

Assistant setup X

System message Add your data

Specify how the chat should act

Use a template to get started, or just start writing your own system message below. Want some tips? [Learn more](#)

Use a system message template

Select a template ▼

System message (1)

You are an AI assistant that helps people find information. (ADD HERE) 

잠재적 사용 패턴

역할 정의

도우미를 원하는 역할을 정의할 수 있습니다. 예를 들어 지원 봇을 빌드하는 경우 "사용자가 새로운 문제를 해결하는 데 도움이 되는 전문가 인시던트 지원 도우미입니다."를 추가 할 수 있습니다.

검색할 데이터 형식 정의

도우미에 제공하는 데이터의 특성을 추가할 수도 있습니다.

- "재무 보고서", "학술지" 또는 "인시던트 보고서"와 같은 데이터 세트의 주제 또는 범위를 정의합니다. 예를 들어 기술 지원을 위해 "검색된 문서에서 유사한 인시던트 정보를 사용하여 쿼리에 응답합니다."를 추가할 수 있습니다.
- 데이터에 특정 특성이 있는 경우 시스템 메시지에 이러한 세부 정보를 추가할 수 있습니다. 예를 들어 문서가 일본어인 경우 "일본어 문서를 검색하고 일본어로 주의 깊게 읽고 일본어로 대답해야 합니다."를 추가할 수 있습니다.
- 문서에 재무 보고서의 테이블과 같은 구조화된 데이터가 포함된 경우 시스템 프롬프트에 이 팩트를 추가할 수도 있습니다. 예를 들어 데이터에 테이블이 있는 경우 "재무 결과와 관련된 테이블 형식의 데이터가 제공되고 사용자 질문에 대답하기 위해 계산을 수행하기 위해 테이블 줄을 읽어야 합니다."를 추가할 수 있습니다.

출력 스타일 정의

시스템 메시지를 정의하여 모델의 출력을 변경할 수도 있습니다. 예를 들어 도우미 답변이 프랑스어로 되어 있는지 확인하려면 다음과 같은 프롬프트를 추가할 수 있습니다. "프랑스어를 이해하는 사용자가 정보를 찾는 데 도움이 되는 AI 도우미입니다. 사용자 질문

은 영어 또는 프랑스어로 할 수 있습니다. 검색된 문서를 주의 깊게 읽고 프랑스어로 답변하세요. 모든 답변이 프랑스어로 표시되도록 문서에서 프랑스어로 지식을 번역하세요."

중요한 동작 재확인

Azure OpenAI On Your Data는 데이터를 사용하여 사용자 쿼리에 응답하는 프롬프트 형식으로 큰 언어 모델에 지침을 전송하여 작동합니다. 애플리케이션에 중요한 특정 동작이 있는 경우 시스템 메시지에서 동작을 반복하여 정확도를 높일 수 있습니다. 예를 들어 문서에서만 답변하도록 모델을 안내하려면 다음을 추가할 수 있습니다. "정보를 사용하지 않고 검색된 문서만 사용하여 답변하세요. 답변의 모든 클레임에 대해 검색된 문서에 대한 인용을 생성하세요. 검색된 문서를 사용하여 사용자 질문에 대답할 수 없는 경우 문서가 사용자 쿼리와 관련된 이유를 설명하세요. 어떤 경우에도 자신의 지식을 사용하여 대답하지 마십시오".

프롬프트 엔지니어링 트릭

프롬프트 엔지니어링에는 출력을 개선하기 위해 시도할 수 있는 많은 트릭이 있습니다. 한 가지 예는 다음을 추가할 수 있는 생각 체인 프롬프트입니다. "사용자 쿼리에 응답하기 위해 검색된 문서의 정보에 대해 단계별로 생각해 봅시다. 문서에서 사용자 쿼리에 대한 관련 지식을 단계별로 추출하고 관련 문서에서 추출된 정보에서 맨 아래까지 답변을 작성합니다".

