환경변수 설정

```
from dotenv import load_dotenv
import os
load_dotenv()
```

```
PINECONE_API_KEY = os.getenv('PINECONE_API_KEY')
PINECONE_API_KEY
```

Pinecone 클라이언트 초기화

```
from pinecone import Pinecone

# Pinecone 클라이언트를 초기화합니다.
# PINECONE_API_KEY는 환경 변수에서 가져온 API 키입니다.
pc = Pinecone(api_key=PINECONE_API_KEY)
```

```
# Pinecone 클라이언트를 사용하여 현재 사용 가능한 모든 인덱스의 목록을 반환합니다.
pc.list_indexes()
```

```
# 사용할 인덱스 이름을 지정합니다.
index_name = "wiki"

# 지정한 이름으로 Pinecone 인덱스를 가져옵니다.
index = pc.Index(index_name)

# 인덱스의 통계 정보를 설명합니다.
index.describe_index_stats()
```

```
{'dimension': 1536,
 'index_fullness': 0.0,
 'metric': 'cosine',
 'namespaces': {'wiki-ns1': {'vector_count': 1080}},
 'total_vector_count': 1080,
 'vector_type': 'dense'}
```

검색하기 (Pinecone)

```
from langchain_openai import OpenAIEmbeddings

# OpenAIEmbeddings 객체를 초기화합니다.
# 모델은 "text-embedding-3-small"을 사용하고, API 키는 OPENAI_API_KEY를 사용합니다.
embedding = OpenAIEmbeddings(model="text-embedding-3-small")

# 질문 리스트를 정의합니다.
question = ["U.S television channel G4"]

# 질문을 임베딩하여 벡터로 변환합니다.
embedded_question = embedding.embed_documents(question)

# 임베딩된 질문 벡터를 출력합니다.
print(embedded_question)
```

```
[[0.038361743092536926, 0.0569768100976944, 0.003173826029524207, 0.026196883991360664, -0.04045523330569267, -0.008423457853496075, -0.015899188816
```

```
# Pinecone 인덱스에서 쿼리를 실행합니다.
query_result = index.query(
    namespace="wiki-ns1",  # wiki-ns1 네임스페이스를 지정합니다.
    vector=embedded_question,  # 임베딩된 질문 벡터를 사용합니다.
    top_k=5,  # 상위 5개의 결과를 반환합니다.
    include_vector=False,  # 벡터를 포함하지 않습니다.
    include_metadata=True  # 메타데이터를 포함합니다.
)

# 쿼리 결과에서 각 매치의 ID를 추출하여 리스트로 만듭니다.
result_ids = [r.id for r in query_result.matches]
print(result_ids)  # 결과 ID를 출력합니다.
```

```
['190176a4-d1bc-48b3-aadf-9bd306c43f67', '2565ee2c-6350-450f-a3c8-0ddacab33d37', '801d7987-f93e-412f-9b74-0b71ecef113f', '837ae324-d70d-4438-b425-f7
```

```
# 쿼리 결과에서 각 매치의 점수와 메타데이터를 출력합니다.
for r in query_result.matches:
```

```
    print(r.score, r.metadata)
```

```
0.660370886 {'chunk_id': 0.0, 'head_chunk': 'This is a list of television programs formerly broadcast by the U.S. cable television channel G4.₩n₩nSe
0.556059837 {'chunk_id': 9.0, 'head_chunk': 'Voyagers! (2013)₩n₩nProgramming blocks ₩n Expansion Pack (2002-2003)₩n G4 Global Gamer (2003-2004)₩n Rea
0.543539107 {'chunk_id': 14.0, 'head_chunk': 'Morgan Minute - Hosted by Morgan Webb, this segment shows the latest games designed specifically for w
0.538311 {'chunk_id': 4.0, 'head_chunk': "G4's Late Night Peepshow (2005-2006)₩n G4's Training Camp (2005-2006)₩n Street Fury (2005-2006)₩n Video Gai
0.527693748 {'chunk_id': 6.0, 'head_chunk': "G4's Proving Ground (2011; 2014; reruns)₩n Jump City: Seattle (2011)₩n Top 100 Video Games of All Time
```

검색하기 (LangChain)

```
from langchain_pinecone import PineconeVectorStore

# PineconeVectorStore 객체를 초기화합니다.
# 인덱스는 이전에 정의된 index 객체를 사용하고, 임베딩은 embedding 객체를 사용합니다.
# 텍스트 키는 "full_text"로 설정합니다.
vector_store = PineconeVectorStore(index=index, embedding=embedding, text_key="head_chunk")

# 질문에 대한 유사도 검색을 수행합니다.
# query는 question 리스트의 첫 번째 요소를 사용하고, 상위 5개의 결과를 반환합니다.
docs = vector_store.similarity_search(query=question[0], k=5, namespace="wiki-ns1")
print(docs)

# 검색된 문서의 메타데이터를 출력합니다.
for doc in docs:
    print(doc.metadata)
```

```
[Document(id='190176a4-d1bc-48b3-aadf-9bd306c43f67', metadata={'chunk_id': 0.0, 'title': 'List of programs broadcast by G4', 'url': 'https://en.wiki
{'chunk_id': 0.0, 'title': 'List of programs broadcast by G4', 'url': 'https://en.wikipedia.org/wiki/List%20of%20programs%20broadcast%20by%20G4', 'w
{'chunk_id': 9.0, 'title': 'List of programs broadcast by G4', 'url': 'https://en.wikipedia.org/wiki/List%20of%20programs%20broadcast%20by%20G4', 'w
{'chunk_id': 14.0, 'title': 'List of programs broadcast by G4', 'url': 'https://en.wikipedia.org/wiki/List%20of%20programs%20broadcast%20by%20G4', '
{'chunk_id': 4.0, 'title': 'List of programs broadcast by G4', 'url': 'https://en.wikipedia.org/wiki/List%20of%20programs%20broadcast%20by%20G4', 'w
{'chunk_id': 6.0, 'title': 'List of programs broadcast by G4', 'url': 'https://en.wikipedia.org/wiki/List%20of%20programs%20broadcast%20by%20G4', 'w
```