RAG 평가

강사 장철원

Section 0

목차

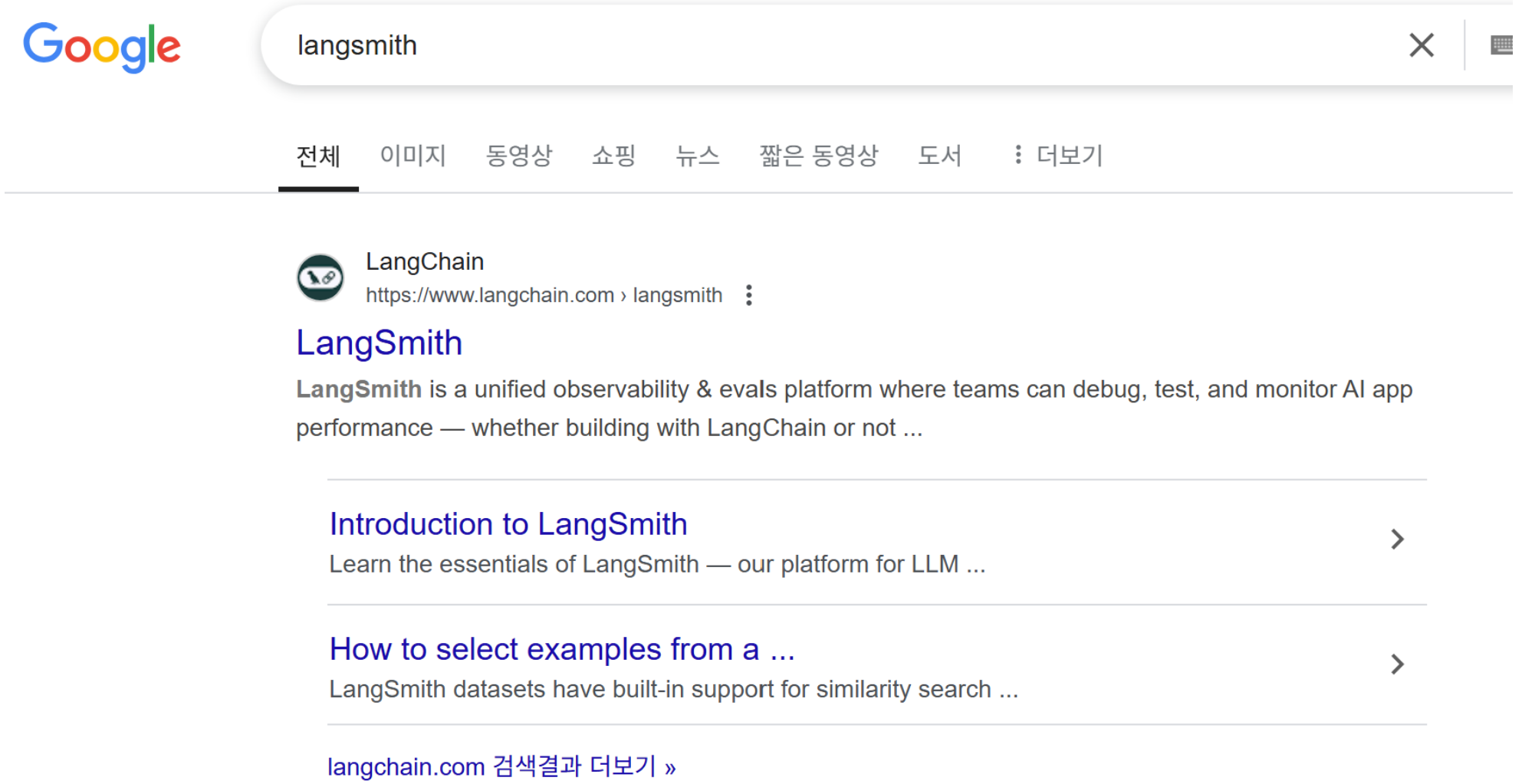
❑ LangSmith 소개

RAG 평가

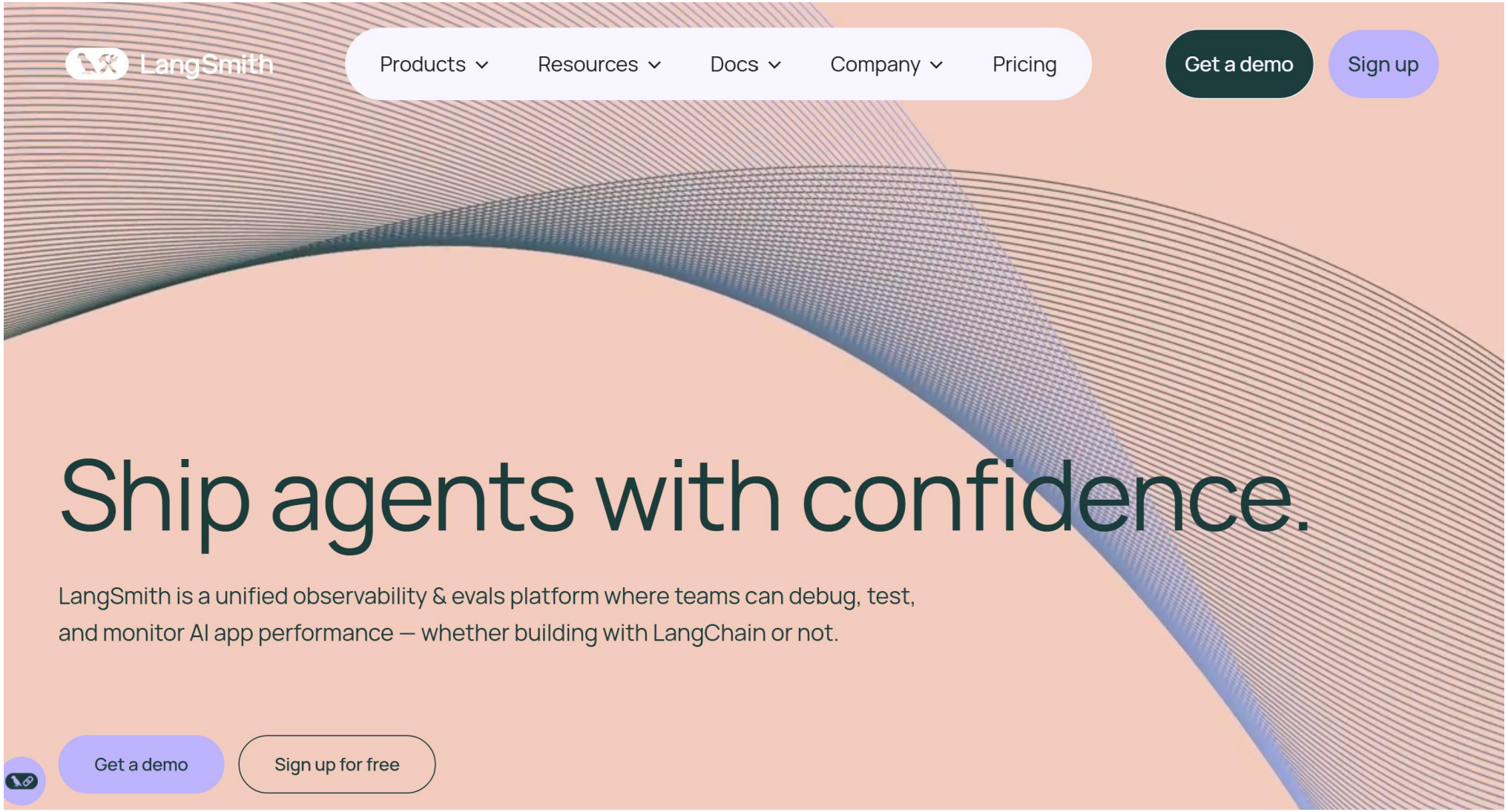
Section 1. LangSmith

Section 1-1. LangSmith의 개념

