

/* elice */

Enterprise

커스텀 + 출처

실습: 커스텀 프롬프트 적용 → 출처 표시 구현 → 연습 과제

14:30 ~ 15:20

실습 범위 정리

여태까지 실습한 내용

- 사양서+성적서 → 벡터DB 구축
- RetrievalQA 체인 동작 확인
- 질의 5개 테스트 + PDF 대조
- 체크포인트 4개 통과

실습할 내용

- 커스텀 프롬프트 적용 (Section 4)
- 출처 표시 함수 구현 (Section 5)
- 다양한 질문으로 검증 (Section 6)

Section 4~6을 순서대로 진행합니다. 막히면 ① 에러 읽기 → ② 조원 상의 → ③ 강사 호출!

커스텀 프롬프트: 왜 필요한가?

기본 체인으로 질문

Q: "절연저항 시험 결과는?"

A: 절연저항 시험 결과는 적합입니다.

수치($385M\Omega$), 기준값, 단위 누락

규칙을 추가한 커스텀 체인

Q: "절연저항 시험 결과는?"

A: RF900의 절연저항 시험 결과는 $385M\Omega$ 으로,
기준값($\geq 1.0M\Omega$) 대비 적합 판정입니다.

수치 + 기준값 + 단위 포함

오전에 진행한 브레인스토밍 기억나시죠?

"우리 업무에 맞는 규칙 하나 더 만들어보자"

지금 그 규칙을 코드에 직접 넣어봅니다!

PromptTemplate 코드 작성

PromptTemplate은 2주차에 배웠습니다. RAG 맥락의 {context}, {question}만 새로 봅니다.

```
# Section 4: 커스텀 프롬프트 ~ 답변 규칙 추가
from langchain.prompts import PromptTemplate

custom_prompt = PromptTemplate(
    template="""다음 문서를 참고하여 질문에 답하세요.

문서 내용: {context}
질문: {question}

답변 규칙:
1. 문서에 있는 정보만 사용하세요
2. 문서에 없으면 "해당 정보를 찾을 수 없습니다"라고 답하세요
3. 수치를 인용할 때는 단위를 반드시 포함하세요
답변:""",
    input_variables=["context", "question"]
)
```

핵심: **{context}** = 검색된 문서 조각, **{question}** = 사용자 질문 → LangChain이 자동으로 채워줍니다

커스텀 체인 생성 + 규칙 추가

```
# 커스텀 프롬프트를 적용한 RAG 체인

custom_qa = RetrievalQA.from_chain_type(
    llm=llm,
    chain_type="stuff",
    retriever=vectorstore.as_retriever(search_kwargs={"k": 3}),
    return_source_documents=True,
    chain_type_kwargs={"prompt": custom_prompt}
)
```

chain_type_kwargs={"prompt": custom_prompt} ← 이 한 줄!

여러분의 규칙 추가!

기본 3가지 규칙에 더해서:

4. (오전 브레인스토밍 규칙)

예시:

- 표 형식으로 정리해 답변
- 안전 수치는 기준값도 포함
- 출처 문서명을 답변에 포함

비교 테스트: 같은 질문을 qa(기본)과 custom_qa(커스텀)에 각각 던져보세요. 차이가 보이면 성공!

같은 질문, 다른 답변 - 직접 확인

기본 체인 (qa)

Q: "RF9000의 소비전력은?"

A: RF9000의 소비전력은 36W입니다.

Q: "RF9000 가격은?"

A: RF9000의 가격은 약 200만원입니다.

근거 없이 지어냄!

커스텀 체인 (custom_qa)

Q: "RF9000의 소비전력은?"

A: RF9000의 정격 소비전력은 36W입니다.
(제품사양서 p.2 기준)

Q: "RF9000 가격은?"

A: 해당 정보를 찾을 수 없습니다.
제품사양서와 시험성적서에 가격
정보는 포함되어 있지 않습니다.

솔직하게 모른다고 답함!

규칙 3가지가 만드는 차이: 단위 포함 + 없으면 솔직하게 답변 = 할루시네이션 방지!

