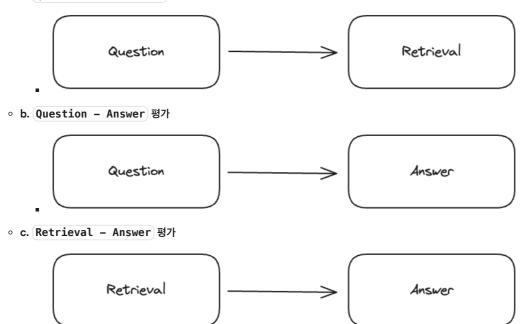
- 출처: LangChain 공식 문서 또는 해당 교재명
- 원본 URL: https://smith.langchain.com/hub/teddynote/summary-stuff-documents

4. LangSmith 데이터셋 생성

- 자체 RAG 평가용 데이터셋 구축하는 방법 알아보기
 - ∘ a. Question Retrieval 평가



- Retrieval 에 대한 Ground Truth 구축 어려움
 - a. Retrieval 에 대한 Ground Truth 가 존재한다면 모두 데이터셋으로 저장하여 활용

if langchain_tracing_v2 == "true" and os.getenv('LANGCHAIN_API_KEY') and langchain_project: print(f"☑ LangSmith 추적 활성화됨 (LANGCHAIN_TRACING_v2='{langchain_tracing_v2}')")

b. 그렇지 않다면 Question, Answer 만으로 데이터셋 구축 → 활용 가능

print(f"▼ LangSmith 프로젝트: '{langchain_project}'")
print(f"▼ LangSmith API Key: {langchain_api_key_status}")
print(" -> 이제 LangSmith 대시보드에서 이 프로젝트를 확인해 보세요.")

print("★ LangSmith 추적이 완전히 활성화되지 않았습니다. 다음을 확인하세요:")

• 환경 설정

else:

```
# API 키를 환경변수로 관리하기 위한 설정 파일
from dotenv import load_dotenv

# API 키 정보 로드
load_dotenv() # True

from langsmith import Client
from langsmith import traceable
import os

# LangSmith 환경 변수 확인

print("\n--- LangSmith 환경 변수 확인 ---")
langchain_tracing_v2 = os.getenv('LANGCHAIN_TRACING_v2')
langchain_project = os.getenv('LANGCHAIN_PROJECT')
langchain_api_key_status = "설정됨" if os.getenv('LANGCHAIN_API_KEY') else "설정되지 않음" # API 키 값은 직접 출력하지 않음
```

```
if langchain_tracing_v2 != "true":
    print(f" - LANGCHAIN_TRACING_V2가 'true'로 설정되어 있지 않습니다 (현재: '{langchain_tracing_v2}').")
if not os.getenv('LANGCHAIN_API_KEY'):
    print(" - LANGCHAIN_API_KEY가 설정되어 있지 않습니다.")
if not langchain_project:
    print(" - LANGCHAIN_PROJECT가 설정되어 있지 않습니다.")
```

셀 출력

```
--- LangSmith 환경 변수 확인 ---
☑ LangSmith 추적 활성화됨 (LANGCHAIN_TRACING_V2='true')
☑ LangSmith 프로젝트: 'LangChain-prantice'
☑ LangSmith API Key: 설정됨
-> 이제 LangSmith 대시보드에서 이 프로젝트를 확인해 보세요.
```