LangSmith의 개념



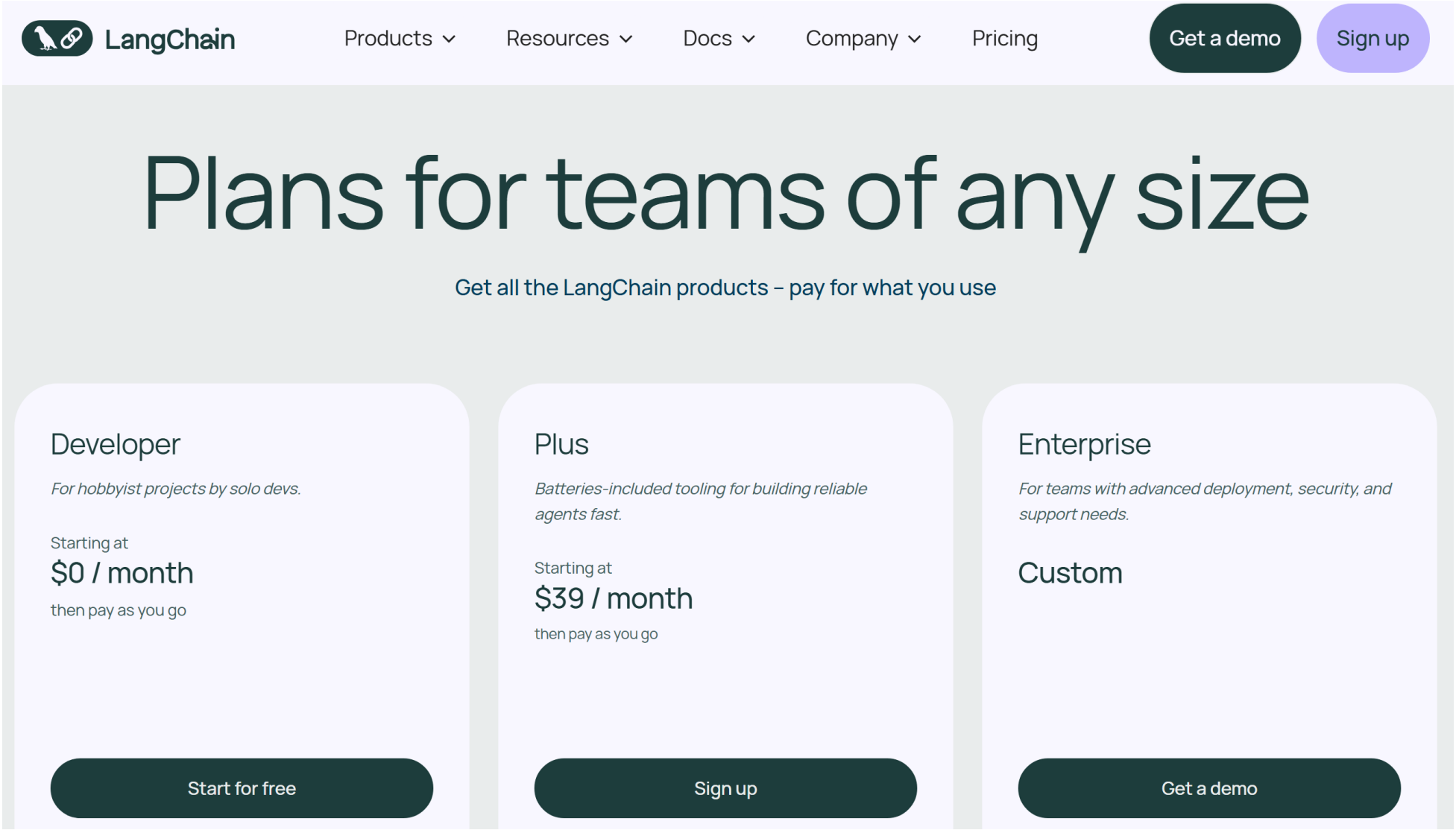
LangSmith의 개념



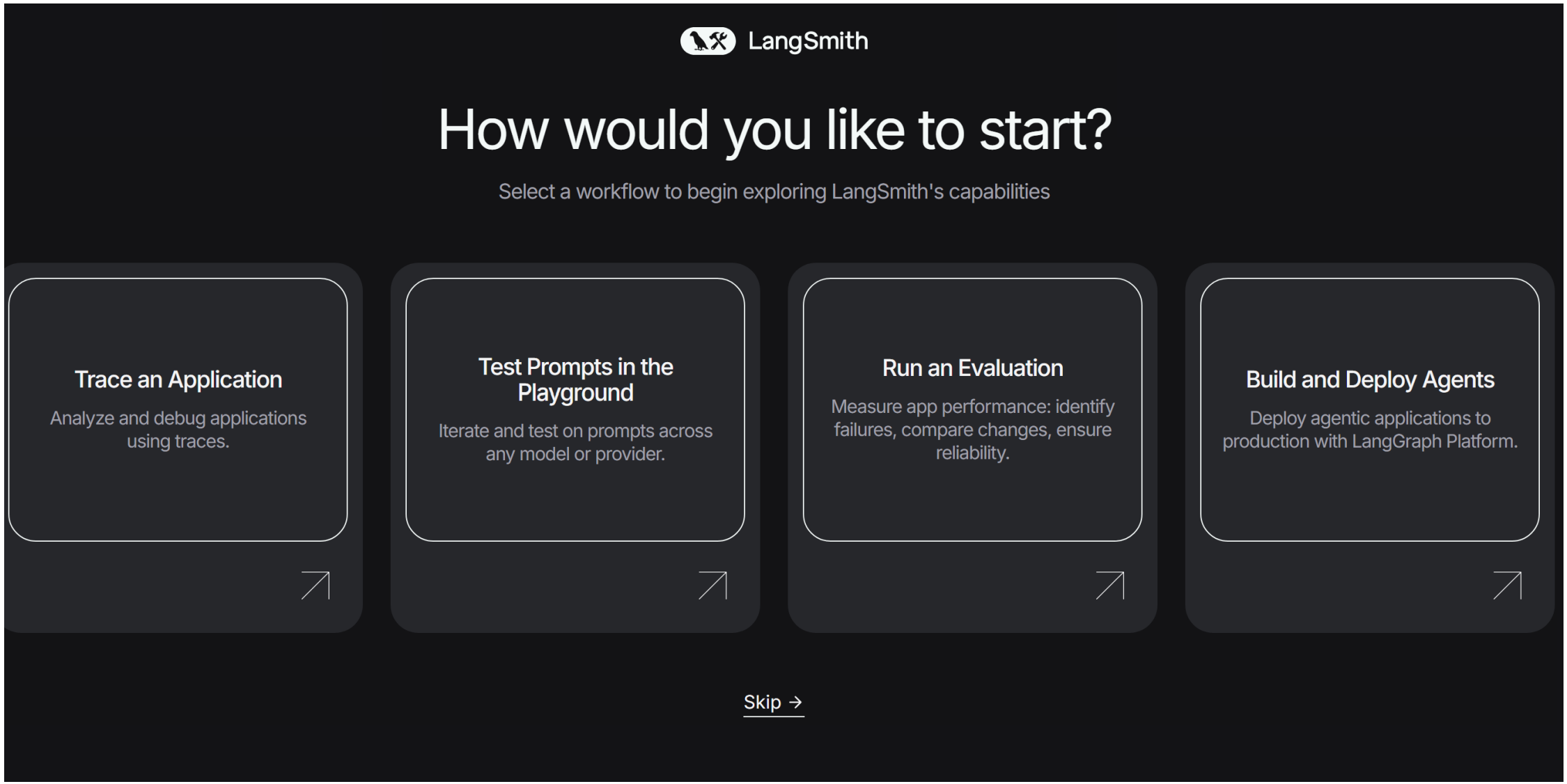
Section

LangSmith의 개념

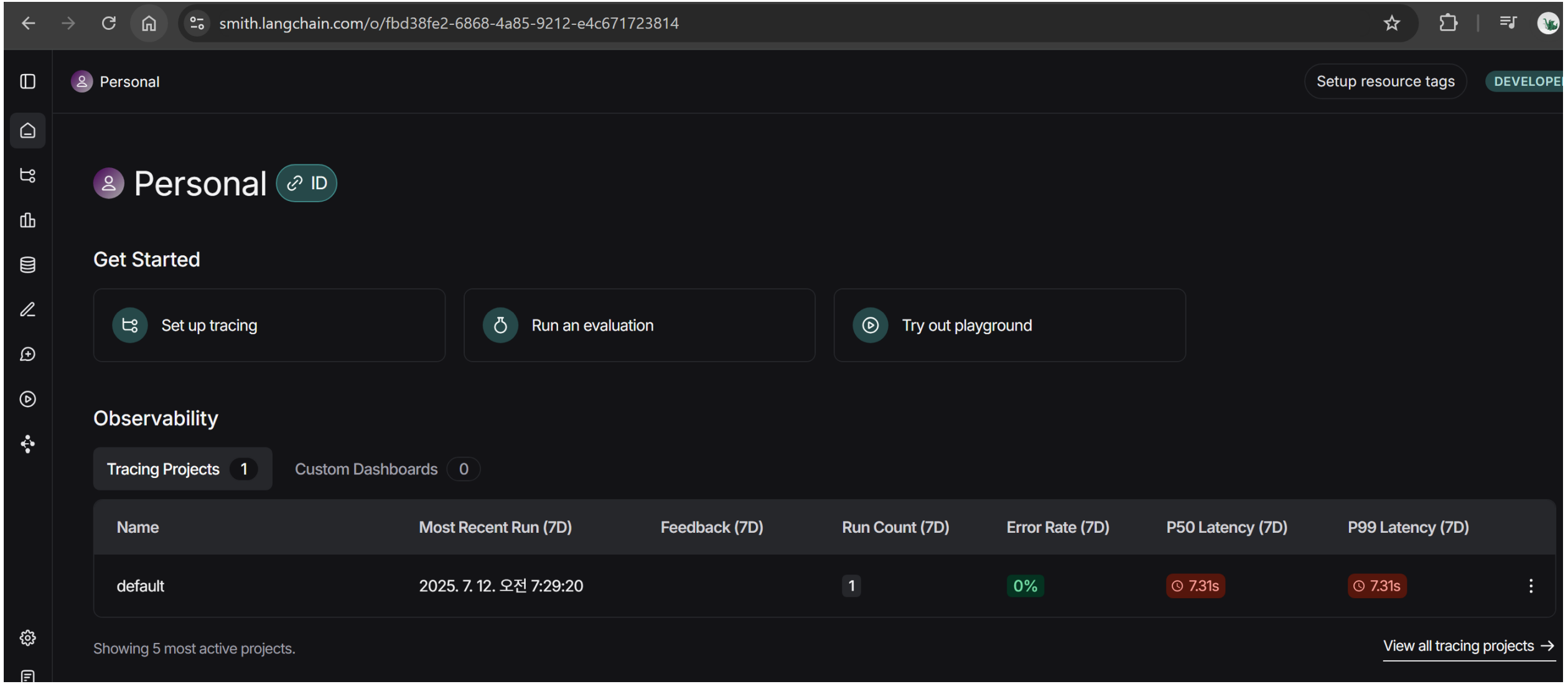
가격 정책