① 참고

시스템 메시지는 GPT 도우미가 검색된 문서에 따라 사용자 질문에 응답하는 방법을 수정하는 데 사용됩니다. 검색 프로세스에는 영향을 주지 않습니다. 검색 프로세스에 대한 지침을 제공하려는 경우 질문에 포함하는 것이 좋습니다. 시스템 메시지는 단지 지침일 뿐입니다. 모델은 객관성과 논란의 여지가 있는 진술을 피하는 등의 특정 동작으로 준비되어 있기 때문에 지정된 모든 지침을 준수하지 않을 수 있습니다. 시스템 메시지가 이러한 동작과 모순되는 경우 예기치 않은 동작이 발생할 수 있습니다.

최대 응답

모델 응답당 토큰 수 한도를 설정합니다. 데이터에 대한 Azure OpenAI의 상한은 1500입니다. 이는 API에서 `max_tokens` 매개 변수를 설정하는 것과 같습니다.

데이터에 대한 응답 제한

이 옵션은 모델이 데이터만 사용하여 응답하도록 권장하며 기본적으로 선택됩니다. 이 옵션을 선택 취소하면 모델이 내부 지식을 더 쉽게 적용하여 응답할 수 있습니다. 사용 사례

와 시나리오에 따라 올바른 선택을 결정합니다.

모델과의 상호 작용

모델과 대화할 때 최상의 결과를 얻으려면 다음 방법을 따릅니다.

대화 기록

- 새 대화를 시작하거나 이전 대화와 관련이 없는 질문을 하기 전에 채팅 기록을 지웁니다.
- 대화 기록이 모델의 현재 상태를 변경하기 때문에 첫 번째 대화 차례와 후속 차례 사이에 동일한 질문에 대해 서로 다른 응답을 가져올 것으로 예상할 수 있습니다. 잘못된 답변을 받은 경우 품질 버그로 신고해 주세요.

모델 응답

- 특정 질문에 대한 모델 응답에 만족하지 않는 경우 질문을 보다 구체적이거나 더 일반적인 것으로 만들어 모델이 응답하는 방식을 확인하고 그에 따라 질문을 재구성해 보세요.
- [CoT\(Chain-of-thought\) 프롬프팅](#)은 모델이 복잡한 질문/작업에 대해 원하는 결과를 생성하도록 하는 데 효과적인 것으로 나타났습니다.

질문 길이

긴 질문은 피하고 가능하면 여러 질문으로 나누세요. GPT 모델에는 허용할 수 있는 토큰 수에 제한이 있습니다. 토큰 제한은 사용자 질문, 시스템 메시지, 검색된 검색 문서(청크), 내부 프롬프트, 대화 기록(있는 경우) 및 응답에 계산됩니다. 질문이 토큰 제한을 초과하면 잘립니다.

다국어 지원

- 현재 Azure OpenAI On Your Data 지원 쿼리의 키워드 검색 및 의미 체계 검색은 인덱스의 데이터와 동일한 언어로 제공됩니다. 예를 들어, 데이터가 일본어로 되어 있으면 입력 쿼리도 일본어로 되어 있어야 합니다. 언어 간 문서 검색의 경우 [벡터 검색](#)을 사용하도록 설정한 인덱스를 빌드하는 것이 좋습니다.
- 정보 검색 및 모델 응답의 품질을 향상시키려면 영어, 프랑스어, 스페인어, 포르투갈어, 이탈리아어, 독일, 중국어(Zh), 일본어, 한국어, 러시아어, 아랍어 언어에 대한 [의미 체계 검색](#)을 사용하도록 설정하는 것이 좋습니다.
- 데이터가 다른 언어로 되어 있음을 모델에 알리려면 시스템 메시지를 사용하는 것이 좋습니다. 예시:

- *** 사용자가 검색된 일본어 문서에서 정보를 추출할 수 있도록 디자인된 AI 도우미입니다. 응답을 작성하기 전에 일본어 문서를 주의해서 조사하세요. 사용자의 쿼리는 일본어로 되어 있으며 일본어로도 응답해야 합니다."
- 여러 언어로 된 문서가 있는 경우 각 언어에 대한 새 인덱스를 빌드하고 별도로 Azure OpenAI에 연결하는 것이 좋습니다.