1) 데이터셋 생성

- inputs , outputs → 데이터셋 생성하기
- 데이터셋 구성: question, answer

(1) 직접 생성하기

```
import pandas as pd

# 질문과 답변 목록
inputs = [
   "삼성전자가 만든 생성형 AI의 이름은 무엇인가요?",
   "미국 바이든 대통령이 안전하고 신뢰할 수 있는 AI 개발과 사용을 보장하기 위한 행정명령을 발표한 날은 언제인가요?",
   "코히어의 데이터 출처 탐색기에 대해서 간략히 말해주세요.",
]

# 질문에 대한 답변 목록
outputs = [
   "삼성전자가 만든 생성형 AI의 이름은 삼성 가우스 입니다.",
   "2023년 10월 30일 미국 바이든 대통령이 행정명령을 발표했습니다.",
   "코히어의 데이터 출처 탐색기는 AI 모델 훈련에 사용되는 데이터셋의 출처와 라이선스 상태를 추적하고 투명성을 확보하기 위한 플랫폼입니다. 12개 기관과 협력하여 2,000
```

```
# 데이터프레임으로 변환
df = pd.DataFrame(qa_pairs)
```

qa_pairs = [{"question": q, "answer": a} for q, a in zip(inputs, outputs)]

데이터프레임 출력 df.head()

- df.head()
 - Independent of the second of the sec

✓ ② 이전에서 생성한 Synthetic Dataset 활용하기

- 업로드했던 HuggingFace Dataset 활용하기
- 사전에 VS Code 터미널에 설치할 것

```
pip install -qU datasets
```

```
import pandas as pd
from datasets import load_dataset, Dataset
import os

# huggingface Dataset에서 repo_id로 데이터셋 다운로드
dataset = load_dataset(
    "livemylife23/rag-synthetic-dataset-korean", # 데이터셋 이름
    token=os.environ["HUGGINGFACEHUB_API_TOKEN"], # private 데이터인 경우 필요
)
```

• 다운로드 완료 - (7.1s)

```
      ✓ 7.1s

      README.md: 100%
      500/500 [00:00<00:00, 44.6kB/s]</td>

      data/train-00000-of-00001.parquet: 100%
      15.7k/15.7k [00:01<00:00, 8.29kB/s]</td>

      Generating train split: 100%
      9/9 [00:00<00:00, 151.66 examples/s]</td>
```

```
# 데이터셋에서 split 기준으로 조회
huggingface_df = dataset["train"].to_pandas()
huggingface_df.head()
```

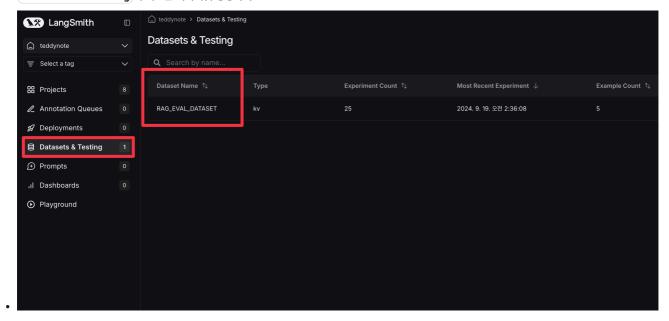
huggingface_df.head()

