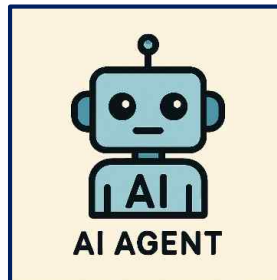
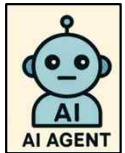


# AI 에이전트와 랭체인 소개



## 1.1 AI 에이전트 소개



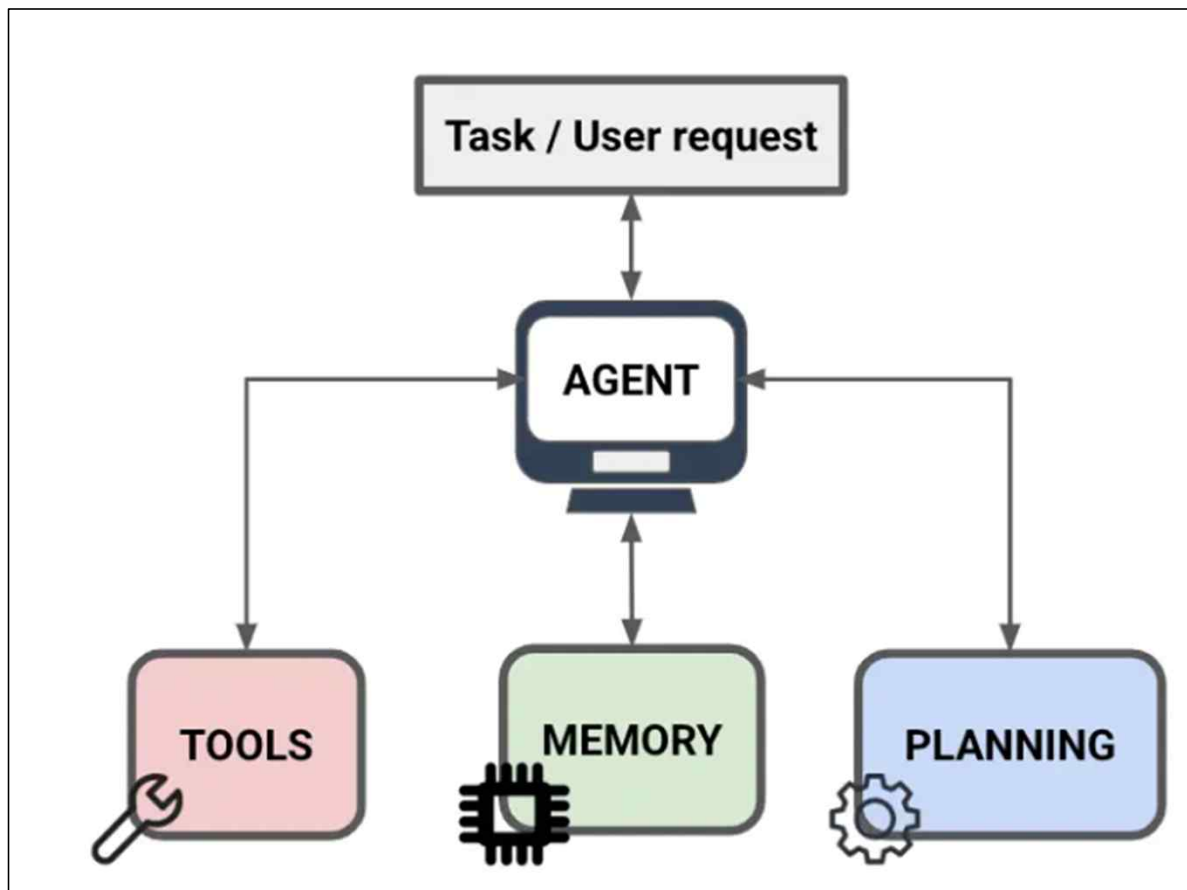
# AI 에이전트란?

**AI 에이전트**는 사용자의 요청이나 환경의 변화를 인식하고, 이를 기반으로 **목표 달성을 위해 자율적으로 판단·행동하는 인공지능 시스템**입니다.

LLM(대규모 언어 모델)을 기반으로 하며, 단순 질의응답(Q&A)을 넘어 **툴 사용, 데이터 검색, 외부 API 호출, 문서 처리** 등 복합적인 작업을 자동화할 수 있습니다.

## 1.1 AI 에이전트 소개

### AI 에이전트의 구성 요소



<https://www.promptingguide.ai/agents/introduction>

# AI 에이전트의 구성 요소별 기능

## 1. Task / User Request (사용자 요청)

- 사용자가 에이전트에게 수행을 요청하는 작업  
예: "오늘 날씨 알려줘", "이 문서 요약해줘", "데이터 분석해줘"
  - 이 요청이 에이전트의 입력이 됩니다.
- 

## 2. Agent (에이전트 본체)

- 중앙 컨트롤러 역할을 하는 인공지능 모듈
- 사용자 요청을 분석하고, 필요한 리소스(도구·메모리·계획)를 조합해 작업을 수행
- LLM(대규모 언어모델) 기반으로 동작하며, 상황에 따라 외부 API나 내부 데이터베이스를 호출

