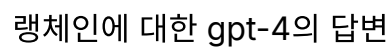




이외에 Google Gemini, META LLaMA Cohere etc...





나

Langchain을 이해하기 쉽게 설명해줄래?



**ChatGPT**

Langchain은 인공지능 언어 모델을 쉽게 배포하고, 통합하며 확장할 수 있도록 돕는 오픈 소스 프레임워크입니다. 이 플랫폼의 목표는 개발자들이 복잡한 인프라 구축 작업 없이도 AI 언어 모델을 사용하여 응용 프로그램을 만들 수 있도록 지원하는 것입니다. 여기에는 대화형 AI, 자동화된 텍스트 생성 및 분석 등 여러 가지 기능이 포함됩니다.

Langchain을 사용하는 주된 이점은 다음과 같습니다:

1. **접근성 향상:** 다양한 AI 언어 모델을 쉽게 연동하여 사용할 수 있으며, 프로그래밍에 대한 깊은 지식이 없는 사용자도 활용할 수 있습니다.
2. **확장성:** 서로 다른 소스에서 제공하는 여러 언어 모델을 하나의 프레임워크 안에서 조합하여 사용할 수 있어, 다양한 요구 사항과 환경에 맞춤화할 수 있습니다.
3. **개발 효율성:** 기존의 복잡한 코드를 재작성하지 않고도, 새로운 기능과 모델을 쉽게 추가하고 업데이트 할 수 있습니다.

Langchain은 특히 사용자가 자신의 애플리케이션에 적합한 도구와 모델을 선택하고, 이를 유연하게 통합할 수 있도록 다양한 지원을 제공합니다. 이로 인해 개발자는 AI 기반 애플리케이션을 더 빠르고 효율적으로 개발할 수 있습니다.

### LangChain 공식 문서 속 소개 글

LangChain은 언어 모델로 구동되는 애플리케이션을 개발하기 위한 프레임워크입니다.

가장 강력하고 차별화된 애플리케이션은 API를 통해 언어 모델을 호출할 뿐만 아니라 이를 통해 개발될 것이라고 믿습니다:

- **데이터 인식:** 언어 모델을 다른 데이터 소스에 연결합니다.
- **에이전트 기능:** 언어 모델이 환경과 상호 작용할 수 있도록 합니다.



ChatGPT

### • 1. 정보 접근 제한

ChatGPT(GPT-3.5)는 2021년까지의 데이터를 학습한 LLM(초거대언어모델)이므로, 2022년부터의 정보에 대해서는 답변을 하지 못하거나, 거짓된 답변을 제공한다.

### • 2. 토큰 제한

ChatGPT에서 제공하는 모델인 GPT-3.5와 GPT-4는 각각 4096, 8192토큰이라는 입력 토큰 제한이 존재한다.

### • 3. 환각현상(Hallucination)

Fact에 대한 질문을 했을 때, 엉뚱한 답을 하거나 거짓말을 하는 경우가 많다.



ChatGPT

### • 1. 정보 접근 제한

ChatGPT(GPT-3.5)는 2021년까지의 데이터를 학습한 LLM(초거대언어모델)이므로, 2022년부터의 정보에 대해서는 답변을 하지 못하거나, 거짓된 답변을 제공한다.

### • 2. 토큰 제한

ChatGPT에서 제공하는 모델인 GPT-3.5와 GPT-4는 각각 4096, 8192토큰이라는 입력 토큰 제한이 존재한다.

### • 3. 환각현상(Hallucination)

Fact에 대한 질문을 했을 때, 엉뚱한 답을 하거나 거짓말을 하는 경우가 많다.

## LangChain은 In-context Learning의 도구

ChatGPT 개량

→ Fine-tuning:

기존 딥러닝 모델의 weight를 조정하여 원하는 용도의 모델로 업데이트 합니다.

→ N-shot Learning:

0개~n개의 출력 예시를 제시하여, 딥러닝이 용도에 알맞은 출력을 하도록 조정합니다.

→ Fine-tuning:

기존 딥러닝 모델의 weight를 조정하여 원하는 용도의 모델로 업데이트 합니다.

→ N-shot Learning:

0개~n개의 출력 예시를 제시하여, 딥러닝이 용도에 알맞은 출력을 하도록 조정합니다.

→ In-context Learning:

문맥을 제시하고, 이 문맥 기반으로 모델이 출력하도록 조정합니다.



ChatGPT

•1. 정보 접근 제한

ChatGPT(GPT-3.5)는 2021년까지의 데이터를 학습한 LLM(초거대언어모델)이므로, 2022년부터의 정보에 대해서는 답변을 하지 못하거나, 거짓된 답변을 제공한다.



