Text Preprocessing using SpaCy and Python Libraries

Objective:
To apply basic text preprocessing techniques on texts in both Arabic and English using Python libraries (other than NLTK). This includes techniques such as tokenization, stopword removal, POS tagging, noise removal, and text normalization.

Instructions:

1. **Choose the Texts:**
   Select short texts (sentences or paragraphs) in both Arabic and English. You can choose texts from any domain (e.g., news articles, social media posts, books, etc.).
2. **Apply the following preprocessing steps using any Python library except NLTK:**
   - **Tokenization:**
     Tokenize the text into words or sentences. You should explain why tokenization is an important step in text preprocessing.
   - **Stopword Removal:**
     Remove non-essential words such as common stopwords (e.g., "the", "and", "is" in English, or "في" ,"و" ,"من" in Arabic). Explain the significance of removing stopwords in text preprocessing.
   - **POS Tagging (Part-of-Speech Tagging):**
     Perform POS tagging to identify the grammatical categories of words (e.g., noun, verb, adjective) in both languages. Explain why POS tagging is important in text analysis.
   - **Noise Removal:**
     Identify and remove any elements that could be considered "noise" in the text (e.g., symbols, unnecessary punctuation, numbers). Show examples before and after removing the noise.
   - **Normalization:**
     Normalize the text by applying transformations like converting the text to lowercase, removing extra spaces, or correcting common spelling errors. Explain how normalization can improve the quality of the text.
3. **Analysis and Interpretation:**
   After applying the preprocessing steps, provide an analysis of how the text has changed. Discuss the benefits of each preprocessing step and explain how it improves the text for further analysis or modeling.

4. **Report:**
   Submit a report that includes:
   - Original text in Arabic and English.
   - The text after applying the preprocessing steps.
   - The Python code used for each step.
   - Results of the preprocessing, including any observations or challenges faced during the process.

Evaluation Criteria:

- **Accuracy and Completeness:** Ensure all steps are applied to both the Arabic and English texts.
- **Code Implementation:** The quality and clarity of the code used to implement each preprocessing step.
- **Analysis and Interpretation:** How well you explain the impact of the preprocessing steps and the reasoning behind them.
- **Clarity of Report:** The organization, language, and quality of the report, written in colloquial Arabic.

Deadline:

The assignment must be submitted **before the next session**.