## 1.1 AI 에이전트 소개

### 3. TOOLS (도구)

- 에이전트가 작업을 수행하기 위해 호출할 수 있는 기능 모듈
  - 예:
    - 웹 검색 API
    - 계산기
    - 데이터베이스 쿼리
    - 파일 읽기/쓰기
  - LangChain에서는 "Tool" 객체로 정의해 에이전트가 선택적으로 호출
- 

### 4. MEMORY (메모리)

- 에이전트가 이전 대화·작업 이력을 기억하는 저장소
- 단기 메모리: 현재 세션의 문맥 유지
- 장기 메모리: 여러 세션에 걸쳐 지속적으로 기억
- 예:
  - 사용자의 이름/취향 저장
  - 지난 회의 내용 기억
- 이를 통해 연속적이고 개인화된 대화 가능

## 1.1 AI 에이전트 소개

### 5. PLANNING (계획 수립)

- 복잡한 작업을 수행할 때, 어떤 순서로 어떤 도구와 데이터를 사용할지 결정
- 예:
  1. 사용자 요청 분석
  2. 필요한 데이터 수집(툴 호출)
  3. 결과 가공
  4. 최종 답변 생성
- "Task decomposition(작업 분해)"과 "Reasoning(추론)" 과정이 포함

#### 동작 흐름

1. 사용자 요청이 들어옴
2. **Agent**가 요청을 해석
3. 필요 시 **계획(Planning)** 단계에서 처리 순서 결정
4. 필요한 **\*\*도구(Tools)\*\***나 **\*\*메모리(Memory)\*\***를 활용
5. 최종 결과를 사용자에게 응답

## 1.1 AI 에이전트 소개

AI 에이전트는 **인식, 처리, 행동, 학습 및 적용, 자율성**의 5가지 주요 특징을 바탕으로 수행 및 문제 해결

특징	내용
인식(Perception)	<ul style="list-style-type: none"><li>• 센서, 데이터 파일, 인터넷 등 다양한 데이터 수집 방법을 통해 주변 환경 인식 및 파악</li></ul>
처리(Brain)	<ul style="list-style-type: none"><li>• 입력 데이터 기반 주요 정보 추출 및 새로운 정보 학습을 통한 의사 결정</li></ul>
행동(Action)	<ul style="list-style-type: none"><li>• 도출된 결정, 해답 기반 텍스트 생성, API 호출, 물리적 동작 등 실제 수행</li></ul>
학습 및 적용 (Learning and Adaptation)	<ul style="list-style-type: none"><li>• 경험 기반 지속적 학습, 과거 데이터를 기반으로 미래 행동 개선</li></ul>
자율성(Autonomy)	<ul style="list-style-type: none"><li>• 스스로 결정을 내리고 행동하며 인간의 개입 없이 목표를 달성하기 위한 작업 수행</li></ul>

\* 자료 : SK C&C(2024)