Vectorstore 기반 정보 탐색 or Agent 활용한 검색 결합

•2. 토큰 제한

ChatGPT에서 제공하는 모델인 GPT-3.5와 GPT-4는 각각 4096, 8192토큰이라는 입력 토큰 제한이 존재한다.



TextSplitter를 활용한 문서 분할

•3. 환각현상(Hallucination)

Fact에 대한 질문을 했을 때, 엉뚱한 대답을 하거나 거짓말을 하는 경우가 많다.



주어진 문서에 대해서만 답하도록 Prompt 입력

## LangChain의 구조



LangChain

**LLM**

: 초거대 언어모델로, 생성 모델의 엔진과 같은 역할을 하는 핵심 구성 요소

예시: GPT-3.5, PALM-2, LLAMA, StableVicuna, WizardLM, MPT,...

**Prompts**

: 초거대 언어모델에게 지시하는 명령문

요소: Prompt Templates, Chat Prompt Template, Example Selectors, Output Parsers

**Index**

: LLM이 문서를 쉽게 탐색할 수 있도록 구조화 하는 모듈

예시: Document Loaders, Text Splitters, Vectorstores, Retrievers,...

**Memory**

: 채팅 이력을 기억하도록 하여, 이를 기반으로 대화가 가능하도록 하는 모듈

예시: ConversationBufferMemory, Entity Memory, Conversation Knowledge Graph Memory,...

**Chain**

: LLM 사슬을 형성하여, 연속적인 LLM 호출이 가능하도록 하는 핵심 구성 요소

예시: LLM Chain, Question Answering, Summarization, Retrieval Question/Answering,...

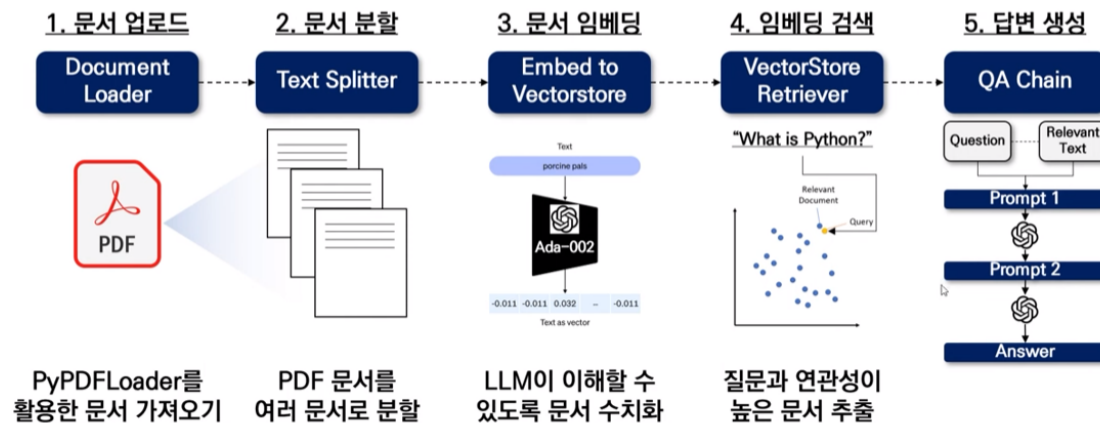
**Agents**

: LLM이 기존 Prompt Template으로 수행할 수 없는 작업을 가능케하는 모듈

예시: Custom Agent, Custom MultiAction Agent, Conversation Agent,...

