# Hands-on: RAG Pipeline

🔧 **LangChain RAG Implementation Flow**

### 1. Load Documents
```
loader = PDFLoader(
    "medical.pdf"
)
docs = loader .load()
```

### 2. Split Text
```
splitter =
TextSplitter(
 chunk_size = 1000
)
```

### 3. Embeddings
```
embeddings =
OpenAIEmbeddings()
vectors =
embeddings .embed()
```

### 4. Vector Store
```
vectorstore  =
Qdrant(
 embeddings
)
```

### 5. Query & Response Pipeline

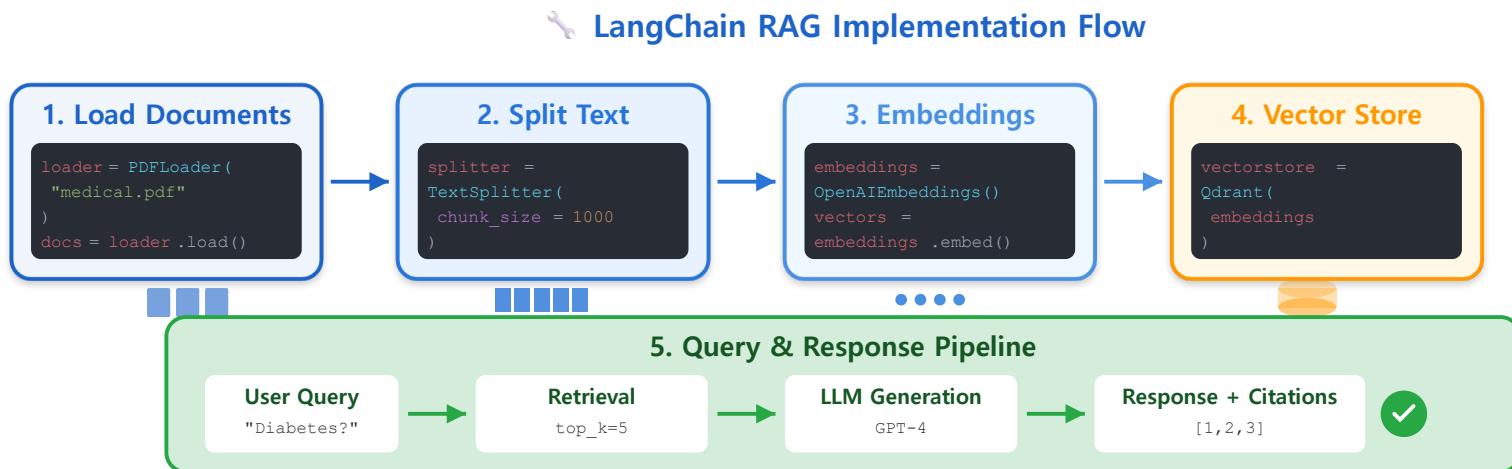| User Query | Retrieval | LLM Generation | Response + Citations |
|------------|-----------|----------------|----------------------|
| "Diabetes?" | top_k=5 | GPT-4 | [1,2,3] |

✅

## 💻 Complete Implementation Code

```python
from langchain.vectorstores import Qdrant
from langchain.embeddings import OpenAIEmbeddings
from langchain.chains import RetrievalQA

# Setup Vector Store
vectorstore = Qdrant(embeddings=OpenAIEmbeddings(), collection="medical_kb")

# Create RAG Chain
qa_chain = RetrievalQA.from_chain_type(
    llm=OpenAI(temperature=0),
    retriever=vectorstore.as_retriever(search_kwargs={"k": 5}),
    return_source_documents=True
)

# Query with Citations
result = qa_chain({"query": "Treatment for Type 2 Diabetes?"})
print(result['answer'], result['source_documents'])
```