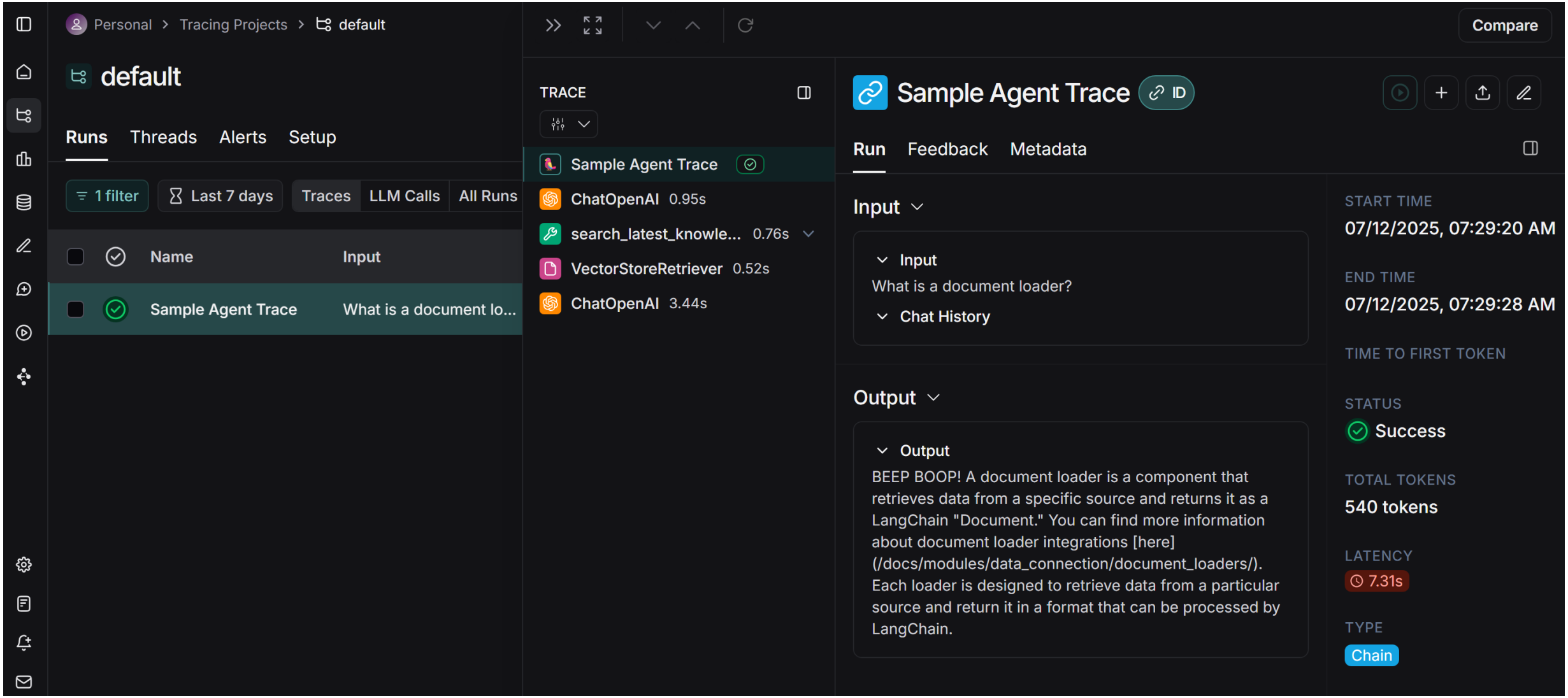
LangSmith의 개념



LangSmith의 개념



LangSmith의 개념

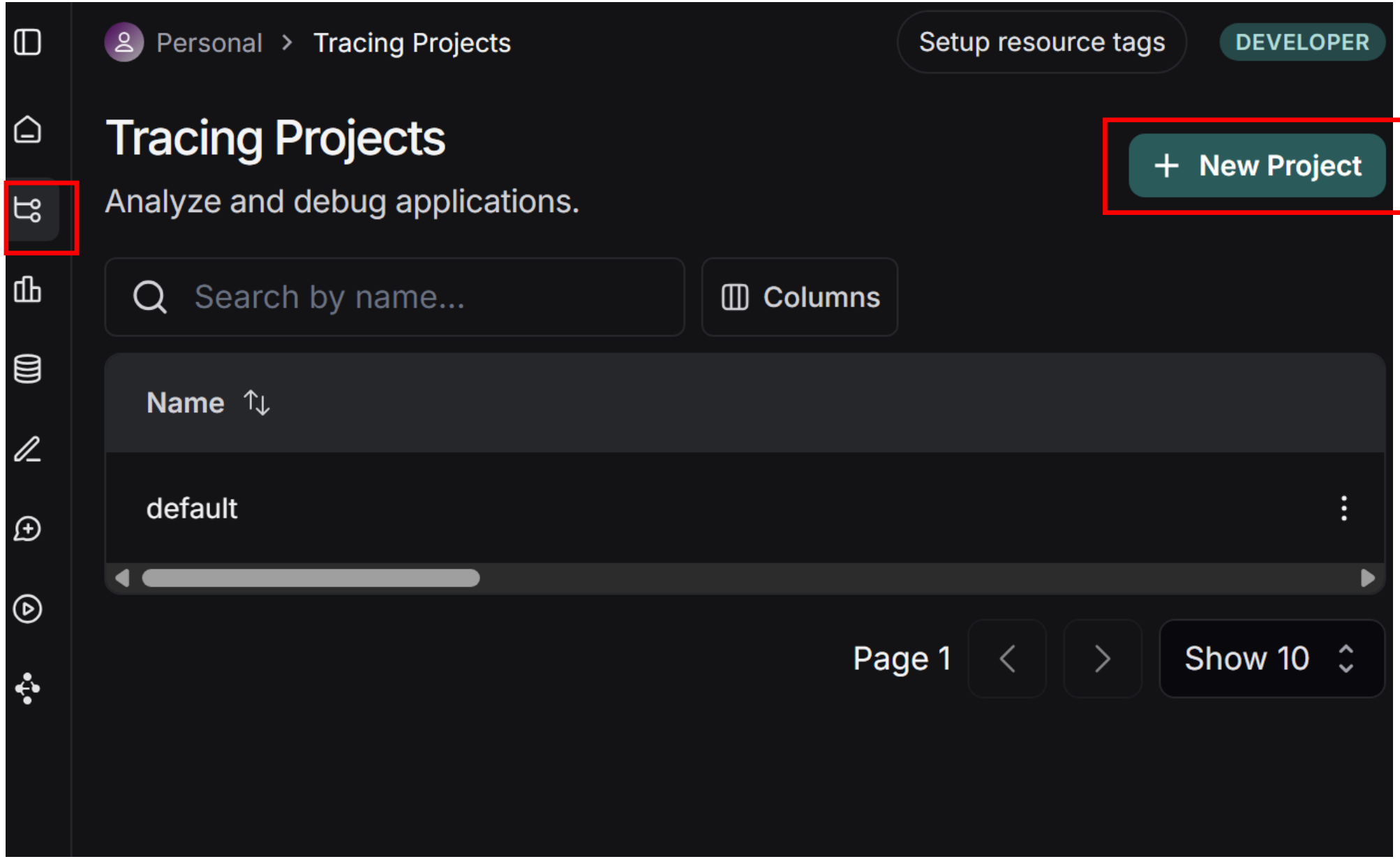


LangSmith

Section 1. LangSmith

Section 1-2. Tracing - LangChain

