**Chunking Stratergy**

**Chunking** is breaking large text into smaller, manageable pieces so the LLM can process them without exceeding its token limit

**Reasons for chunk\_size=100 and chunk\_overlap=10:**

Chunk Size (100 tokens):

Optimal size for readability: 100 tokens is small enough to avoid overloading the model's token limit, yet large enough to capture meaningful information.

Prevents truncation: A chunk size of 100 ensures that important parts of the text (like sentences or paragraphs) are not split inappropriately. This helps in maintaining the integrity of the information.

Chunk Overlap (10 tokens):

Context retention: The 10-token overlap ensures that the boundaries between chunks don't cause the loss of critical context. It helps the model to maintain continuity between adjacent chunks, which is especially important for documents where information in one chunk may reference or build upon the next.

Smoothing transitions: The overlap makes the transition between chunks smoother and more coherent, preventing the model from losing context when moving from one chunk to another.

**Reason for Choosing Recursive character splitter**

A **recursive character splitter** is used to handle complex and nested text structures, ensuring that information is split in a way that preserves context and meaning. It helps maintain relationships between parts of the text, avoids losing important details, and allows for better tokenization, especially in complex documents. This approach is flexible and adapts to different text formats, ensuring accurate processing of various data types