0

△ reference_contexts		△ synthesizer_name		user_input		△ reference		duser_input_translated		☆ reference_translated	
누락: 고유:	0 (0%) 5 (100%)		0 (0%) 3 (60%)		0 (0%) (100%)		(0%) 100%)		(0%) 00%)		0 (0% (100%
['미국 바이든 대통령이 2023 ['Δ1026 플롭스(FLOPS, Floa. ['행정명령은 ΔΑI의 안전과 보. 기타	20%	simple reasoning multi_context		미국 바이든 대통령은 안전하고 AI의 안전과 보안 기준 마련과 미국 행정명령은 AI가 사회 전 기타	20%	법률, 주택, 보건 분야에서 AI의	20%	AI의 안전과 보안 기준 마련과 미국 행정명령은 AI가 사회 전	20%	2023년 10월 30일 단일 데이터센터에서 1,000Gbi 법률, 주택, 보건 분야에서 AI의 기타	
['미국 바이든 대통령이 2023년	10월 30	simple		미국 바이든 대통령은 안전하고	신뢰할 수	2023년 10월 30일		미국 바이든 대통령은 안전하고 신화	리할 수	2023년 10월 30일	
['Δ1026 플롭스(FLOPS, Float	ting Point	reasoning		AI의 안전과 보안 기준 마련과 곤	련하여, 1	단일 데이터센터에서 1,000Gbit/s	s 이상으	AI의 안전과 보안 기준 마련과 관련	하여, 대	단일 데이터센터에서 1,000Gbit,	/s 0
['행정명령은 ΔAI의 안전과 보	안 기준 마	multi_context		미국 행정명령은 AI가 사회 전반	에 미치는	법률, 주택, 보건 분야에서 AI의 무	책임한	미국 행정명령은 AI가 사회 전반에	미치는	법률, 주택, 보건 분야에서 AI의 두	무책임
['G7이 첨단 AI 시스템을 개발하	하는 기업을	simple		G7이 AI 기업을 대상으로 마련한	국제 행	고급 인공지능 시스템에 대한 국제	행동강	G7이 AI 기업을 대상으로 마련한 국	제 행	고급 인공지능 시스템에 대한 국제	제 행
['행동강령은 AI 수명주기 전반	에 걸친 위	reasoning		AI 국제 행동강령에서 AI 수명주	기 전반에	AI 수명주기 전반에 걸쳐 위험을 평	경가 및 1	AI 국제 행동강령에서 AI 수명주기	전반에	AI 수명주기 전반에 걸쳐 위험을 :	평가

2) LangSmith 테스트를 위한 Dataset 생성

• Dataset & Testing 에 새로운 데이터셋 생성하기



• (csv) 파일에서 (csv) 파일에서 (csv) 파일에서 (csv) 사용 (csv) 작접 데이터셋 생성 가능

dataset = client.create_dataset(
 dataset_name=dataset_name,

• 참고: LangSmith UI 문서

```
from langsmith import Client

client = Client()
dataset_name = "RAG_EVAL_DATASET"

# 데이터셋 생성 함수

def create_dataset(client, dataset_name, description=None):
    for dataset in client.list_datasets():
        if dataset.name == dataset_name:
            return dataset
```

```
description=description,
     return dataset
 # 데이터셋 생성
 dataset = create_dataset(client, dataset_name)
                                                     # 2.9s
 # 생성된 데이터셋에 예제 추가
 client.create_examples(
     inputs=[{"question": q} for q in df["question"].tolist()],
     outputs=[{"answer": a} for a in df["answer"].tolist()],
     dataset_id=dataset.id,
• client.creat_examples() 추가하기 - (0.1s)
   {'example_ids': ['8d34ad4e-e3be-47ca-9a6a-9929608b60ca',
   '71841ffc-cfbe-4e05-b0b0-ef85e8a6ebd4',
   'e0d645a9-6d97-41da-8d45-cc5e8d6f0a37'],
   'count': 3}
• 데이터셋에 예제 나중에 추가 가능
 # 새로운 질문 목록
 new_questions = [
     "삼성전자가 만든 생성형 AI의 이름은 무엇인가요?",
     "구글이 테디노트에게 20억달러를 투자한 것이 사실입니까?",
 ]
 # 새로운 답변 목록
 new_answers = [
     "삼성전자가 만든 생성형 AI의 이름은 테디노트 입니다.",
     "사실이 아닙니다. 구글은 앤스로픽에 최대 20억 달러를 투자하기로 합의했으며, 이 중 5억 달러를 우선 투자하고 향후 15억 달러를 추가로 투자하기로 했습니다.",
 ]
 # UI에서 업데이트된 버전 확인
 client.create_examples(
     inputs=[{"question": q} for q in new_questions],
     outputs=[{"answer": a} for a in new_answers],
     dataset_id=dataset.id,
```

• UI에서 업데이트된 버전 확인하기 - (0.1s)

```
{'example_ids': ['7080404f-878d-4e22-9256-70a5d1b86ab3',
'b8599d26-9990-4153-8a92-0dc779087f2c'],
'count': 2}
```

- ✓ 데이터셋 준비 완료됨
- next: 05. LLM-as-Judge