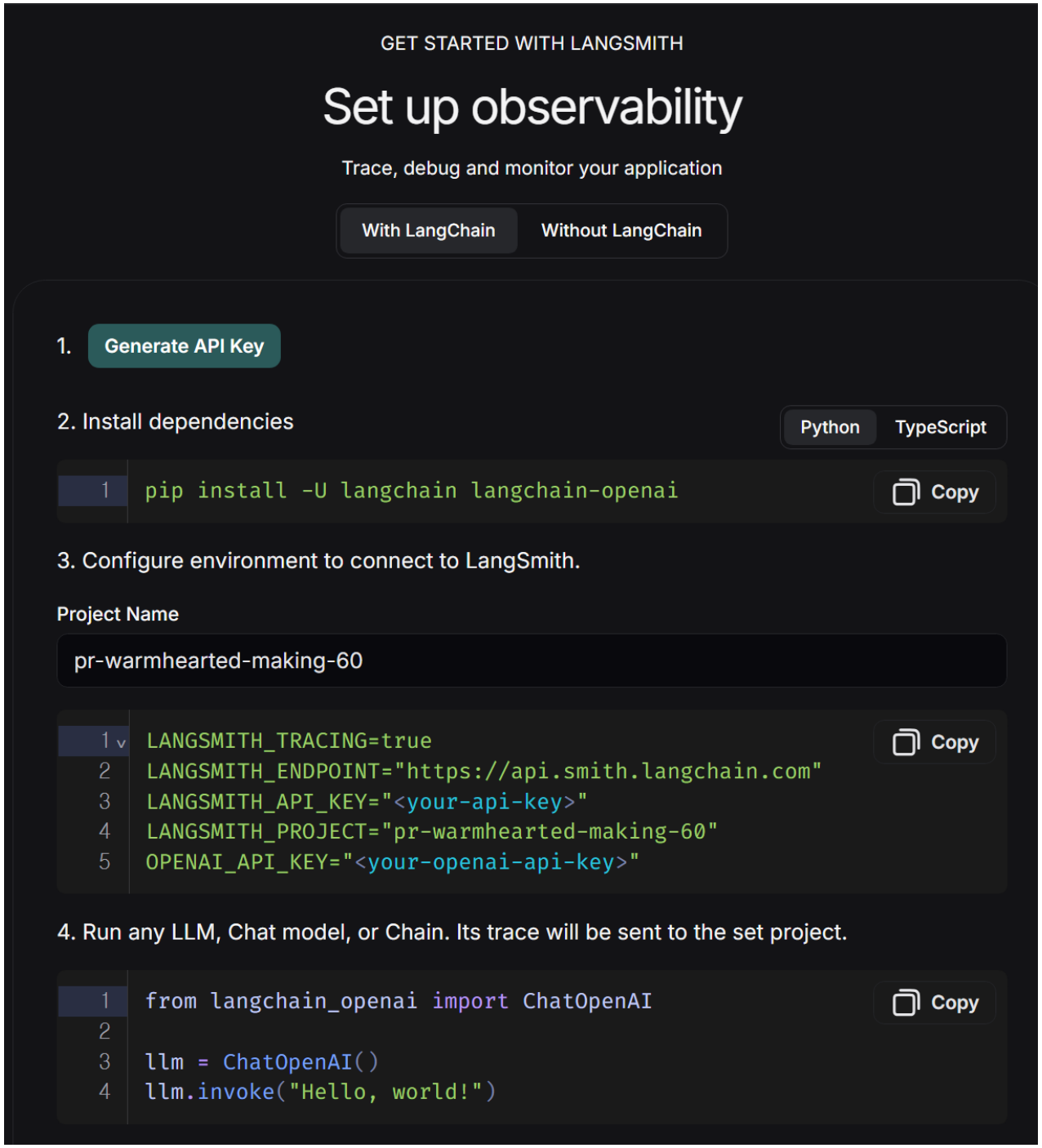
Tracing


Personal
Tracing Projects
Setup resource tags
DEVELOPER
Tracing Projects
Analyze and debug applications.
+ New Project
