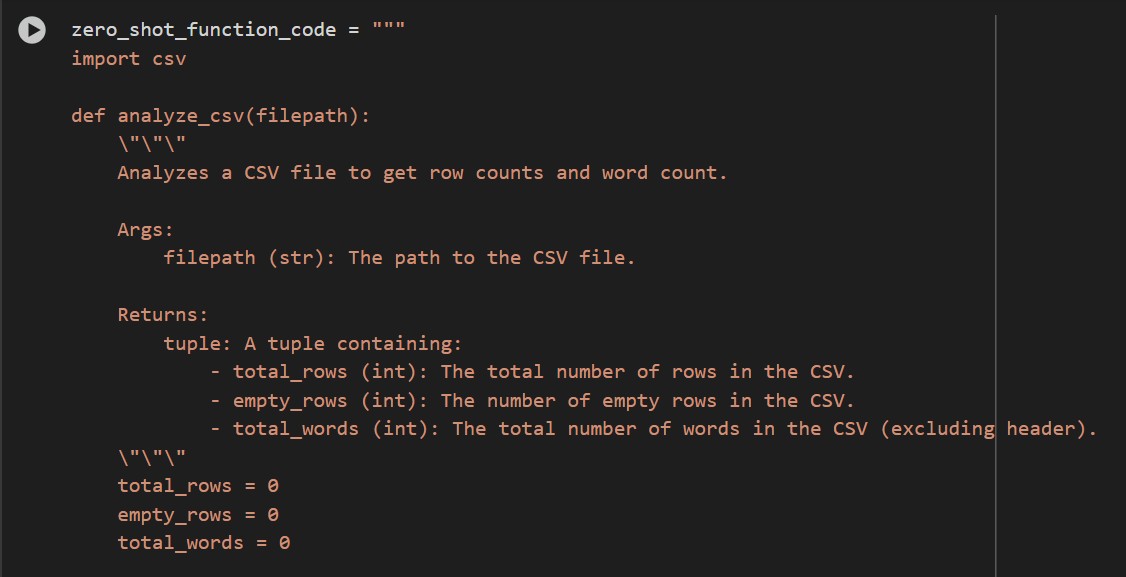
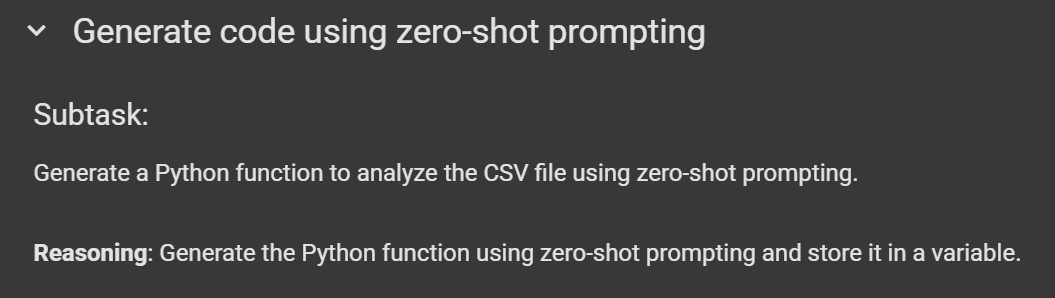
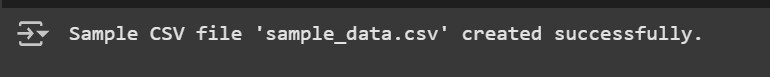
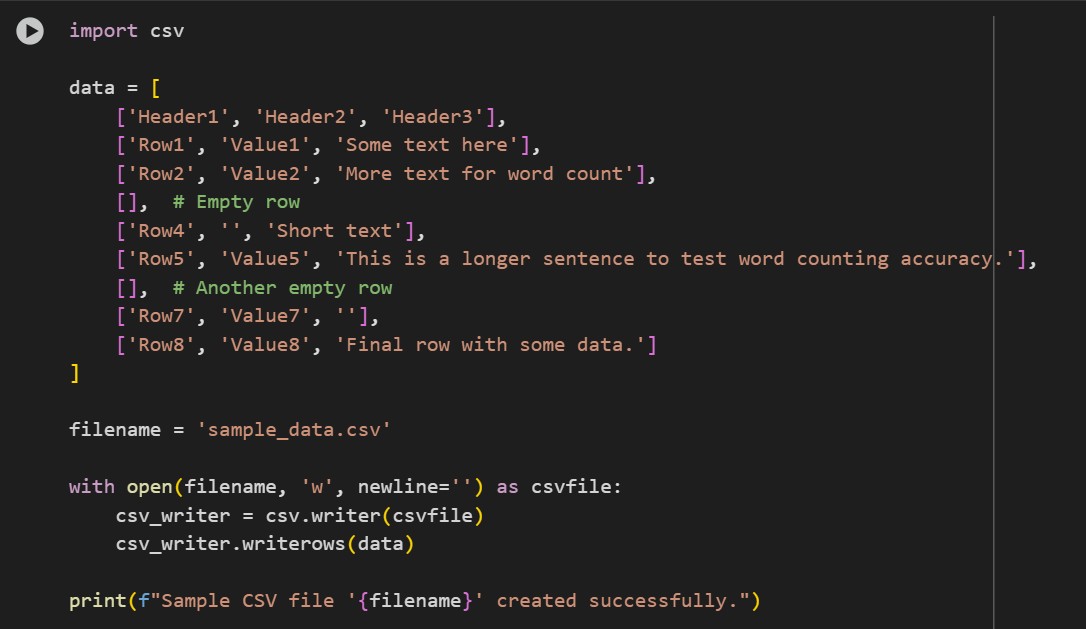
Requirements

• Each generated function should:

o Read a .csv file o Return the total number of rows o Count the number of empty rows o Count the number of words across the file

Expected Output

Working Python functions for each prompting style, with a brief reflection comparing their accuracy, clarity, and efficiency.



# Task #5 – Few-Shot Prompting for Text Processing and Word Frequency

Objective

Use few-shot prompting (with at least 3 examples) to generate a Python function that processes text and analyzes word frequency.

Requirements The function must:

* Accept a paragraph as input
* Convert all text to lowercase
* Remove punctuation
* Return the most frequently used word