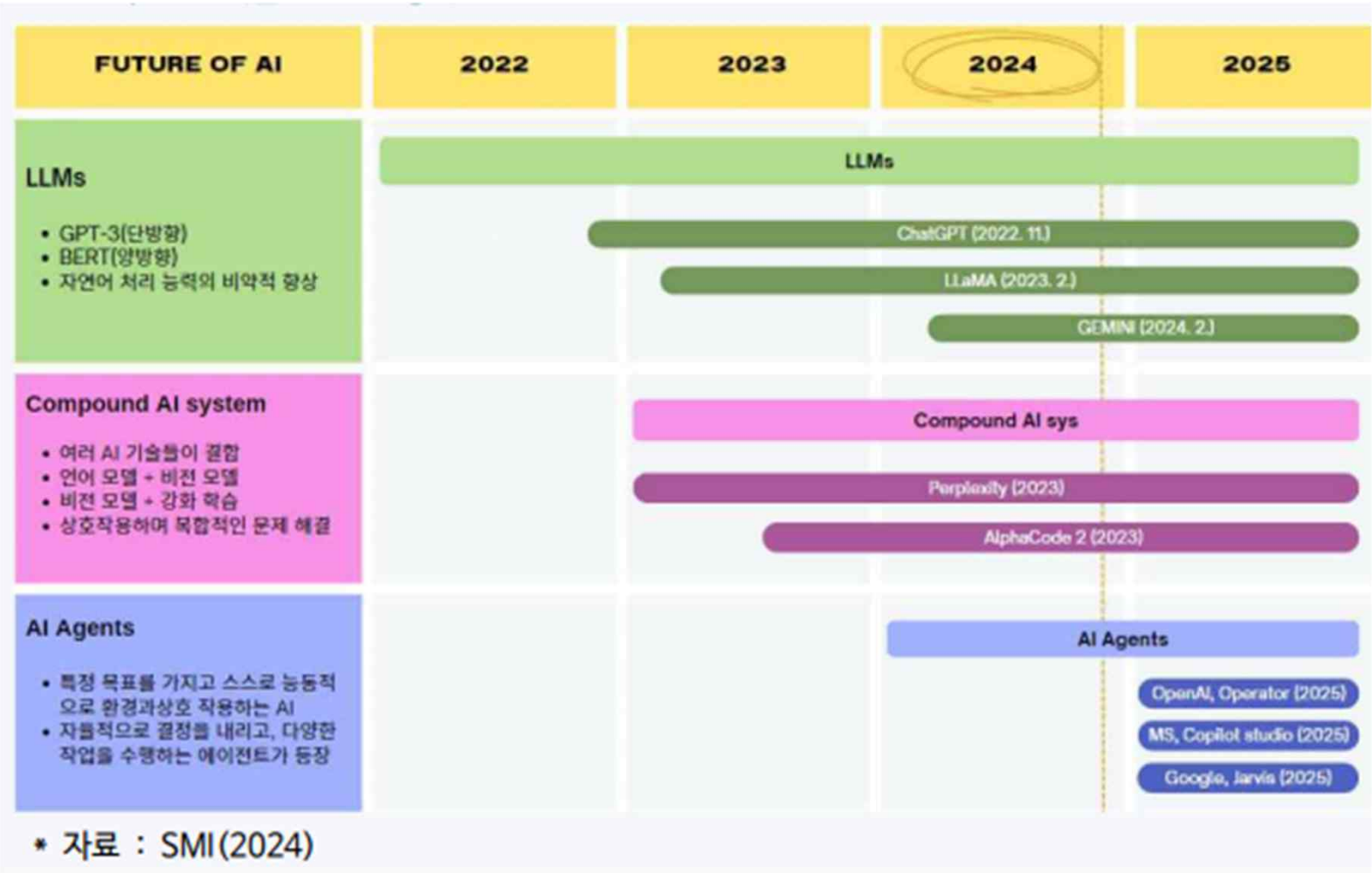
# 1.1 AI 에이전트 소개

	 일반적인 언어 모델	 AI 에이전트
사용 범위	프로세스 자동화	전체 워크플로우 및 프로세스 자동화
계획	워크플로우를 계획하거나 조율할 수 없는 경우	사용자의 목표를 달성하기 위해 다단계 계획을 수립 및 실행하여, 실시간 피드백에 따라 작업 조정
메모리 및 미세조정	메모리 유지 및 미세 조정 기능이 제한적	단기 및 장기 메모리를 활용하여 이전 사용자 상호 작용에서 학습하고 개인화된 응답을 제공하며, 시스템 내 여러 에이전트 간에 메모리 공유 가능
도구 통합	본질적으로 외부 도구 또는 시스템과 통합되도록 설계되지 않음	API와 도구(데이터 추출기, 이미지 선택기, 검색API)로 내재된 언어 모델의 기능을 강화하여 작업 수행
데이터 통합	고정된 학습 데이터와 훈련 시점에 기반한 정적 지식에 의존	새로운 정보와 실시간 지식 소스에 동적으로 적응
정확성	일반적으로 자체 평가 기능이 부족하고 학습 데이터에 기반한 확률적 추론으로 제한	특정 작업에 특화된 기능, 지식, 메모리를 활용하여 시스템 내 자체 출력물과 다른 에이전트의 출력물을 검증하고 개선 가능

\* 자료 : Deloitte(2024) 재구성

# 1.1 AI 에이전트 소개

AI 기술 발전 현황과 주요 모델 , AI 에이전트



# AI 에이전트의 일반적인 사용 사례

---

업계에서 에이전트가 적용되는 일반적인 사용 사례의 전체 목록은 다음과 같습니다.

- 추천 시스템: 제품, 서비스 또는 콘텐츠에 대한 제안을 개인화합니다.
- 고객 지원 시스템: 문의 처리, 문제 해결, 지원 제공.
- 연구: 법률, 금융, 건강 등 다양한 분야에 걸쳐 심층적인 조사를 수행합니다.
- 전자상거래 애플리케이션: 온라인 쇼핑 경험 향상, 주문 관리, 개인화된 추천 제공.
- 예약: 여행 준비 및 이벤트 계획 지원.
- 보고: 방대한 양의 데이터를 분석하고 포괄적인 보고서를 생성합니다.
- 재무 분석: 시장 동향을 분석하고, 재무 데이터를 평가하고, 전례 없는 속도와 정확성으로 보고서를 생성합니다.