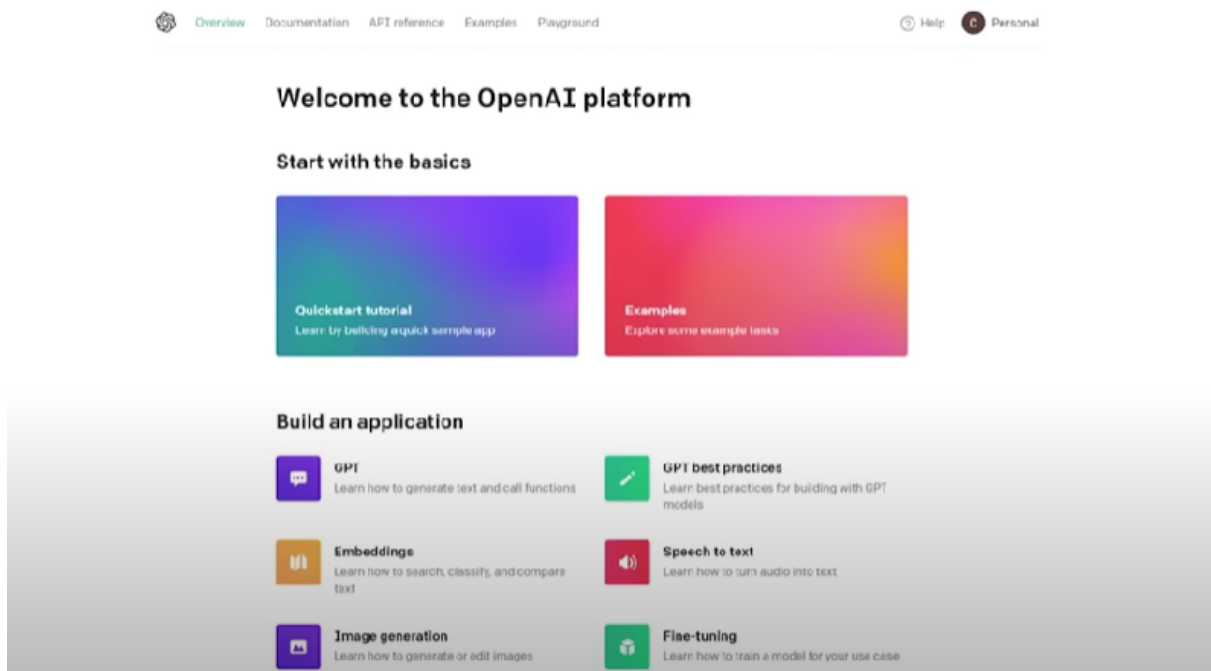
예시: PDF 챗봇 구축

문서를 기반으로 챗봇을 구축할 경우, 아래와 같은 과정을 통해 대화가 가능하도록 합니다.



