Libraries Used in the Notebook

1. spacy

SpaCy is a library for Natural Language Processing (NLP). It allows you to analyze and understand text with Python.

In this notebook, **spaCy** is used to:

- Split a sentence into individual words or punctuation marks (called tokens).
- Figure out what grammatical role each word plays (called part-of-speech or POS tagging), such as noun, verb, or adjective.

The line used to load the English model:

nlp = spacy.load('en_core_web_sm')

loads a small English model that comes with vocabulary, grammar rules, and statistical patterns.

Note: If you haven't downloaded this model before, you'll need to run this in your terminal:

python -m spacy download en_core_web_sm

2. pandas

A popular Python library for working with structured data.

In this notebook, it's used to organize and display the POS tagging results in a readable format called a DataFrame, which looks like a table with rows and columns.

Example of creating an empty DataFrame:

pos_df = pd.DataFrame(columns=['token', 'pos_tag'])

What the Code Does

Step 1: Load the NLP Model

nlp = spacy.load('en_core_web_sm')

This line prepares spaCy to process English text.

The model understands grammar and can label each word with its role in the sentence.

Step 2: Add a Text Sample

emma_ja = "emma woodhouse handsome clever and rich..."

This is a paragraph from Jane Austen's Emma.

The text is already cleaned: it's all lowercase and doesn't contain punctuation.

This makes it simpler to analyze.

Step 3: Process the Text

spacy_doc = nlp(emma_ja)

The text is passed through the NLP model.

The result is a Doc object, which contains all the individual words and information about them (like POS tags).

Step 4: Set Up a Data Table

pos_df = pd.DataFrame(columns=['token', 'pos_tag'])

This creates a table structure where each word and its part-of-speech tag will be added.