Search by name...
Columns
Name
default
Page 1
Show 10

Section

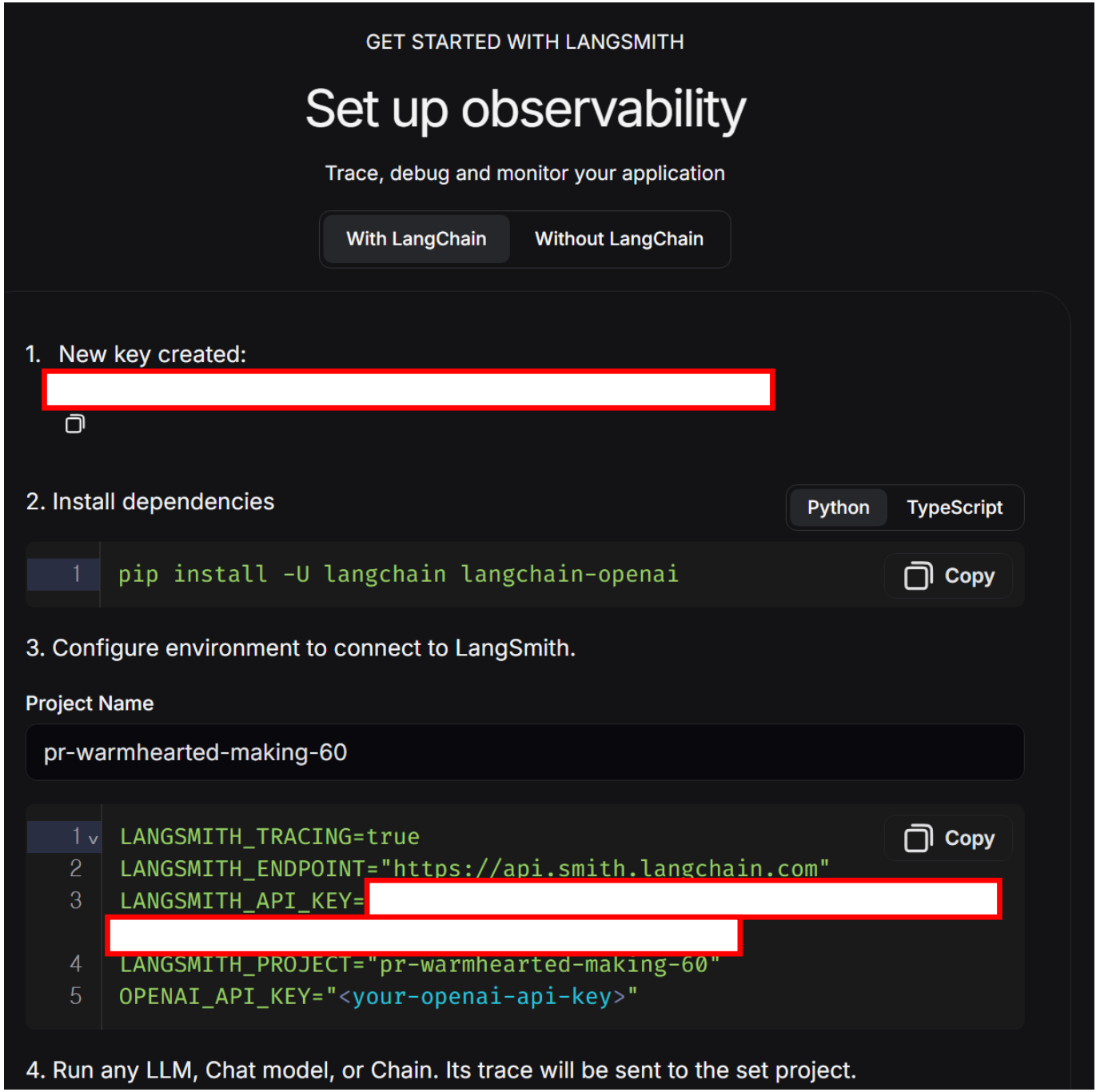
Tracing - LangChain

Tracing



Tracing

보관 주의



Section

Tracing - LangChain

코드 실행(1)