<https://www.promptingguide.ai/agents/introduction>

## 1.2 랭체인 소개

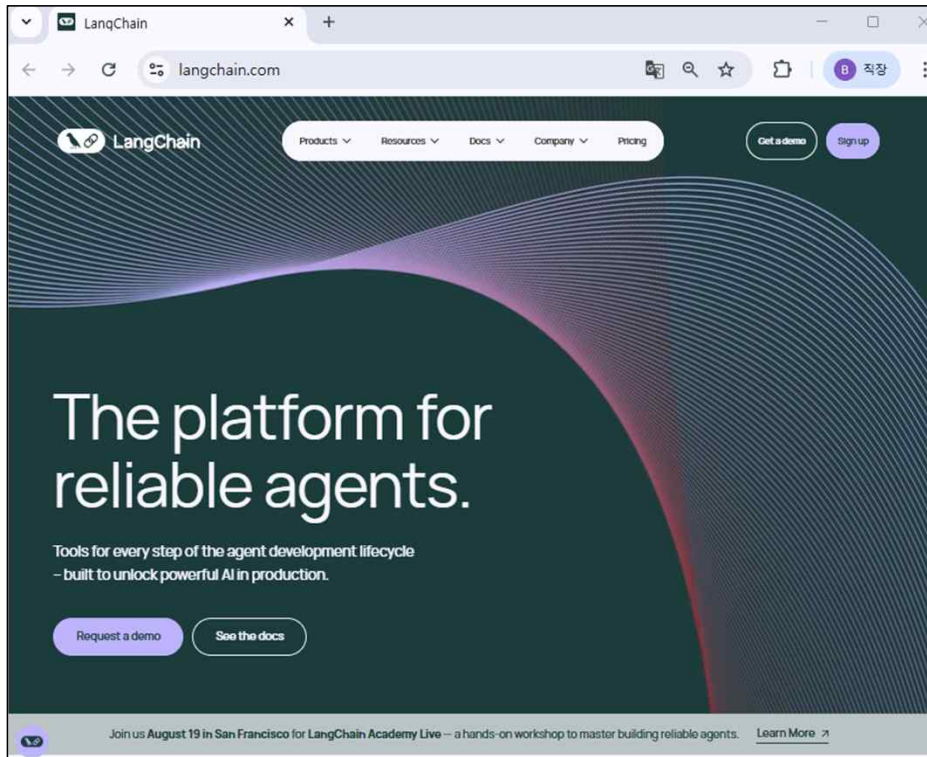


## 1.2 랭체인 소개

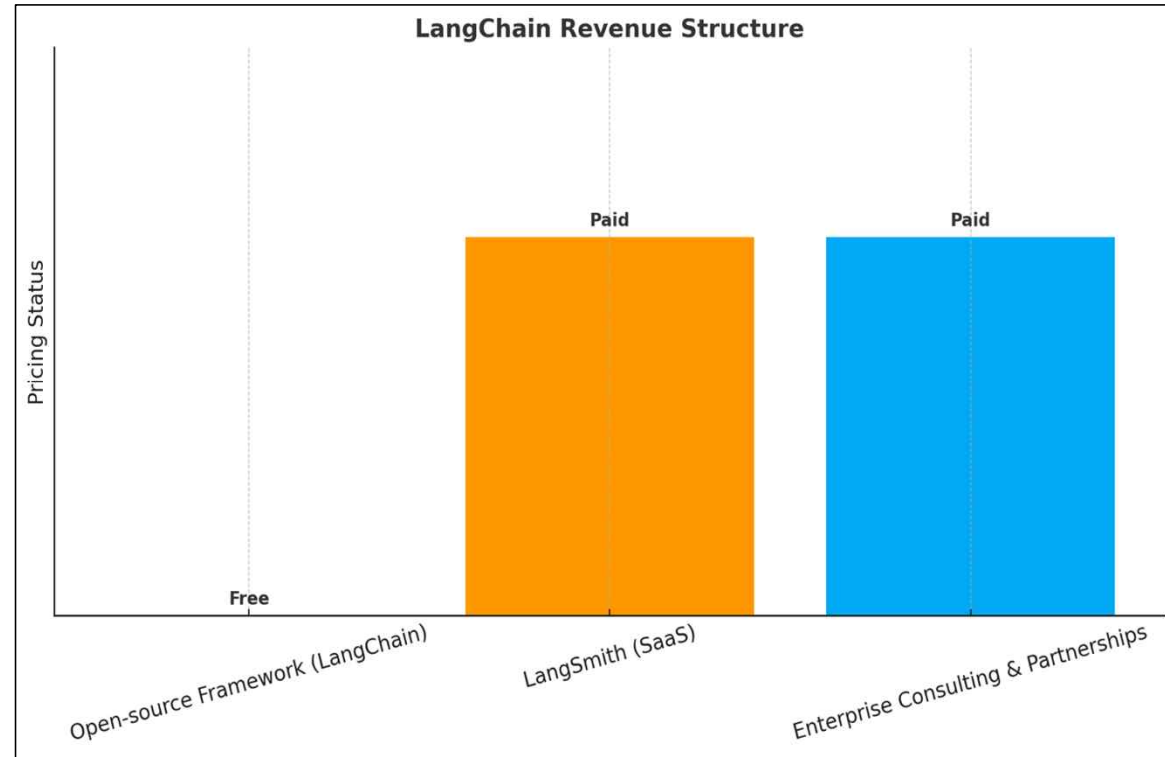
### 랭체인이란?

- 랭체인(LangChain): 언어 모델 기반 AI 애플리케이션을 쉽게 개발할 수 있도록 돕는 프레임워크.
- 장점
  - 검색, 문서 생성, 질의응답 등 기능을 손쉽게 구현 가능
  - 복잡한 애플리케이션 개발 시 미리 준비된 모듈을 활용해 개발 속도 향상
- 특징
  - 다양한 언어 모델(Gemini, Llama, Claude, GPT 등) 간 손쉬운 교체 가능
  - 코드 변경 없이 언어 모델을 바꿔 쓸 수 있어 특정 모델의 장점을 활용 가능
- 활용 효과
  - 특정 모델에 종속되지 않고 유연하고 확장성 있는 AI 애플리케이션 개발 가능

## 1.2 랭체인 소개



<https://www.langchain.com/>



랭체인의 수익 구조



## 1.2 랭체인 소개

### 1 프로젝트 시작 배경

- 2022년 10월, 해리슨 체이스(Harrison Chase)가 오픈 소스 프로젝트로 시작
- Robust Intelligence에서 근무하면서 **LLM 애플리케이션 개발에 필요한 공통 패턴과 도구의 부재**를 느끼고,  
이를 해결하기 위해 누구나 쉽게 챗봇, 질의응답, 요약 시스템을 만들 수 있는 프레임워크 구상