스트리밍 데이터

`stream` 매개 변수를 사용하여 스트리밍 요청을 보내면 전체 API 응답을 기다리지 않고도 데이터를 점진적으로 보내고 받을 수 있습니다. 이는 특히 대규모 또는 동적 데이터의 경우 성능과 사용자 환경을 개선시킬 수 있습니다.

```
JSON

{
  "stream": true,
  "dataSources": [
    {
      "type": "AzureCognitiveSearch",
      "parameters": {
        "endpoint": "'$AZURE_AI_SEARCH_ENDPOINT'",
        "key": "'$AZURE_AI_SEARCH_API_KEY'",
        "indexName": "'$AZURE_AI_SEARCH_INDEX'"
      }
    }
  ],
  "messages": [
    {
      "role": "user",
      "content": "What are the differences between Azure Machine Learning and Azure AI services?"
    }
  ]
}
```

더 나은 결과를 위한 대화 기록

모델과 채팅할 때 채팅 기록을 제공하면 모델이 더 높은 품질의 결과를 반환하는 데 도움이 됩니다. 더 나은 응답 품질을 위해 API 요청에 도우미 메시지의 `context` 속성을 포함할 필요가 없습니다. [예제는 API 참조 설명서](#)를 참조하세요.

함수 호출

일부 Azure OpenAI 모델을 사용하면 함수 호출을 사용하도록 [도구 및 tool_choice 매개 변수](#)를 정의할 수 있습니다. REST API `/chat/completions` 을 통해 함수 호출을 설정할 수 있습니다. `tools` 및 [데이터 원본](#)이 모두 요청에 있는 경우 다음 정책이 적용됩니다.

1. `tool_choice` 이(가) `none` 인 경우, 도구가 무시되고 데이터 원본만 사용하여 답변을 생성합니다.
2. 그렇지 않으면 `tool_choice` 지정되지 않았거나 `auto` 또는 개체로 지정되면 데이터 원본이 무시되고 응답에 선택한 함수 이름과 인수(있는 경우)가 포함됩니다. 모델이 함수를 선택하지 않는다고 결정하더라도 데이터 원본은 여전히 무시됩니다.

위의 정책이 요구 사항을 충족하지 않는 경우 [프롬프트 흐름](#) 또는 [Assistants API](#) 같은 다른 옵션을 고려하세요.

데이터의 Azure OpenAI에 대한 토큰 사용량 예측

Azure OpenAI On Your Data Retrieval Augmented Generation(RAG) 서비스는 검색 서비스(예: Azure AI Search) 및 생성(Azure OpenAI 모델)을 모두 활용하여 사용자가 제공된 데이터를 기반으로 질문에 대한 답변을 얻을 수 있도록 합니다.

이 RAG 파이프라인의 일부로 상위 수준에서 다음 세 단계가 있습니다.

1. 사용자 쿼리를 검색 의도 목록으로 다시 포맷합니다. 이 작업은 지침, 사용자 질문 및 대화 기록을 포함하는 프롬프트를 사용하여 모델을 호출하여 수행됩니다. [의도 프롬프트](#)를 호출해 보겠습니다.
2. 각 의도에 대해 검색 서비스에서 여러 문서 청크가 검색됩니다. 사용자가 지정한 엄격한 임계값을 기준으로 관련 없는 청크를 필터링하고 내부 논리에 따라 청크를 다시 생성/집계한 후 사용자가 지정한 문서 청크 수가 선택됩니다.
3. 이러한 문서 청크는 사용자 질문, 대화 기록, 역할 정보 및 지침과 함께 최종 모델 응답을 생성하기 위해 모델로 전송됩니다. 이를 [생성 프롬프트](#)를 호출해 보겠습니다.

모델에 대한 호출은 총 두 가지입니다.

- 의도 처리: [의도 프롬프트](#)에 대한 토큰 예측에는 사용자 질문, 대화 기록 및 의도 생성을 위해 모델로 전송된 지침이 포함됩니다.
- 응답을 생성하기 위해: [생성 프롬프트](#)에 대한 토큰 예측에는 사용자 질문, 대화 기록, 검색된 문서 청크 목록, 역할 정보 및 생성을 위해 전송된 지침이 포함됩니다.

모델에서 생성된 출력 토큰(의도 및 응답 모두)은 총 토큰 추정을 고려해야 합니다. 아래의 4개 열을 모두 합산하면 응답을 생성하는데 사용되는 평균 총 토큰이 제공됩니다.