[10]:
```
import os

os.environ["LANGCHAIN_TRACING_V2"]="true"
os.environ["LANGCHAIN_ENDPOINT"]="https://api.smith.langchain.com"
os.environ["LANGCHAIN_API_KEY"]=
os.environ["LANGCHAIN_PROJECT"]="pr-shadowy-scow-41"
```

[11]:
```
from langchain_huggingface import HuggingFacePipeline
from transformers import AutoTokenizer, AutoModelForCausalLM, pipeline
```

[12]:
```
# 토크나이저 & 모델 불러오기
model_id = "skt/kogpt2-base-v2"
tokenizer = AutoTokenizer.from_pretrained(model_id)
model = AutoModelForCausalLM.from_pretrained(model_id)
```

[13]:
```
# 파이프라인 생성
text_gen = pipeline(
    "text-generation",
    model=model,
    tokenizer=tokenizer,
    device=-1,
    truncation=True,
    max_length=50,
    do_sample=True,
    temperature=0.7,
)
```

```
Device set to use cpu
```

Section

Tracing - LangChain

코드 실행(2)

[14]:
```python
# 프롬프트 예시
prompt = "산 속에 토끼 한 마리가 살고 있었습니다. 그러던 어느 날 "

# LangChain으로 실행
response = llm.invoke(prompt)
```

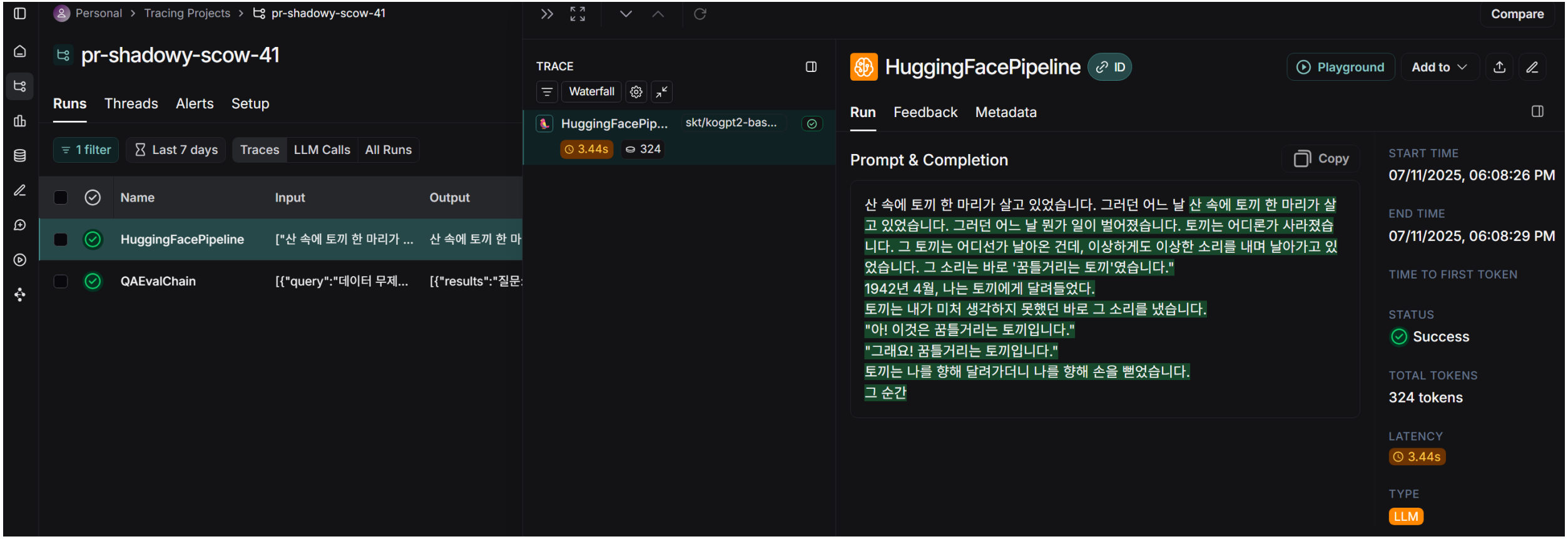
[15]:
```python
print("생성된 문장:", response)
```

```
생성된 문장: 산 속에 토끼 한 마리가 살고 있었습니다. 그러던 어느 날 뭔가 일이 벌어졌습니다. 토끼는 어디론가 사라졌습니다. 그 토끼는 어디선가 날아온 건데, 이상하게도 이상한 소리를 내며 날아가고 있었습니다. 그 소리는 바로 '꿈틀거리는 토끼'였습니다."
1942년 4월, 나는 토끼에게 달려들었다.
토끼는 내가 미처 생각하지 못했던 바로 그 소리를 냈습니다.
"아! 이것은 꿈틀거리는 토끼입니다."
"그래요! 꿈틀거리는 토끼입니다."
토끼는 나를 향해 달려가더니 나를 향해 손을 뻗었습니다.
그 순간
```

Section

Tracing - LangChain

실행 결과



RAG 평가

Section 1. LangSmith

Section 1-3. Tracing - LangGraph

Section

Tracing - LangGraph

Tracing

[1]:
```
import os

os.environ["LANGCHAIN_TRACING_V2"]="true"
os.environ["LANGCHAIN_ENDPOINT"]="https://api.smith.langchain.com"
os.environ["LANGCHAIN_API_KEY"]=
os.environ["LANGCHAIN_PROJECT"]="pr-shadowy-scow-41"
```

[2]:
```
from langgraph.graph import StateGraph, END

# 분류 노드
def classification(state):
    text = state.get("user_input", "")

    if any(word in text for word in ["좋아요", "네", "좋다", "응"]):
        label = "positive"
    elif any(word in text for word in ["싫어요", "아니요", "별로", "싫다"]):
        label = "negative"
    else:
        label = "neutral"
    return {**state, "label": label}

# 응답 노드
def positive_answer(state):
    return {**state, "response": "좋게 생각해주셔서 감사합니다!"}

def negative_answer(state):
    return {**state, "response": "무엇이 불편하셨나요?"}

def neutral_answer(state):
    return {**state, "response": "조금 더 자세히 말씀해 주세요."}

def get_label(state):
    return state.get("label", "")
```