### 2 주요 목적과 특징

- **LLM 애플리케이션과 파이프라인을 신속하게 구축**할 수 있는 플랫폼 제공
- 프롬프트 관리, 체인(Chain) 구성, 외부 데이터 연결(RAG), 메모리·에이전트 기능까지  
**LLM 활용에 필요한 모든 구성요소를 모듈화**해 개발자가 재사용 가능하도록 설계

### 3 성장 과정

- 2023년 4월, 법인 전환 및 세쿼이아 캐피탈 등 VC 투자 유치 → 커뮤니티와 생태계 빠르게 성장
- 오픈소스 커뮤니티 중심으로 다양한 통합(데이터베이스, 벡터DB, API 등) 생태계 확장

### 4 최신 동향

- 2024년 1월, **v0.1.0 첫 안정(stable) 버전** 공개
- 그 전까지 지적받던 잦은 업데이트·호환성 문제·버그를 대폭 개선
- 보다 일관된 **API**와 개발자 친화적 문서 제공으로 프로덕션에서도 활용하기 쉬워짐

## 1.2 랭체인 소개

### 랭체인이란?

- 랭체인은 챗봇, Q&A 시스템 등 **LLM 기반 애플리케이션**을 쉽게 만들 수 있도록 돕는 프레임워크입니다.
- LLM 애플리케이션 개발 시 필요한 다양한 도구를 연결하고 활용하는 역할을 합니다.
- 예:
  - **Document Loader**로 문서를 불러올 수 있음
  - `WebBaseLoader` → 웹 페이지 내용을 불러오기
  - `PDFLoader` → PDF 내용 불러오기
  - `Text Splitter` → 긴 텍스트를 분할하여 처리

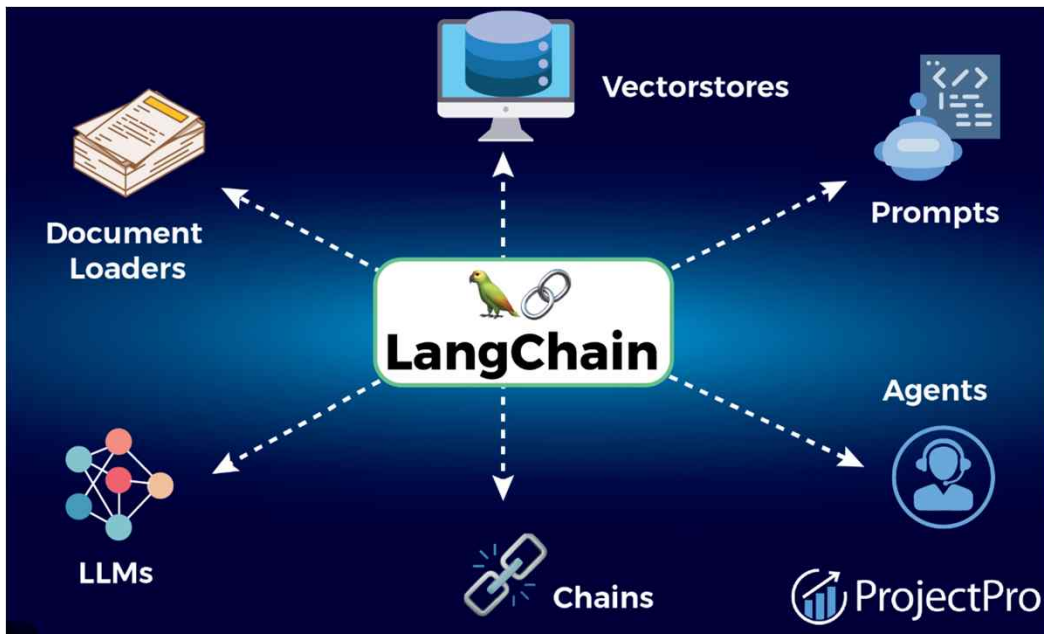
즉, 랭체인은 **LLM 애플리케이션 제작에 필요한 데이터 로딩·처리 도구들을 쉽게 연계할 수 있게** 해주는 도구입니다.