## 1. OpenAPI KEY 발급

# OpenAI API Key 생성 & 카드 등록

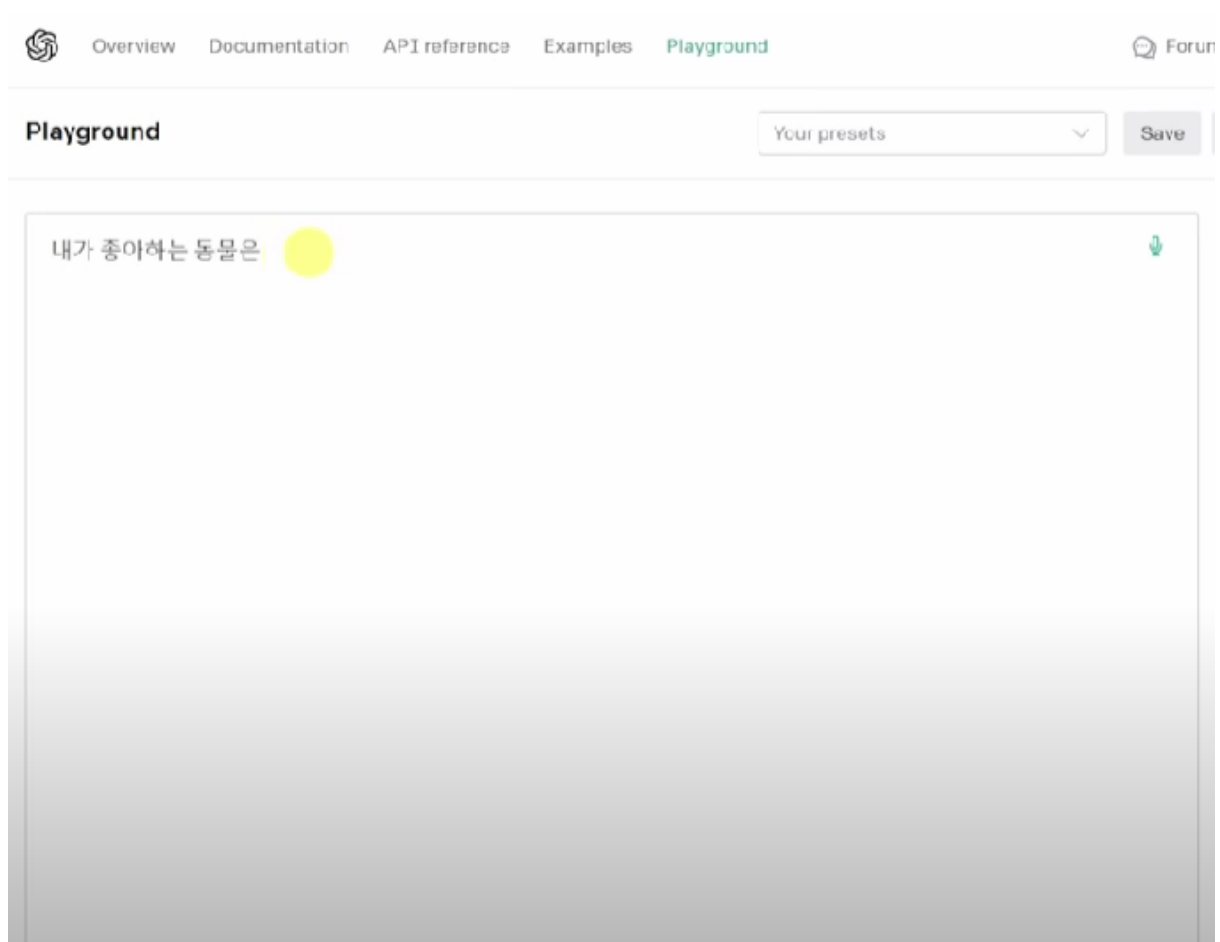


openai는เครดิต카드등록해서api호출할때마다지불이되지만,라마는무료라하니라마로사용가능하다고함.

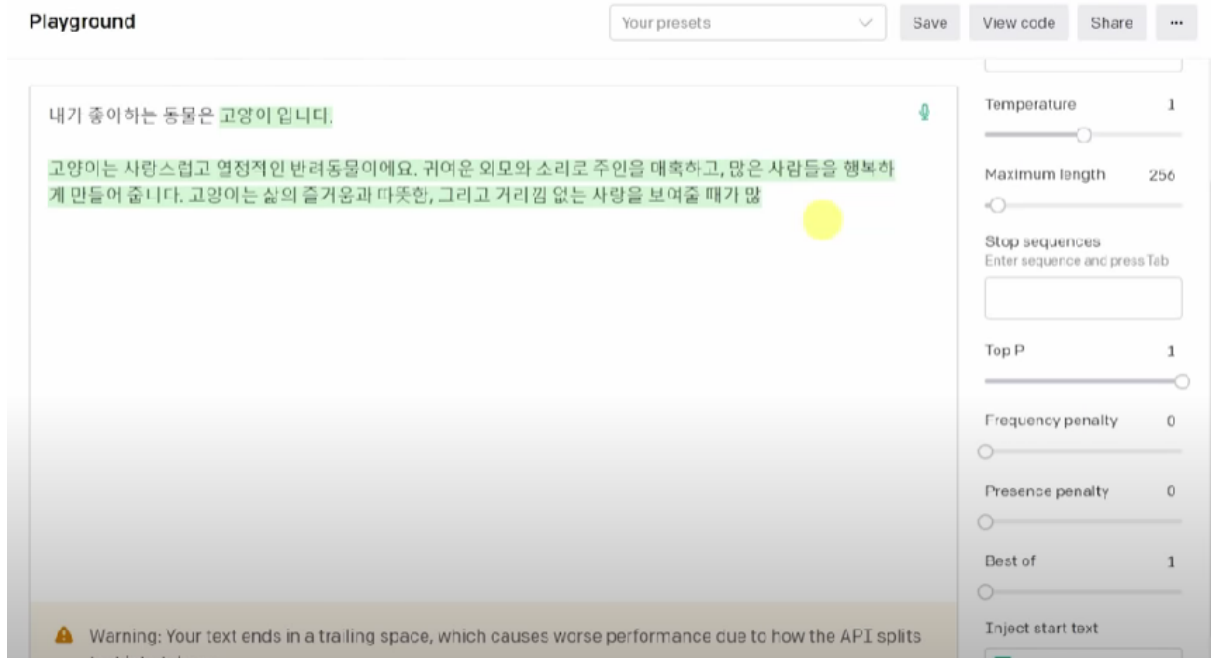
(메타에서만든LLM모델무료로상업적이용이가능합니다)

