**RAG**

* 오늘의 강의: [풀스택 GPT: #6.0부터 ~ #6.10까지](https://nomadcoders.co/fullstack-gpt/lectures/4571)
* 오늘의 과제: 위의 강의를 시청하신 후, 아래 코드 챌린지를 제출하면 됩니다.
* 제출기간: 3일 챌린지! 월요일 오전 6시까지

**챌린지**

**(EN)**

* Implement a complete RAG pipeline with a Stuff Documents chain.
* You must implement the chain manually.
* Give a ConversationBufferMemory to the chain.
* Use this document to perform RAG: <https://gist.github.com/serranoarevalo/5acf755c2b8d83f1707ef266b82ea223>
* Ask the following questions to the chain:
  + Is Aaronson guilty?
  + What message did he write in the table?
  + Who is Julia?

**(KR)**

* Stuff Documents 체인을 사용하여 완전한 RAG 파이프라인을 구현하세요.
* 체인을 수동으로 구현해야 합니다.
* 체인에 ConversationBufferMemory를 부여합니다.
* 이 문서를 사용하여 RAG를 수행하세요: <https://gist.github.com/serranoarevalo/5acf755c2b8d83f1707ef266b82ea223>
* 체인에 다음 질문을 합니다:
  + Aaronson 은 유죄인가요?
  + 그가 테이블에 어떤 메시지를 썼나요?
  + Julia 는 누구인가요?

**제출방법**

* **노트북을 먼저 실행**한 다음 커밋을 생성하고 Github에 푸시하세요.
* 노트북을 실행한 후 커밋을 생성하고 Github에 푸시합니다.
* 제출 링크: 깃허브 리포지토리가 아닌 **github 커밋** 이어야 합니다.
* 제출기간: 3일 챌린지! 월요일 오전 6시까지

**TA's 힌트**

* 다음과 같은 절차대로 구현하면 챌린지를 해결할 수 있습니다.
* (1) 문서 로드하기 : TextLoader 등 을 사용해서 파일에서 텍스트를 읽어옵니다. ([Document Loaders 관련 문서](https://python.langchain.com/v0.1/docs/modules/data_connection/document_loaders/)
* (2) 문서 쪼개기 : CharacterTextSplitter 등 을 사용해서 문서를 작은 문서 조각들로 나눕니다. [Character Split 관련 문서](https://python.langchain.com/v0.1/docs/modules/data_connection/document_transformers/character_text_splitter/)
* (3) 임베딩 생성 및 캐시 : OpenAIEmbeddings, CacheBackedEmbeddings 등 을 사용해 문서 조각들을 임베딩하고 임베딩을 저장합니다. [Caching 관련 문서](https://python.langchain.com/v0.1/docs/modules/data_connection/text_embedding/caching_embeddings/)
* (4) 벡터 스토어 생성 : FAISS 등 을 사용해서 임베딩된 문서들을 저장하고 검색할 수 있는 데이터베이스를 만듭니다. [FAISS 관련 문서](https://python.langchain.com/v0.1/docs/integrations/vectorstores/faiss/)
* (5) 대화 메모리와 질문 처리 : ConversationBufferMemory를 사용해 대화 기록을 관리합니다.
* (6) 체인 연결 : 앞에서 구현한 컴포넌트들을 적절하게 체인으로 연결합니다.