## 1.2 랭체인 소개

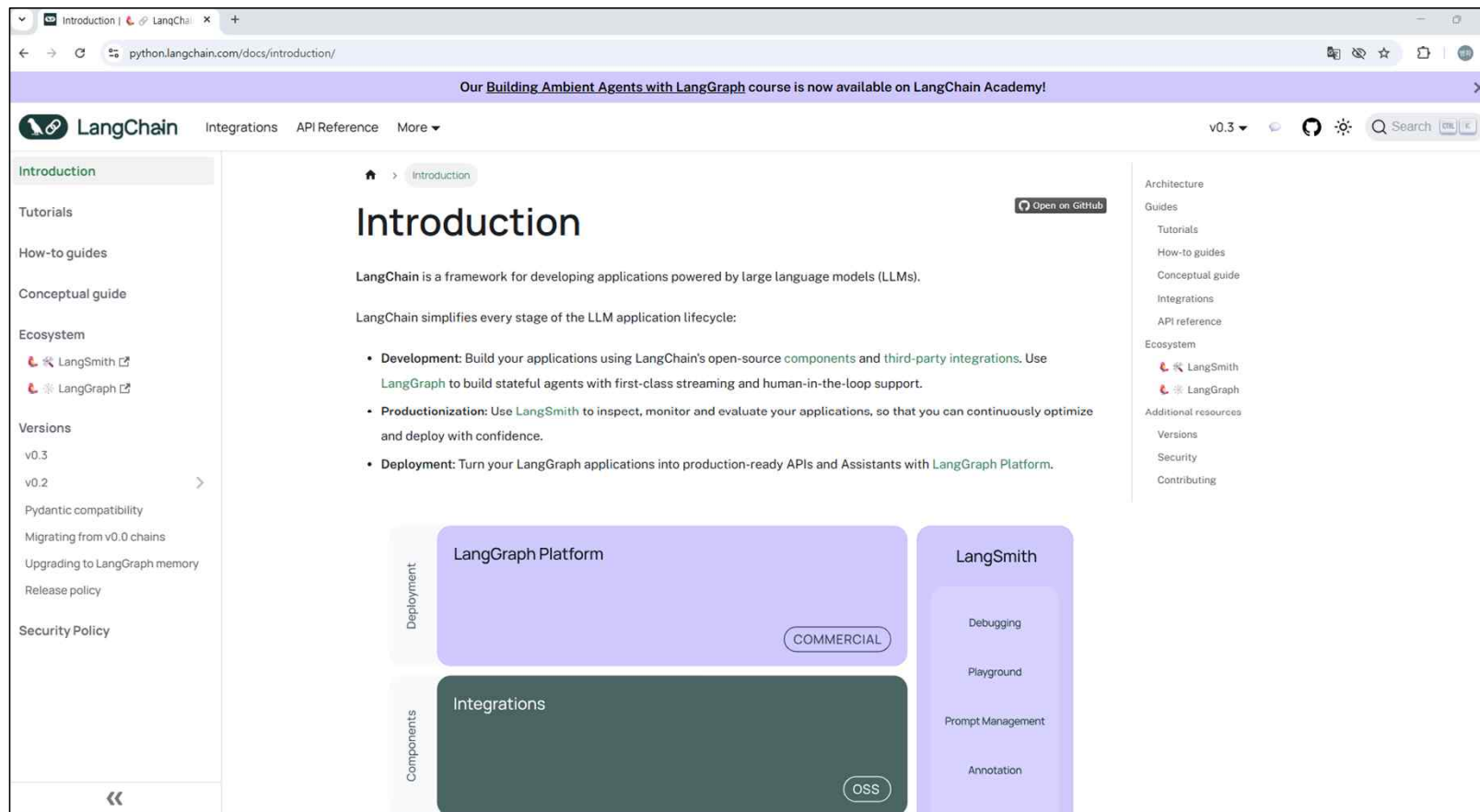
### 🌐 LangChain 구성 요소

- **Language** → 언어 모델(LLM, Large Language Model)을 의미
- **Chain** → 여러 기능(프롬프트, 데이터 로더, 파서 등)을 연결(**Chaining**) 하여 파이프라인처럼 동작



## 1.2 랭체인 소개

랭체인 개발자를 위한 공식 사이트 : <https://python.langchain.com/>



# 소개

[Open on GitHub](#)

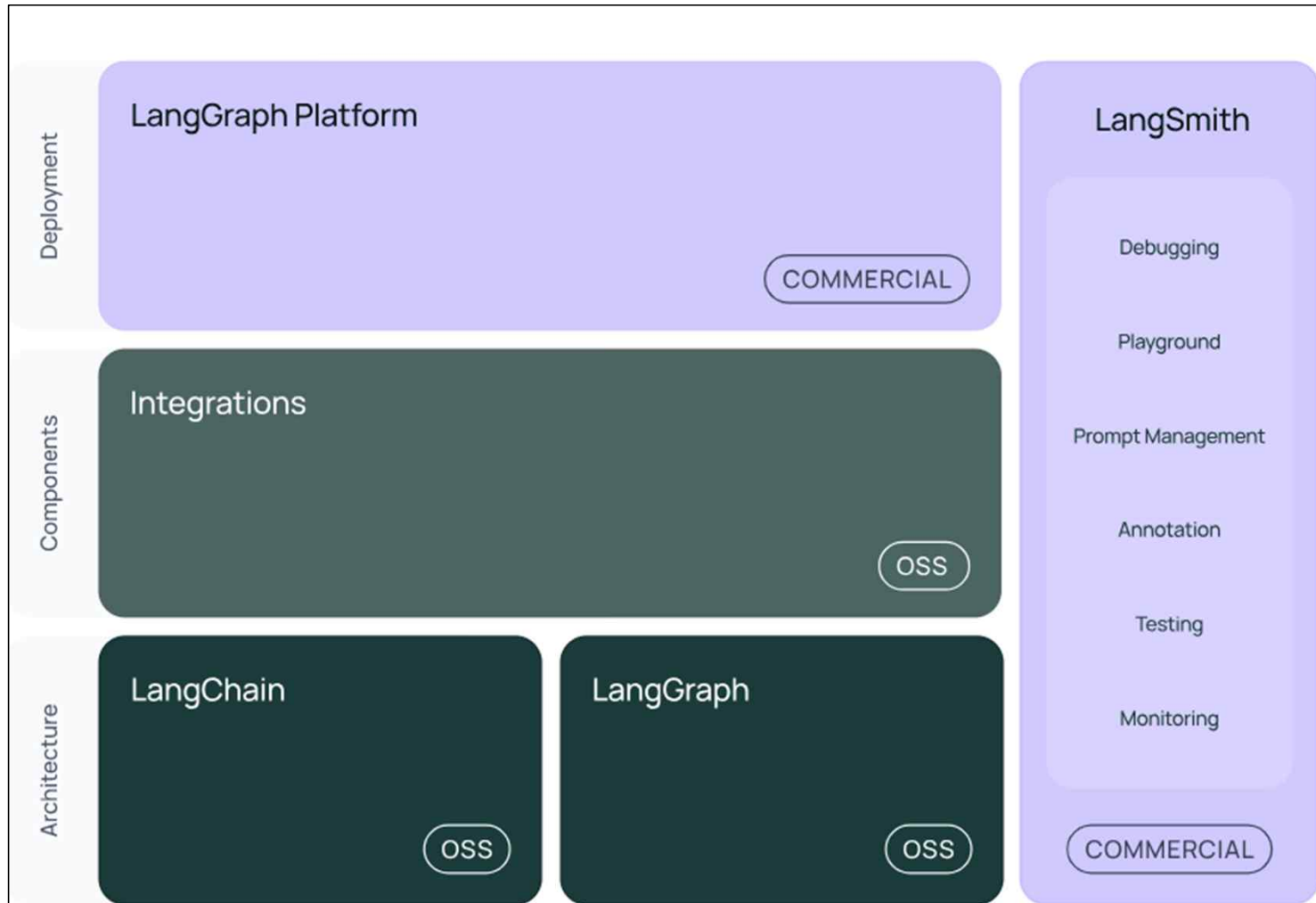
LangChain은 대규모 언어 모델(LLM)을 기반으로 애플리케이션을 개발하기 위한 프레임워크입니다.

