1. What are Corpora?

A: Corpora refers to a collection or **body of text documents used for linguistic analysis** or language modeling. It can be a set of books, articles, transcripts, or any other textual data. Corpora provide a source of text for studying language patterns, extracting information, and developing language models.

Examples of corpora include a collection of news articles, a database of social media posts, or a compilation of scientific papers.

2. What are Tokens?

A: Tokens are the individual units or elements obtained by breaking down a text into smaller pieces. In natural language processing, tokens are typically words, but they can also be punctuation marks, numbers, or other linguistic units. Tokenization is the process of splitting a text into these individual units.

For example, the sentence "I love cats" would be tokenized into the tokens: ["I", "love", "cats"].

3. What are Unigrams, Bigrams, Trigrams?

A: Unigrams, bigrams, and trigrams refer to different types of n-grams, which are contiguous sequences of n tokens or words.

- **Unigrams:** Unigrams refer to individual words in a text. Each word is considered as a separate unit. For example, the sentence "I love cats" would have the unigrams: ["I", "love", "cats"].

- **Bigrams:** Bigrams consist of two consecutive words occurring together in a text. They capture pairs of words and their co-occurrence patterns. For example, the sentence "I love cats" would have the bigrams: ["I love", "love cats"].

- **Trigrams:** Trigrams are sequences of three consecutive words occurring together. They provide more context compared to unigrams or bigrams. For example, the sentence "I love cats" would have the trigrams: ["I love cats"].

4. How to generate n-grams from text?

A: To generate n-grams from text, you can follow these steps:
1. Tokenize the text into individual words or tokens.
2. Slide a window of size n over the tokenized text, where n is the desired number of words in each n-gram.
3. Extract the n-gram from each window position to create a list of n-grams.

For example, let's generate bigrams from the sentence "I love cats":
1. Tokenize: ["I", "love", "cats"].
2. Slide a window of size 2:
   - First window: ["I", "love"]
   - Second window: ["love", "cats"]
3. Extract the bigrams: ["I love", "love cats"].

5. Explain Lemmatization
A: Lemmatization is the process of **reducing words to their base or root form**, known as the lemma. It aims to group together different inflected forms of a word. For example, the lemmatization of the words "running," "runs," and "ran" would be the base form "run." Lemmatization **takes into account the context and part of speech of a word** to determine its lemma. It is useful in tasks like text analysis, information retrieval, and language understanding.

6. Explain Stemming
A: Stemming is the process of reducing words to their stem or root form by removing suffixes or prefixes. It aims to normalize words by truncating them. For example, stemming would convert the words "running," "runs," and "ran" to the common stem "run." Stemming is a simpler and faster process compared to lemmatization, **but it may not always produce actual dictionary words.** Stemming is commonly used in information retrieval and indexing tasks

.

7. Explain Part-of-speech (POS) tagging
A: Part-of-speech (POS) tagging is the process of **assigning grammatical tags** to each word in a sentence. **These tags indicate the part of speech** or syntactic category to which a word belongs, such as **noun, verb, adjective, adverb, etc**. POS tagging **helps in understanding the grammatical structure and meaning of a sentence**.

For example, in the sentence "She eats an apple," POS tagging would assign the tags: ["pronoun", "verb", "determiner", "noun"].

8. Explain Chunking or shallow parsing
A: Chunking, also known as shallow parsing, is **the process of grouping and labeling words or tokens into syntactic chunks or phrases**. It involves identifying contiguous sequences of words that belong to the same grammatical structure, such as noun phrases, verb phrases, or prepositional phrases. Chunking helps in extracting meaningful information from a sentence without fully parsing its syntactic structure.

For example, in the sentence "The black cat is sleeping," chunking would identify the noun phrase "The black cat."

9. Explain Noun Phrase (NP) chunking
A: Noun Phrase (NP) chunking is a specific type of chunking that focuses on identifying noun phrases in a sentence. Noun phrases are phrases that include a noun and any words that modify or describe that noun. NP chunking helps in extracting information about people, places, objects, or concepts mentioned in a text.

For example, in the sentence "The big red apple," NP chunking would identify the noun phrase "The big red apple."

10. Named Entity Recognition

A: Named Entity Recognition (NER) is a process in natural language processing that aims to identify and classify named entities in text. Named entities are specific entities like persons, organizations, locations, dates, or other proper nouns. NER helps in extracting relevant information from text and understanding the context. For example, in the sentence "Apple Inc. is headquartered in Cupertino," NER would recognize "Apple Inc." as an organization and "Cupertino" as a location.