출처 표시 함수 작성

오전 3에서 본 `format_response_with_sources()`를 확장합니다

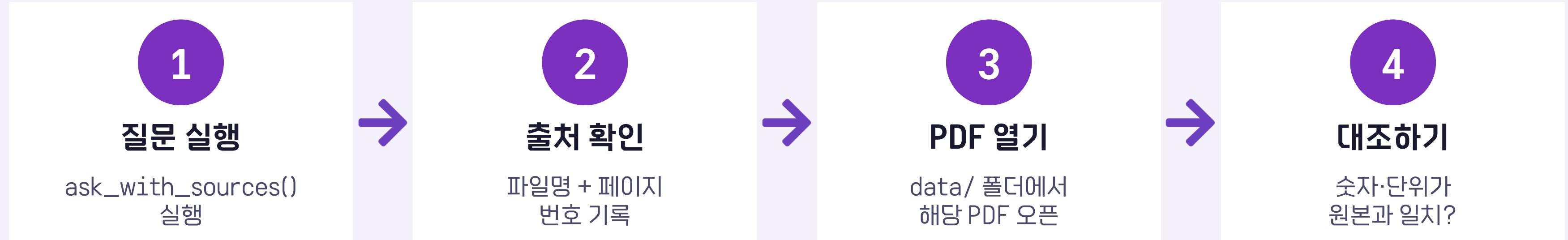
```
# Section 5: 출처 표시 함수 ~ 질문 + 답변 + 출처를 한 번에!
def ask_with_sources(query):
    """질문하고, 답변과 출처를 함께 출력"""
    result = custom_qa.invoke({"query": query})

    print("=" * 50)
    print(f"질문: {query}")
    print("=" * 50)
    print(f"\n💬 답변:\n{result['result']}")

    print(f"\n📋 참고 문서 ({len(result['source_documents'])}건):")
    for i, doc in enumerate(result['source_documents']):
        name = doc.metadata.get('source', '').split('/')[-1]
        pg = doc.metadata.get('page', 0) + 1 # 0→1 변환
        print(f" [{i+1}] {name} (p.{pg})")
    print("=" * 50)
```

사용법: `ask_with_sources("RF9000 냉매 종류는?")` ← 이 함수 하나로 질의응답 + 출처까지!

출처 정확성 검증: PDF와 대조하기



```
# 교차 확인 예시
ask_with_sources("에너지효율 등급은?")
# [1] 시험성적서_RF9000.pdf (p.2) ← 측정값: 28.5kWh/월
# [2] 제품사양서_RF9000.pdf (p.2) ← 사양표: 에너지효율 1등급
```

두 문서에서 교차 확인! 시험성적서에서 측정값, 사양서에서 스펙 ~ 벡터DB에 두 문서가 모두 있기 때문에 가능합니다.

연습 과제: 다양한 질문 던져보기

기본 과제: 질문 5개 + 출처 대조

사양서

"냉장실 용량은 몇 리터?"

사양서

"E3 에러코드는 무엇인가요?"

성적서

"절연저항 시험 합격 기준은?"

성적서

"소음 측정값은?"

교차

"이 제품은 안전한가요?"

심화 과제: "문서에 없는 질문" 3개

"RF9000 가격은 얼마인가요?"

"삼성 냉장고와 비교하면?"

"이 제품 AS 전화번호는?"

→ 커스텀 프롬프트 덕분에
"해당 정보를 찾을 수 없습니다" 라고 답해야 정상!

할루시네이션 대응은 내일 Day4!

각 조에서 최소 5개 이상 질문을 테스트하고, 출처가 올바르게 표시되는지 확인하세요.
잘못된 출처를 발견하면 메모해두세요! → 내일(Day4) 할루시네이션 대응에서 다시 다룹니다.

심화 미션: 출처 필터링 + 신뢰도

```
# 심화: 출처별 필터링 + 신뢰도 점수
def ask_with_confidence(query, filter_source=None):
    # 필터링: 특정 문서만 검색
    where_filter = None
    if filter_source:
        where_filter = {"source": filter_source}

    # similarity_search_with_score로 유사도 점수 포함
    docs_scores = vectorstore.similarity_search_with_score(
        query, k=3, filter=where_filter)

    print(f"\n검색된 문서와 유사도 점수:")
    for doc, score in docs_scores:
        src = doc.metadata.get('source', '').split('/')[-1]
        confidence = max(0, min(100, int((1 - score/2) * 100)))
        bar = '█' * (confidence // 10) + '▒' * (10 - confidence // 10)
        print(f" {bar} {confidence}% | {src}")
```

사용 예: `ask_with_confidence("절연저항", filter_source="시험성적서...pdf")` → 시험성적서에서만 검색!

실습 체크리스트 & 트러블슈팅

완료 체크리스트

- PromptTemplate 작성 완료
- custom_qa 체인 생성 완료
- 기본 vs 커스텀 비교 확인
- ask_with_sources() 함수 동작
- 출처(파일명+페이지) 정확성 검증
- 질의 5개 이상 테스트 완료

자주 묻는 에러

NameError: custom_prompt

→ PromptTemplate 셀을 먼저 실행

"source_documents" KeyError

→ return_source_documents=True 확인

답변에 단위가 없음

→ 프롬프트 규칙 3번 확인

시간 관리

Section 4 커스텀(20분) + Section 5 출처(20분) + Section 6 과제(10분) = 총 50분

Section 5(출처 함수)까지 마치면 오늘의 학습 목표 3개를 모두 달성! Section 6은 보너스.