Section

Tracing - LangGraph

Tracing

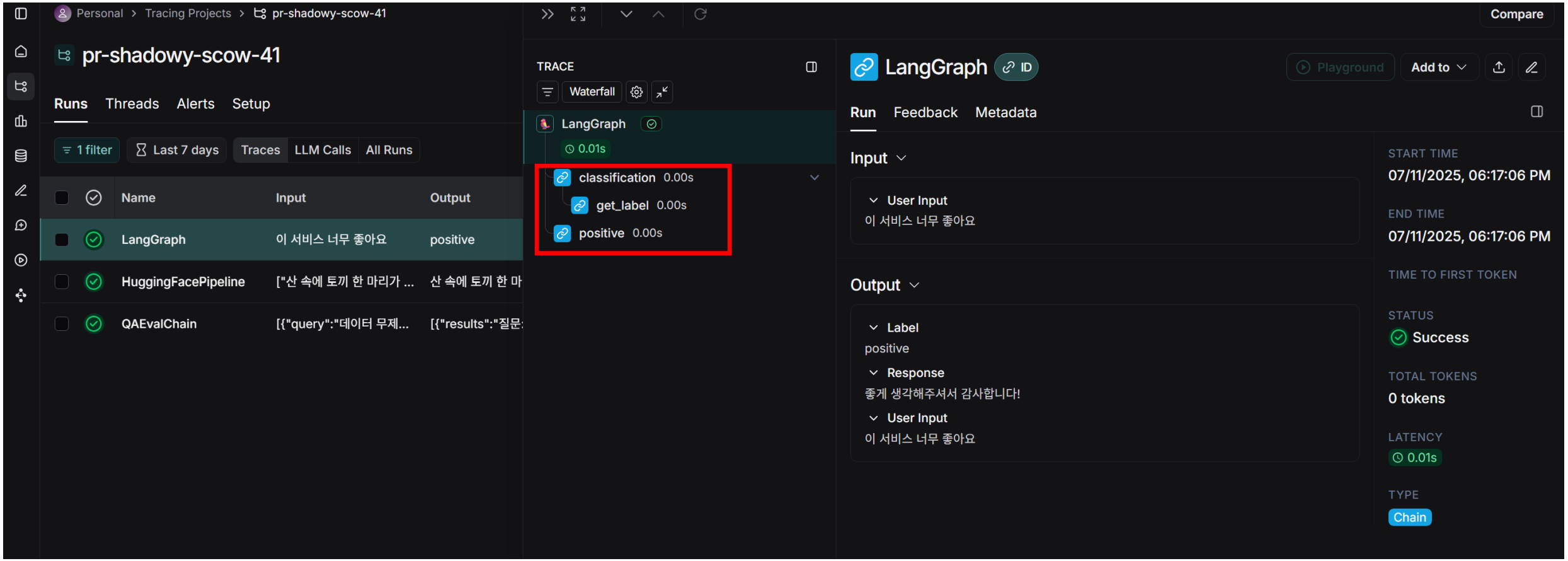
```python
# 그래프 설계
graph = StateGraph(dict)

graph.add_node("classification", classification)
graph.add_node("positive", positive_answer)
graph.add_node("negative", negative_answer)
graph.add_node("neutral", neutral_answer)

graph.set_entry_point("classification")

graph.add_conditional_edges("classification", get_label, {
    "positive": "positive",
    "negative": "negative",
    "neutral": "neutral"
})

# 각 노드 끝나면 종료
graph.add_edge("positive", END)
graph.add_edge("negative", END)
graph.add_edge("neutral", END)

# 5. 실행기 구성
app = graph.compile()

# 6. 테스트 실행 (한글 입력)
user_input = input("한글 입력: ")

final_state = app.invoke({"user_input": user_input})
print("응답:", final_state.get("response", ""))
```

```
한글 입력:  이 서비스 너무 좋아요
응답: 좋게 생각해주셔서 감사합니다!
```