## 2. LLM model, Chat model?

### LLM



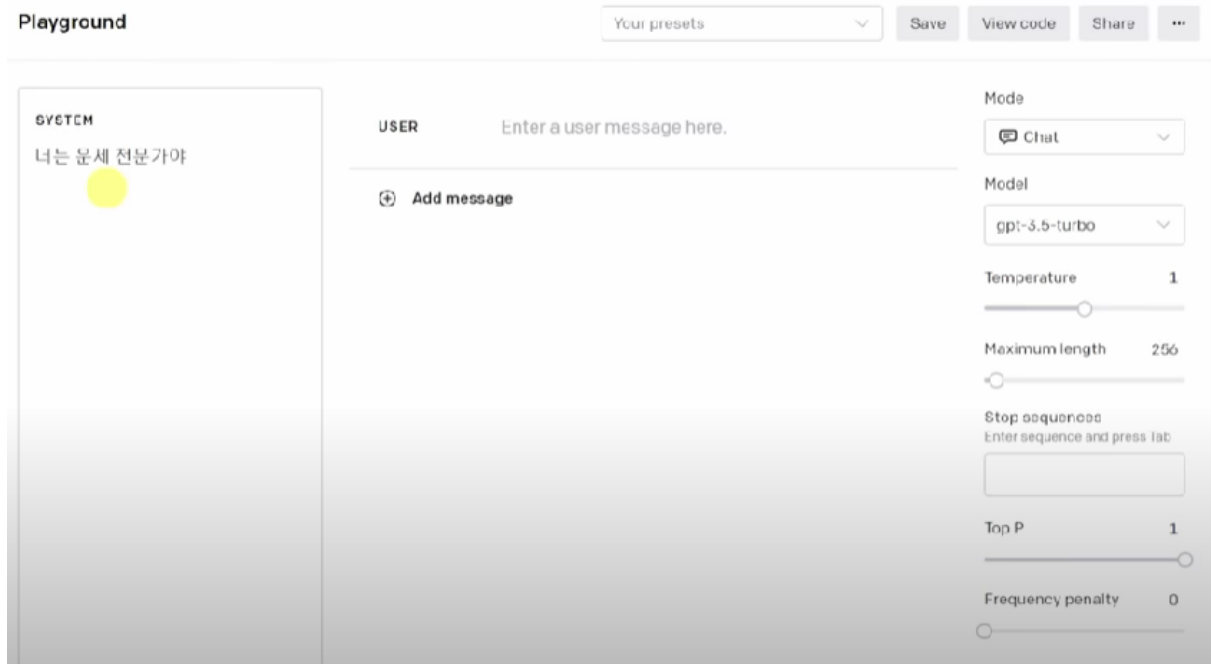
“내가 좋아하는 동물은” 만 치고 요청 시

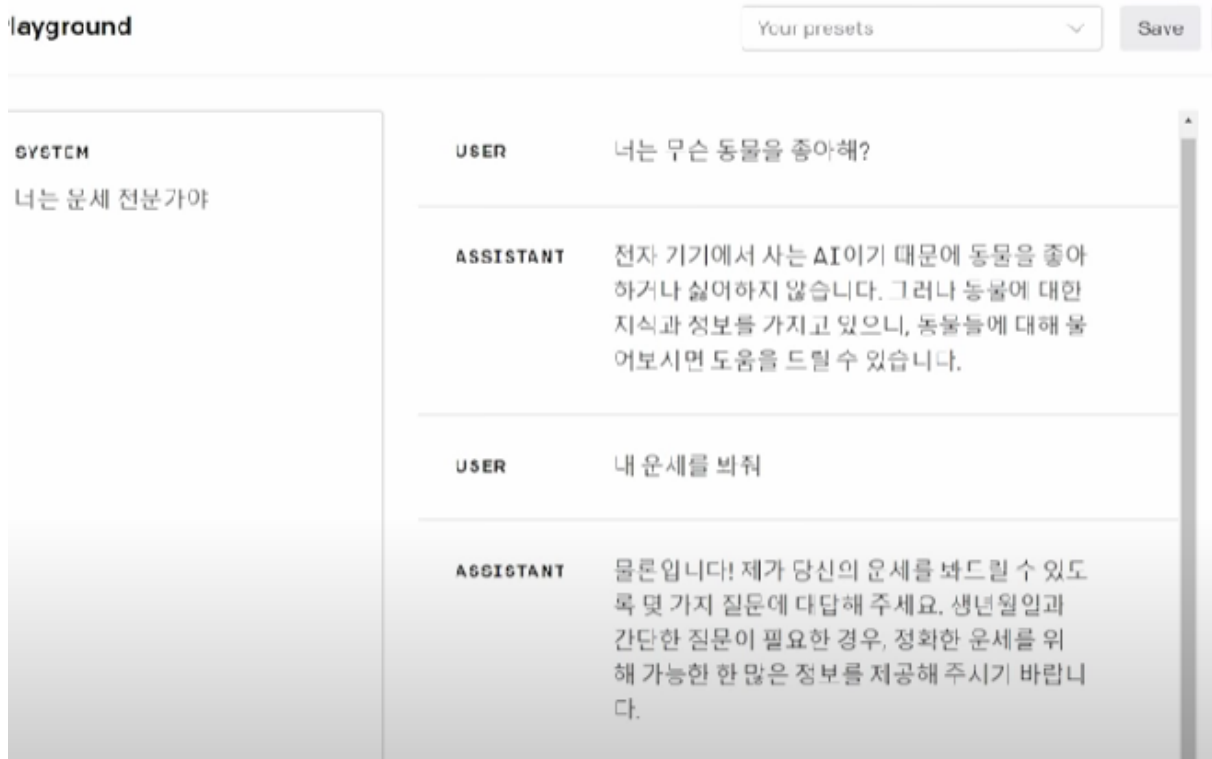


이런 한 글을 만들어주는 것을 LLM

## Chat

Chat 모델은 "역할"부여가 가능. 일반적인 챗봇을 이 모델을 사용.