□ 테이블 확장

모델	생성 프롬프트 토큰 수	의도 프롬프트 토큰 수	응답 토큰 수	의도 토큰 수
gpt-35-turbo-16k	4297	1366	111	25
gpt-4-0613	3997	1385	118	18
gpt-4-1106-preview	4538	811	119	27
gpt-35-turbo-1106	4854	1372	110	26

위의 숫자는 다음을 사용하는 데이터 집합에 대한 테스트를 기반으로 합니다.

- 191개 대화
- 250개 질문
- 질문당 평균 토큰 10개
- 대화당 평균 4회 대화 턴

및 다음 **매개 변수**.

□ 테이블 확장

설정	값
검색된 문서 수	5
엄격성	3
청크 크기	1024
수집된 데이터에 대한 응답을 제한하시겠습니까?	True

이러한 예상치는 위의 매개 변수에 대해 설정된 값에 따라 달라집니다. 예를 들어 검색된 문서 수가 10으로 설정되고 엄격도가 1로 설정된 경우 토큰 수가 늘어나게 됩니다. 반환된 응답이 수집된 데이터로 제한되지 않는 경우 모델에 지정된 지침이 적고 토큰 수가 감소합니다.

추정치는 또한 질문되는 문서 및 질문의 성격에 따라 달라집니다. 예를 들어 질문이 개방형인 경우 응답이 더 길어질 수 있습니다. 마찬가지로 시스템 메시지가 길면 더 많은 토큰을 사용하는 더 긴 프롬프트가 발생하며, 대화 기록이 길면 프롬프트가 더 길어집니다.

□ 테이블 확장

모델	시스템 메시지에 대한 최대 토큰	모델 응답에 대한 최대 토큰
GPT-35-0301	400	1500
GPT-35-0613-16K	1000	3200
GPT-4-0613-8K	400	1500
GPT-4-0613-32K	2000	6400

위의 표에서는 [시스템 메시지](#) 및 모델 응답에 사용할 수 있는 최대 토큰 수를 보여 줍니다. 또한 다음 항목도 토큰을 소비합니다.

- **메타 프롬프트:** 모델의 응답을 (API `inScope=True`에서) 접지 데이터 콘텐츠로 제한하는 경우 최대 토큰 수가 더 높습니다. 그렇지 않으면(예: `inScope=False`인 경우) 최대값이 낮습니다. 이 숫자는 사용자 질문의 토큰 길이와 대화 내역에 따라 달라질 수 있습니다. 이 예측에는 검색을 위한 기본 프롬프트 및 쿼리 다시 쓰기 프롬프트가 포함됩니다.
- **사용자 질문 및 기록:** 변수이지만 2,000개의 토큰으로 제한됩니다.
- **검색된 문서(청크):** 검색된 문서 청크에 사용되는 토큰의 수는 여러 요인에 따라 달라집니다. 이 값의 상한은 검색된 문서 청크의 수에 청크 크기를 곱한 값입니다. 그러나 나머지 필드를 계산한 후 사용 중인 특정 모델에 사용할 수 있는 토큰을 기준으로 잘립니다.

사용 가능한 토큰의 20%는 모델 응답을 위해 예약되어 있습니다. 나머지 80%의 사용 가능한 토큰에는 메타 프롬프트, 사용자 질문 및 대화 내역, 시스템 메시지가 포함됩니다. 남은 토큰 예산은 검색된 문서 청크에 사용됩니다.

입력에서 사용하는 토큰 수(예: 질문, 시스템 메시지/역할 정보)를 계산하려면 다음 코드 샘플을 사용합니다.

Python

```
import tiktoken

class TokenEstimator(object):

    GPT2_TOKENIZER = tiktoken.get_encoding("gpt2")

    def estimate_tokens(self, text: str) -> int:
        return len(self.GPT2_TOKENIZER.encode(text))

token_output = TokenEstimator.estimate_tokens(input_text)
```

문제 해결

실패한 작업을 해결하려면 항상 API 응답 또는 Azure OpenAI 스튜디오에 지정된 오류 또는 경고를 확인합니다. 다음은 몇 가지 일반적인 오류 및 경고입니다.

