**Understand the RAG Architecture**

1. **Retriever**:
   * A retriever fetches relevant documents or pieces of information from a knowledge base (e.g., a vector database or text corpus).
   * Common retrievers include BM25 (traditional keyword-based retrieval) and dense retrievers like FAISS or Pinecone.
2. **Reader (Generator)**:
   * A generative model (e.g., OpenAI's GPT, T5, or BART) processes the retrieved documents and generates a final answer or response.
   * The generative model integrates the retrieved knowledge with the query.
3. **Pipeline**:
   * The pipeline consists of two main stages: **retrieval** and **generation**.

**Set Up Your Environment**

1. Prepare the knowledge base:
   * Collect your corpus or knowledge base. This can be structured as text documents, JSON files, or a database.
   * Example: A collection of documents related to a specific domain like medicine or education.

**Implement the Retriever**

1. **Preprocess the Text**:
   * Tokenize and clean the text data.
   * Split long documents into smaller chunks.

**Create a Vector Index**:

* Use sentence-transformers to create embeddings for your documents.
* Store these embeddings in a vector database like FAISS.

**Implement the Reader (Generator)**

1. **Choose a Pre-trained Generative Model**:
   * Models like T5, BART, or OpenAI's GPT-3 are suitable for this step.
2. **Fine-tune the Model (Optional)**:
   * If you have a domain-specific dataset, fine-tune the model to generate accurate and context-relevant responses.
3. **Integrate the Retriever with the Generator**:
   * Concatenate the query and retrieved documents, then pass them to the generative model.