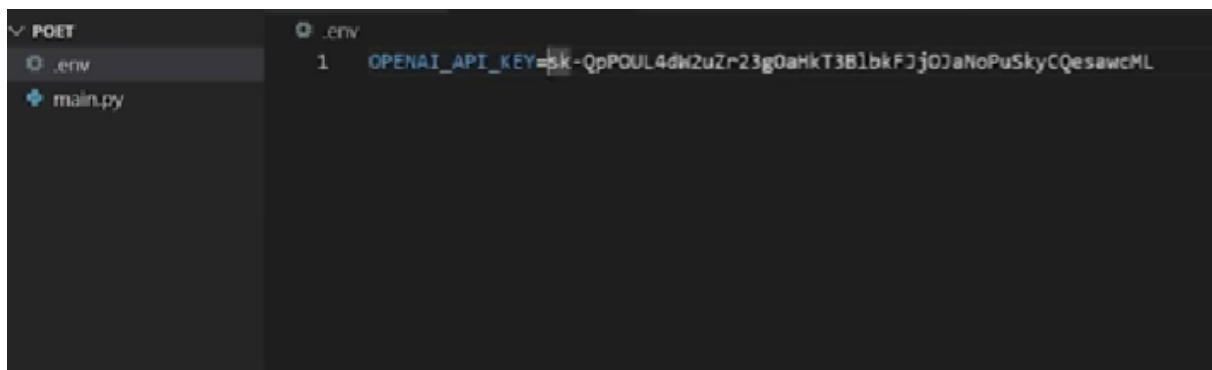


사용자와 상대 ROLE이 부여되어 대화하는 형식으로 이어나가는 모델

즉 LLM모델과 Chat모델은 사용자가 요청할 때 답변 방식에 대한 차이가 있지만, 주어진 파라미터에 대해 OpenAI의 GPT에서 제시어에 대해 뒤에 이어질 말을 확률적으로 나타내줌.

