

AI Agent 학습 2주차 노트 정리

작성자 문영식

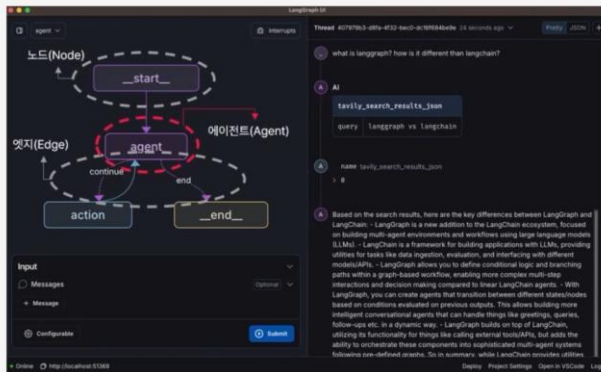
작성일 24년 11월 12일

참고자료/이미지출처 : 모두의 AI 케인의 LangGraph로 끝내는 AI Agent.

1주차에 유튜브로 AI Agent 개념에 대해 알아봤다.

멀티 에이전트 CrewAI, AutoGen, LangGraph 중 LangGraph를 집중적으로 알아보겠다.

Langgraph의 구조



출처: Langchain 공식 블로그

(<https://blog.langchain.dev/langgraph-studio-the-first-agent-ide/>)

노드 : 작업 도구 에이전트 등

엣지 : 화살표 방향 흐름

Langgraph의 코드 구성

State(상태) 정의

```
class State(TypedDict):
    messages: Annotated[list, add_messages]

graph_builder = StateGraph(State)
```

Node(노드) 선언

```
llm = ChatAnthropic(model="claude-3.5-sonnet")

def chatbot(state: State):
    return {"messages": [llm.invoke(state["messages"])]}

graph_builder.add_node("chatbot", chatbot)
```

Edge(엣지) 및 그래프 선언

```
graph_builder.add_edge(START, "chatbot")
graph_builder.add_edge("chatbot", END)
graph = graph_builder.compile()
```

Langgraph의 개념과 구조

공식문서에서 소개하는 Langgraph의 개념

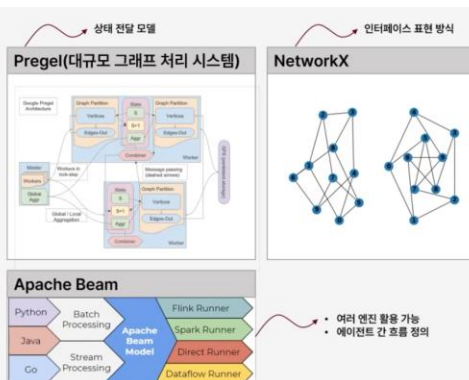
LangGraph: LLM 기반 다중 에이전트 프레임워크

핵심 장점:

- 사이클 지원
 - 세밀한 제어
 - 내장 지속성
- ##### 특징:
- 세밀한 에이전트 워크플로우 생성
 - 순환 구조 가능 (DAG와 차별화)
 - 상태 및 흐름 정밀 제어
 - 고급 휴먼 인터 루프 / 메모리 기능

Langgraph는 어떻게 만들어졌을까?

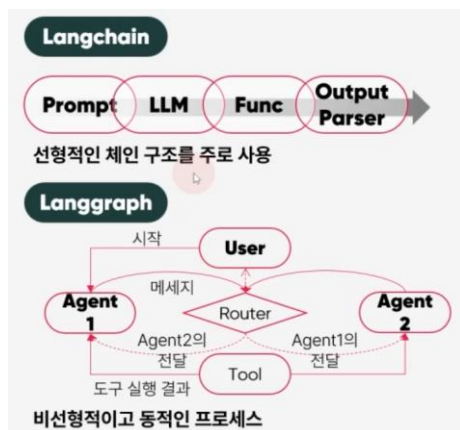
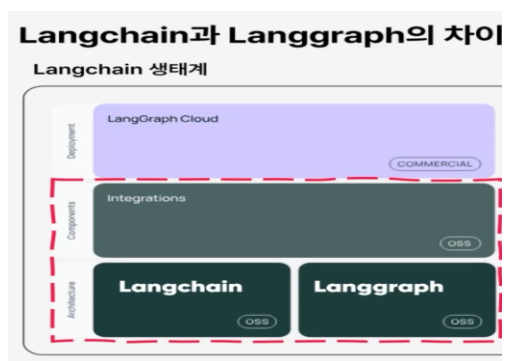
LangGraph는 Pregel과 Apache Beam, NetworkX에서 영감을 얻었습니다.



그래프 : 노드와 노드간의 연결.

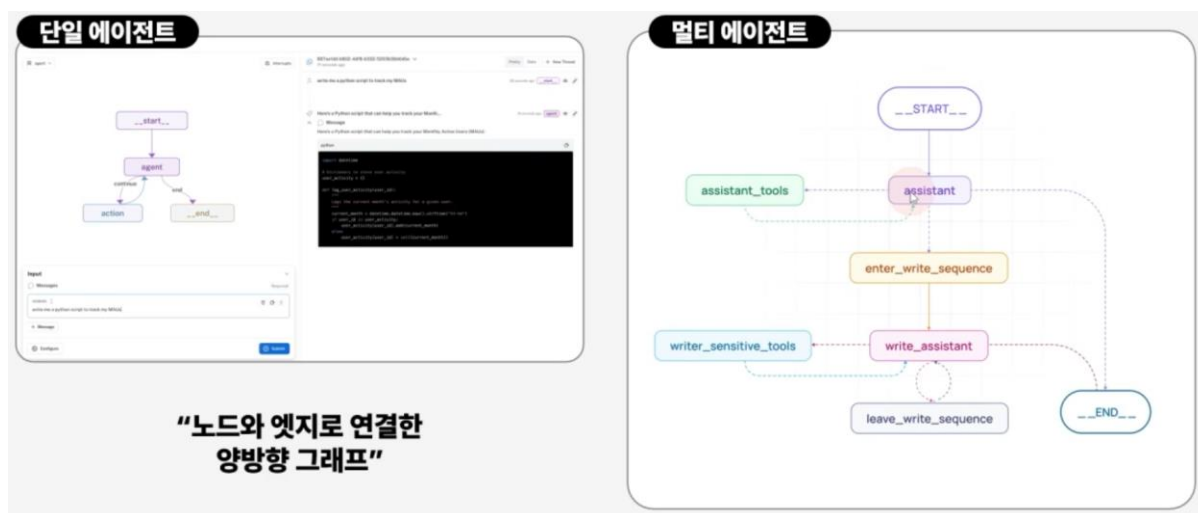
사이클 지원 : 노드(작업, 도구, 에이전트 등)와 노드간의 양방향 작업이 된다.

랭체인과 랭그래프 뭘 차이?



랭체인 : 여러 컴포넌트의 선형적 일방향 사슬 구조.

랭그래프 : 그래프 형태. 판단하고 동적으로 흐름.



주어진 메시지에 따라 판단하고 동적으로 흘러간다~