LangChain은 LLM 애플리케이션 라이프사이클의 모든 단계를 단순화합니다.

- **개발** : LangChain의 오픈 소스 구성 요소와 타사 통합을 활용하여 애플리케이션을 구축하세요. LangGraph를 사용하여 최고 수준의 스트리밍 및 HIL(Human-in-the-Loop) 지원을 제공하는 상태 저장 에이전트를 구축하세요.
- **생산화** : LangSmith를 사용하여 애플리케이션을 검사, 모니터링 및 평가함으로써 지속적으로 최적화하고 자신감을 가지고 배포할 수 있습니다.
- **배포** : LangGraph 플랫폼을 사용하여 LangGraph 애플리케이션을 프로덕션에 바로 적용 가능한 API와 Assistant로 전환하세요.

<https://python.langchain.com/docs/introduction/>

## 1.2 랭체인 소개

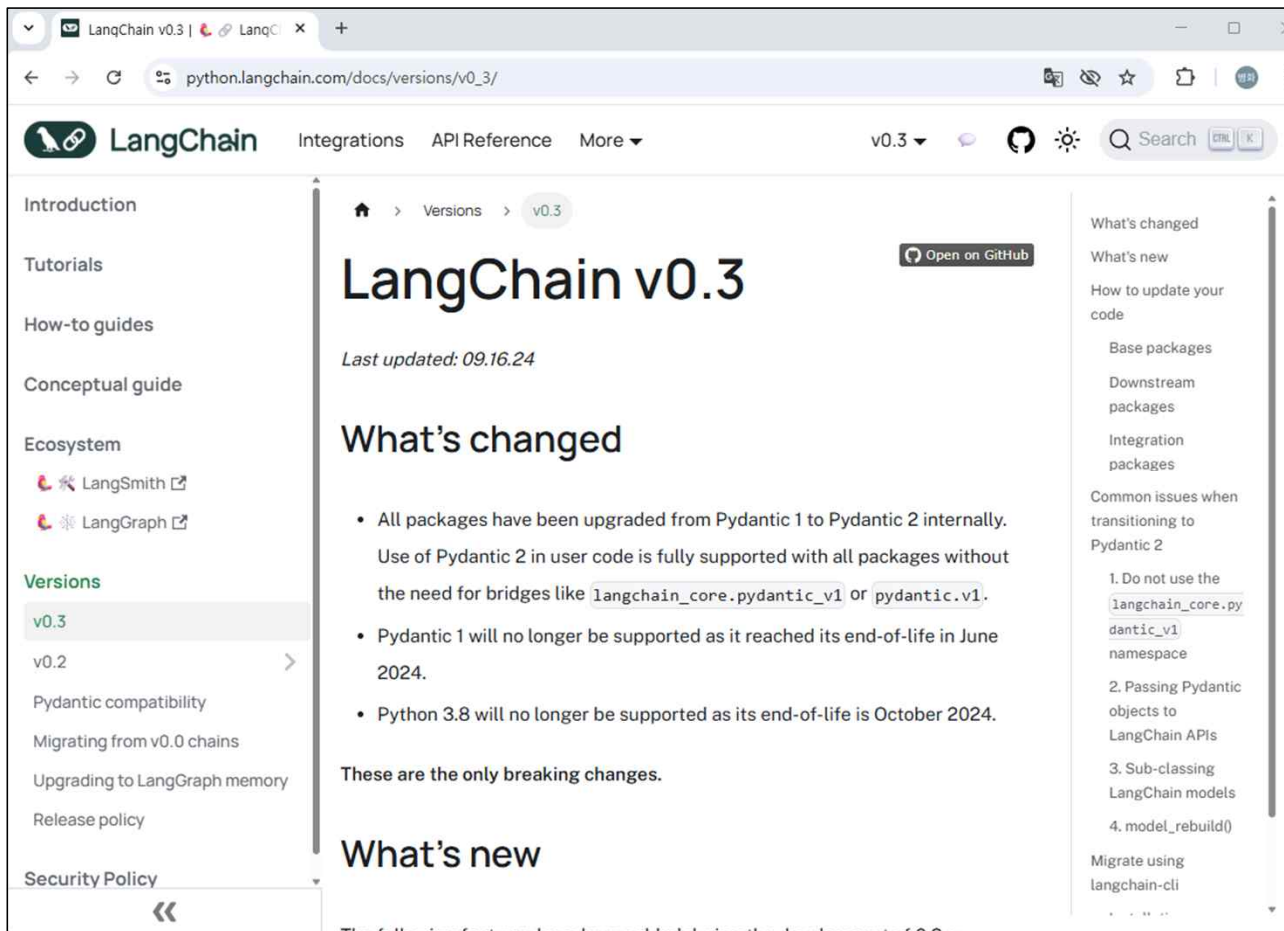


<https://python.langchain.com/docs/introduction/>

## **1.3    랭체인 환경 설정**


## 1.3 랭체인 환경 설정


랭체인 v0.3 공식 문서 : [https://python.langchain.com/docs/versions/v0\\_3/](https://python.langchain.com/docs/versions/v0_3/)



## 1.3 랭체인 환경 설정

 랭체인 패키지 설치 : **`pip install langchain`**

 > [How-to guides](#) > [How to install LangChain packages](#)

 [Open on GitHub](#)

# How to install LangChain packages

The LangChain ecosystem is split into different packages, which allow you to choose exactly which pieces of functionality to install.

## Official release

To install the main `langchain` package, run:

Pip

Conda

```
pip install langchain
```

[https://python.langchain.com/docs/how\\_to/installation/](https://python.langchain.com/docs/how_to/installation/)

### 1.3 랭체인 환경 설정

pip install langchain

- langchain-core (기본 로직)
- 문서 로더(Document Loaders)
- 벡터 스토어(Vector Stores)
- 에이전트/툴 연결(Agents & Tools)
- 통합 API 클라이언트(OpenAI, HuggingFace, Pinecone 등)

pip install langchain-core

- **LangChain의 핵심 로직만 설치**
- 프롬프트 관리, 콜백, 실행 체인 같은 **코어 기능만 제공**
- 벡터DB, 데이터 로더, 외부 API 커넥터 등은 **포함되지 않음**
- 가볍게 필요한 기능만 쓰거나, 맞춤형 최소 환경을 구성할 때 사용

명령어	포함 범위	용도
pip install langchain	코어 + 확장 모듈 + 통합 라이브러리	대부분의 개발자, 전체 기능 사용
pip install langchain-core	코어 기능만	경량 설치, 맞춤형 환경 구성



## 1.3 랭체인 환경 설정

Jupyter Notebook 에서 LangChain 패키지 설치 : **%pip install langchain langchain-openai**