## Python을 이용한 실습

### 1. 생성한 OpenAI KEY 환경변수 설정하기



### 2. main.py에서 랭체인 간단한 실습



## a. LLM 사용

- i. .env에서 환경설정된 API\_KEY IMPORT
- ii. 설치한 OpenAI LLM IMPORT후 llm 변수에 OpenAI 대입
- iii. predict함수를 통해 질문할 컨텍스트 지정

```
main.py x requirements.txt .env
main.py > ...
1 from dotenv import load_dotenv
2 load_dotenv()
3
4 from langchain.llms import OpenAI
5 llm = OpenAI()
6 result = llm.predict("내가 좋아하는 동물은 ")
7 print(result)
8
```

```
EXPLORER
POET
.env
main.py
requirements.txt

main.py > ...
1 from dotenv import load_dotenv
2 load_dotenv()
3
4 from langchain.llms import OpenAI
5 llm = OpenAI()
6 result = llm.predict("내가 좋아하는 동물은 ")
7 print(result)
8

TERMINAL
PS C:\langchain\poet> python main.py

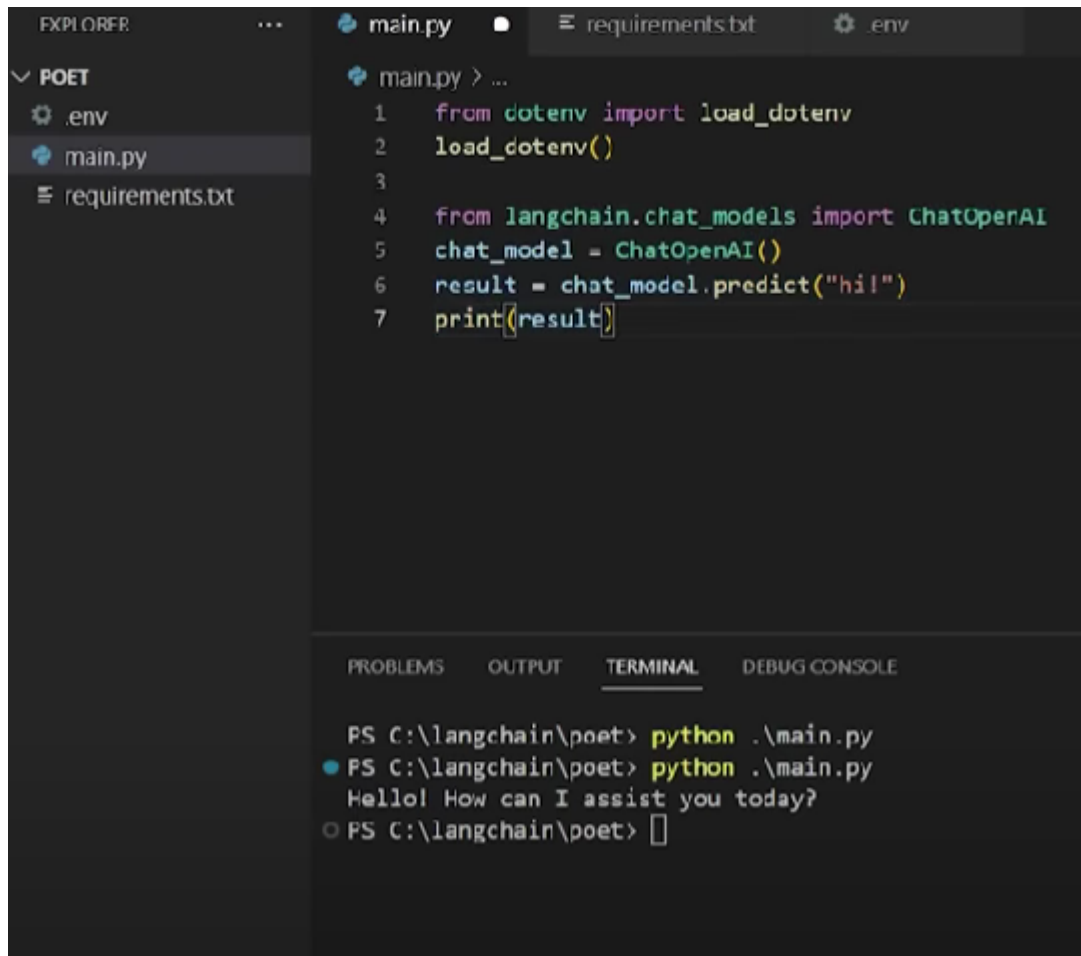
Good Morning

Good morning! How are you doing today?
PS C:\langchain\poet> python main.py
고양이입니다. 고양이는 귀엽고 사랑스러운 동물 중 하나입니다. 고양이는 사람의 연애와 친밀한 관계를 맺을 수 있으며, 애교를 잘 보여줍니다. 고양이는 또한 잘 지내며, 열과 습관을 잘 이어갈 수 있기 때문에 우리와 살기 좋
PS C:\langchain\poet>
```

## b.Chat 사용

- i. 설치한 OpenAI ChatOpenAI IMPORT후 chat\_model 변수에 ChatOpenAI 대입
- ii. predict함수를 통해 질문할 컨텍스트 지정

