

>>

GET STARTED WITH LANGSMITH


Set up observability

Trace, debug and monitor your application

With LangChain

Without LangChain

1. New key created:

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 

2. Install dependencies

Python

TypeScript

```
1 | pip install -U langsmith
```

 Copy

3. Configure environment to connect to LangSmith.

Project Name

strategy-coach


```
1 | LANGSMITH_TRACING="true"
2 | LANGSMITH_ENDPOINT="https://api.smith.langchain.com"
3 | LANGSMITH_API_KEY="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"
4 | LANGSMITH_PROJECT="strategy-coach"
5 | OPENAI_API_KEY="<your-openai-api-key>"
```

 Copy

4. Run any LLM, Chat model, or Chain. Its trace will be sent to the set project.

```
1 | import openai
2 | from langsmith.wrappers import wrap_openai
3 | from langsmith import traceable
4 |
5 | # Auto-trace LLM calls in-context
6 | client = wrap_openai(openai.Client())
7 |
8 | @traceable # Auto-trace this function
9 | def pipeline(user_input: str):
10 |     result = client.chat.completions.create(
11 |         messages=[{"role": "user", "content": user_input}],
12 |         model="gpt-3.5-turbo"
13 |     )
14 |     return result.choices[0].message.content
15 |
16 | pipeline("Hello, world!")
17 | # Out: Hello there! How can I assist you today?
```

 Copy

 Congratulations on logging your first trace in LangSmith. Here are some topics to explore next:

- Read the [conceptual guide](#) to learn about runs, traces, projects and more.
- Set up tracing with [LangChain](#), [LangGraph](#), or the other supported [integration options](#).
- If you prefer video tutorials, check out [Tracing Basics](#) from LangSmith Academy.