```
[1]: %pip install langchain langchain-openai
```

```
Collecting langchain
  Downloading langchain-0.3.27-py3-none-any.whl.metadata (7.8 kB)
Collecting langchain-openai
  Downloading langchain_openai-0.3.30-py3-none-any.whl.metadata (2.4 kB)
Collecting langchain-core<1.0.0,>=0.3.72 (from langchain)
  Downloading langchain_core-0.3.74-py3-none-any.whl.metadata (5.8 kB)
Collecting langchain-text-splitters<1.0.0,>=0.3.9 (from langchain)
  Downloading langchain_text_splitters-0.3.9-py3-none-any.whl.metadata (1.9 kB)
Collecting langsmith>=0.1.17 (from langchain)
  Downloading langsmith-0.4.14-py3-none-any.whl.metadata (14 kB)
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in c:\users\storm\anaconda3\lib\site-packages (from langchain) (2.8.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in c:\users\storm\anaconda3\lib\site-packages (from langchain) (2.0.34)
Requirement already satisfied: requests<3,>=2 in c:\users\storm\anaconda3\lib\site-packages (from langchain) (2.32.3)
Requirement already satisfied: PyYAML>=5.3 in c:\users\storm\anaconda3\lib\site-packages (from langchain) (6.0.1)
Collecting openai<2.0.0,>=1.99.9 (from langchain-openai)
  Downloading openai-1.99.9-py3-none-any.whl.metadata (29 kB)
Collecting tiktoken<1,>=0.7 (from langchain-openai)
  Downloading tiktoken-0.11.0-cp312-cp312-win_amd64.whl.metadata (6.9 kB)
Requirement already satisfied: tenacity!=8.4.0,<10.0.0,>=8.1.0 in c:\users\storm\anaconda3\lib\site-packages (from langchain-openai) (8.2.3)
Requirement already satisfied: jsonpatch<2.0,>=1.33 in c:\users\storm\anaconda3\lib\site-packages (from langchain-core<1.0.0,>=0.3.72) (1.33)
Requirement already satisfied: typing-extensions>=4.7 in c:\users\storm\anaconda3\lib\site-packages (from langchain-core<1.0.0,>=0.3.72) (4.7.1)
```

감사합니다