iii. LLM과 다르게 ChatOpenAI는 답변 후 다시 질문을 하는 형식으로 답을 줌

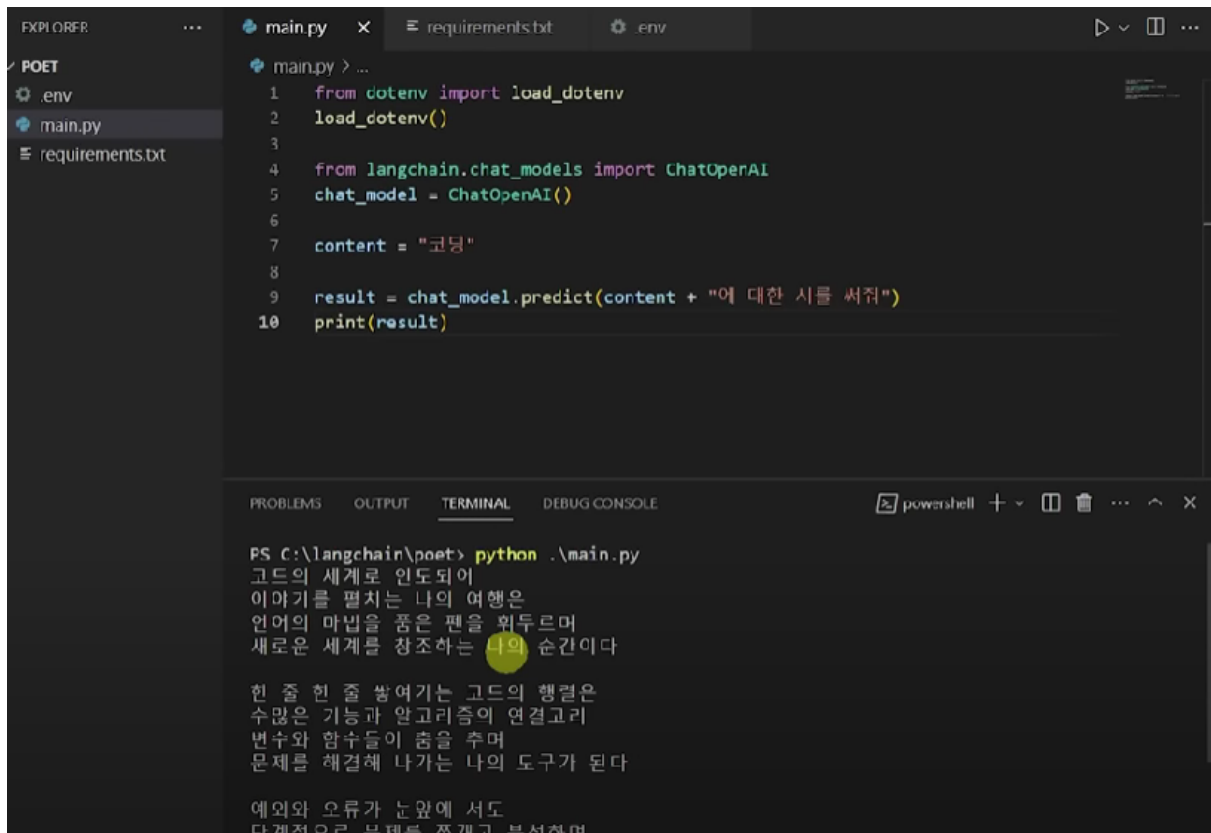


The screenshot shows a VS Code editor with a file explorer on the left and a code editor in the center. The file explorer shows a folder named 'POET' containing files '.env', 'main.py', and 'requirements.txt'. The code editor shows the content of 'main.py' with the following code:

```
1 from dotenv import load_dotenv
2 load_dotenv()
3
4 from langchain.chat_models import ChatOpenAI
5 chat_model = ChatOpenAI()
6 result = chat_model.predict("hi!")
7 print(result)
```

Below the code editor is a terminal window with the following output:

```
PS C:\langchain\poet> python .\main.py
● PS C:\langchain\poet> python .\main.py
Hello! How can I assist you today?
○ PS C:\langchain\poet> 
```



The image shows a Visual Studio Code editor window with a dark theme. The Explorer sidebar on the left shows a project named 'POET' with files 'env', 'main.py', and 'requirements.txt'. The main editor area displays 'main.py' with the following code:

```
1 from dotenv import load_dotenv
2 load_dotenv()
3
4 from langchain.chat_models import ChatOpenAI
5 chat_model = ChatOpenAI()
6
7 content = "코딩"
8
9 result = chat_model.predict(content + "에 대한 시를 써줘")
10 print(result)
```

Below the code editor is a terminal window titled 'powershell'. It shows the command 'python .\main.py' being executed, followed by the output of a poem in Korean:

```
PS C:\langchain\poet> python .\main.py
고드의 세계로 인도되어
이야기를 펼쳐는 나의 여행은
언어의 마법을 품은 펜을 휘두르며
새로운 세계를 창조하는 나의 순간이다

힌 줄 힌 줄 쌓여가는 고드의 행렬은
수많은 기능과 알고리즘의 연결고리
변수와 함수들이 춤을 추며
문제를 해결해 나가는 나의 도구가 된다

예외와 오류가 눈앞에 서도
다계적으로 문제들 주께고 분석하며
```

처음이라 랭체인에 대해 친해지기 위해 랭체인의 기본 개념 및 구조와 OpenAI의 API KEY를 통해 코드상에서 간단한 예시를 가져와서 설명했습니다.