1. **Load Document**: Reads the document text file.
2. **Split Document**: Breaks it into manageable chunks for processing and efficient retrieval.
3. **Create Vector Store**: Uses embeddings (like OpenAI embeddings) to create a searchable vector database for the document's content.
4. **Build QA Pipeline**: Combines a retriever with a pre-trained model (like text-davinci-003) to enable answering questions.
5. **Interactive QA**: Takes user input and processes it using the pipeline.

**Dependencies:**

* **Hugging Face Transformers**: For LLM-based pipelines.
* **LangChain**: For document splitting, vector storage, and retrieval-based QA.
* **FAISS**: For efficient similarity search in the vector space.