**Assessment for Gen AI/Prompt Engineering Role**

**Problem Statement 1: Natural Language Processing (NLP)**

**Problem:** Implement a function to preprocess and tokenize text data. **Requirements:**

* Implement in Python using libraries like NLTK or spaCy.
* Handle edge cases such as punctuation, stop words, and different cases. **Evaluation Criteria:**
* Correctness of the preprocessing steps.
* Efficiency and readability of the code.
* Clean and structured code with appropriate comments.

**Problem Statement 2: Text Generation**

**Problem:** Create a basic text generation model using a pre-trained transformer (e.g., GPT-3). **Requirements:**

* Use the Hugging Face Transformers library.
* Generate coherent text based on a given prompt. **Evaluation Criteria:**
* Ability to load and use pre-trained models.
* Quality and coherence of the generated text.
* Understanding and application of the transformer model.

**Problem Statement 3: Prompt Engineering**

**Problem:** Design and evaluate prompts to improve the performance of a given AI model on a specific task (e.g., summarization, question answering). **Requirements:**

* Experiment with different prompt designs.
* Evaluate the effectiveness of each prompt using appropriate metrics. **Evaluation Criteria:**
* Creativity and effectiveness of prompt designs.
* Use of proper evaluation metrics.
* Clear explanation and documentation of the process and results.

**Problem Statement 4: Data Analysis**

**Problem:** Analyze a dataset and generate insights using a combination of descriptive statistics and visualizations. **Requirements:**

* Use Python libraries like Pandas, NumPy, and Matplotlib/Seaborn.
* Provide a Jupyter notebook with the analysis and visualizations. **Evaluation Criteria:**
* Accuracy and depth of the data analysis.
* Quality and clarity of the visualizations.
* Clean and well-documented code.

**Problem Statement 5: Live Coding Session - API Integration**

**Problem:** Develop a Python script to integrate with an external API and fetch data based on user input. **Requirements:**

* Use the Requests library to make API calls.
* Handle API responses and errors gracefully.
* Parse and display the fetched data in a user-friendly format. **Evaluation Criteria:**
* Correct implementation of API integration.
* Handling of different types of API responses and errors.
* Clean and well-structured code with appropriate comments.