Section

Tracing - LangGraph

Tracing

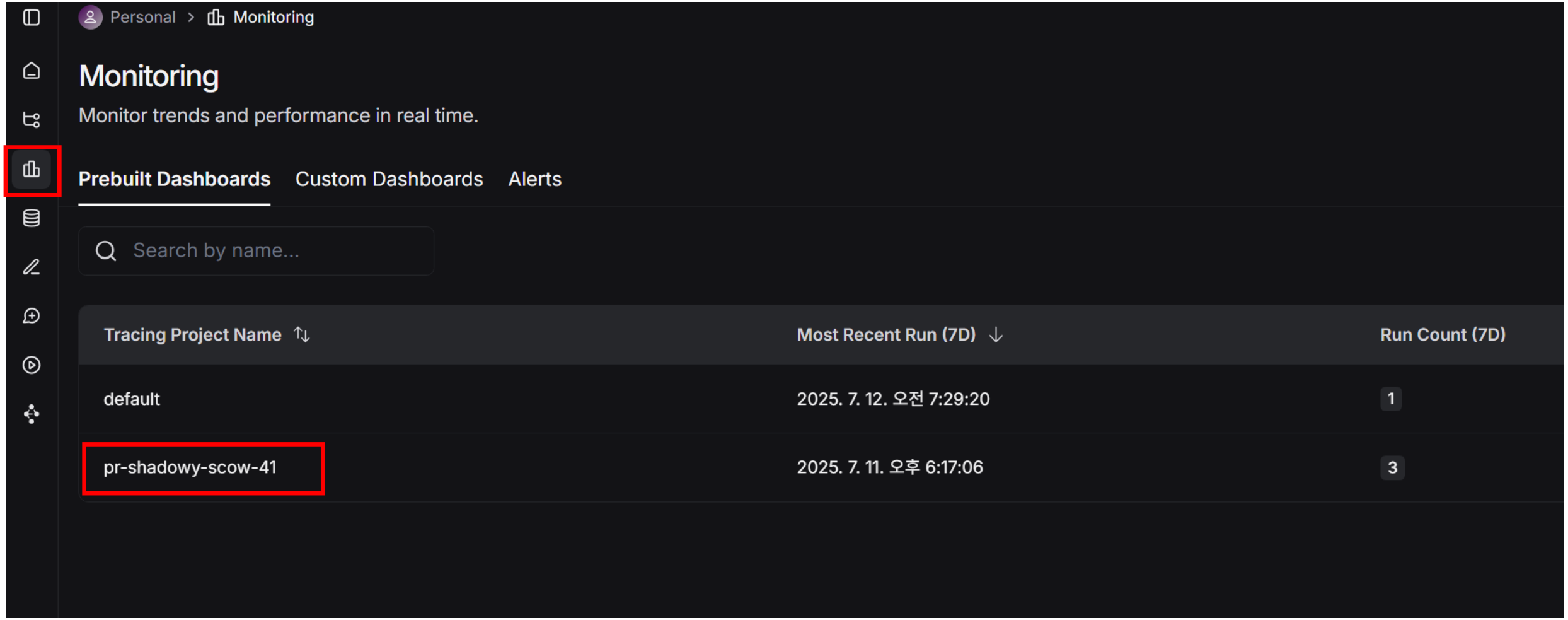


RAG 평가

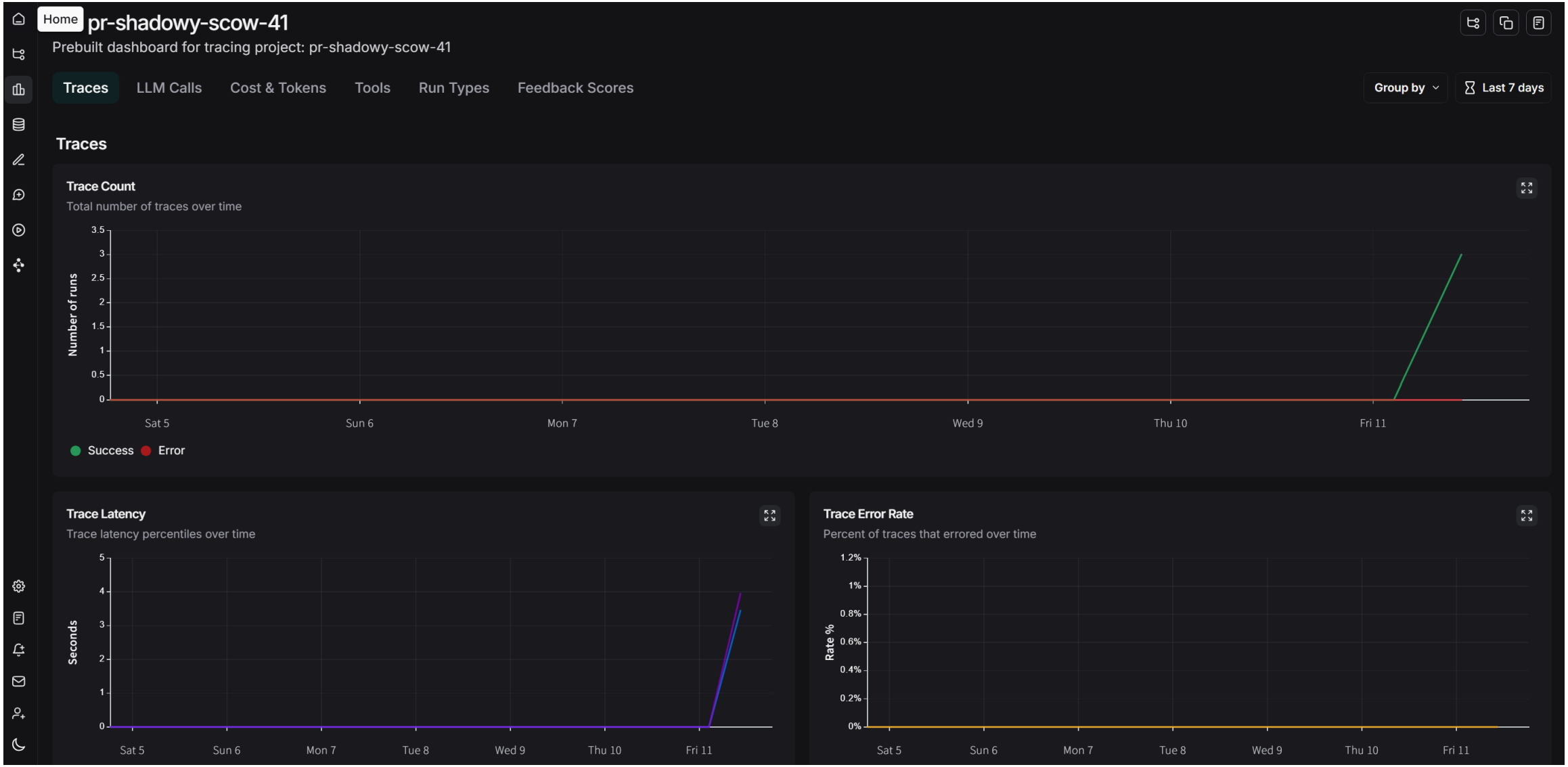
Section 1. LangSmith

Section 2-1. Monitoring

Monitoring


Personal
Monitoring
Monitoring
Monitor trends and performance in real time.
Prebuilt Dashboards
Custom Dashboards
Alerts
Search by name...
Tracing Project Name
Most Recent Run (7D)
Run Count (7D)
default
2025. 7. 12. 오전 7:29:20
1
pr-shadowy-scow-41
2025. 7. 11. 오후 6:17:06
3

Monitoring

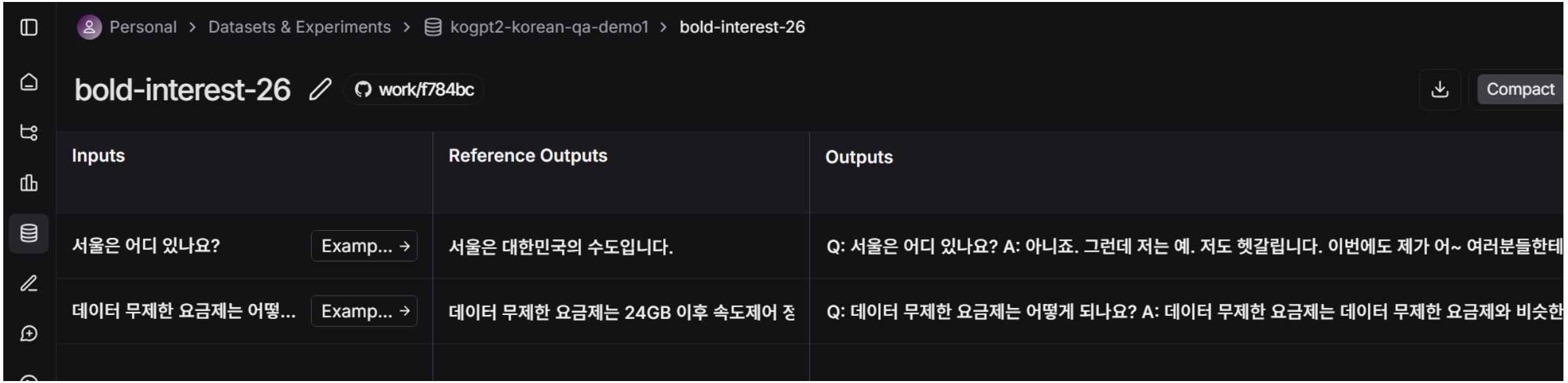

Home
pr-shadowy-scow-41
Prebuilt dashboard for tracing project: pr-shadowy-scow-41
Traces
LLM Calls
Cost & Tokens
Tools
Run Types
Feedback Scores
Group by
Last 7 days
Traces
Trace Count
Total number of traces over time
Number of runs
3.5
3
2.5
2
1.5
1
0.5
0
Sat 5
Sun 6
Mon 7
Tue 8
Wed 9
Thu 10
Fri 11
Success
Error
Trace Latency
Trace latency percentiles over time
Seconds
5
4
3
2
1
0
Sat 5
Sun 6
Mon 7
Tue 8
Wed 9
Thu 10
Fri 11
Trace Error Rate
Percent of traces that errored over time
Rate %
1.2%
1%
0.8%
0.6%
0.4%
0.2%
0%
Sat 5
Sun 6
Mon 7
Tue 8
Wed 9
Thu 10
Fri 11

RAG 평가

Section 1. LangSmith

Section 3-1. Evaluation

Evaluation 평가를 할 수 있긴하지만...



- evaluator API가 까다롭고 불투명
- 오직 OpenAI 기반 evaluator에 맞춰져있음
- 커스텀 evaluator 사용 불편
- evaluate_strings와 같은 비직관적 추상 메소드 강제

감사합니다.

Q & A