실패한 수집 작업

할당량 한도 문제

서비스 Y에서 X라는 이름의 인덱스만 만들 수 없습니다. 이 서비스에 대한 인덱스 할당량이 초과되었습니다. 사용하지 않는 인덱스를 먼저 삭제하거나 인덱스 생성 요청 사이에 지연 시간을 추가하거나 더 높은 한도를 위해 서비스를 업그레이드해야 합니다.

이 서비스에 대한 표준 인덱서 할당량 X를 초과했습니다. 현재 X의 표준 인덱서가 있습니다. 사용하지 않는 인덱서를 먼저 삭제하거나 인덱서 '실행 모드'를 변경하거나 서비스를 업그레이드하여 더 높은 한도를 적용해야 합니다.

해결 방법:

더 높은 가격 책정 계층으로 업그레이드하거나 사용하지 않는 자산을 삭제하세요.

전처리 시간 초과 문제

웹 API 요청이 실패했으므로 기술을 실행할 수 없음

Web API 기술 응답이 옮바르지 않기 때문에 기술을 실행할 수 없습니다.

해결 방법:

입력 문서를 더 작은 문서로 분해하고 다시 시도합니다.

권한 문제

이 요청은 이 작업을 수행할 권한이 없습니다

해결 방법:

즉, 지정된 자격 증명을 사용하여 스토리지 계정에 액세스할 수 없습니다. 이 경우 API에 전달된 스토리지 계정 자격 증명을 검토하고 프라이빗 엔드포인트 뒤에 스토리지 계정이 숨겨지지 않았는지 확인합니다(프라이빗 엔드포인트가 이 리소스에 대해 구성되지 않은 경우).

Azure AI Search를 사용하여 쿼리를 보낼 때 발생하는 503 오류

각 사용자 메시지는 여러 검색 쿼리로 변환할 수 있으며, 모두 검색 리소스로 병렬로 전송됩니다. 이렇게 하면 검색 복제본 및 파티션의 양이 낮을 때 제한 동작이 생성될 수 있습니다. 단일 파티션 및 단일 복제본이 지원할 수 있는 초당 최대 쿼리 수는 충분하지 않을 수 있습니다. 이 경우 복제본 및 파티션을 늘리거나 애플리케이션에서 절전/재시도 논리를 추가하는 것이 좋습니다. 자세한 내용은 [Azure AI Search 설명서](#)를 참조하세요.

지역별 가용성 및 모델 지원

다음 지역에서 Azure OpenAI 리소스와 함께 Azure OpenAI On Your Data를 사용할 수 있습니다.

- 오스트레일리아 동부
- 브라질 남부
- 캐나다 동부
- 미국 동부
- 미국 동부 2
- 프랑스 중부
- 일본 동부
- 미국 중북부
- 노르웨이 동부
- 남아프리카 북부
- 미국 중남부
- 인도 남부
- 스웨덴 중부
- 스위스 북부
- 영국 남부
- 서유럽
- 미국 서부

지원되는 모델

- gpt-4 (0314)
- gpt-4 (0613)
- gpt-4-32k (0314)
- gpt-4-32k (0613)
- gpt-4 (1106-preview)
- gpt-35-turbo-16k (0613)
- gpt-35-turbo (1106)

Azure OpenAI 리소스가 다른 지역에 있는 경우 Azure OpenAI On Your Data를 사용할 수 없습니다.

다음 단계

- Azure OpenAI로 데이터 사용 시작
- 데이터에 대한 Azure OpenAI를 안전하게 사용
- 프롬프트 엔지니어링 소개

Azure OpenAI 스튜디오의 GPT-4 Turbo with Vision(미리 보기)을 사용하는 이미지 포함 데이터를 사용하는 Azure OpenAI

아티클 • 2024. 03. 12.

이 문서를 사용하여 Azure OpenAI의 비전 모델인 GPT-4 Turbo with Vision에 대한 고유한 이미지 데이터를 제공하는 방법을 알아봅니다. 데이터에 대한 GPT-4 Turbo with Vision을 사용하면 모델은 사용자 고유의 이미지 및 이미지 메타데이터를 기반으로 검색 증강 생성을 사용하여 더 많은 사용자 지정 응답과 대상 응답을 생성할 수 있습니다.

ⓘ 중요

이 문서는 GPT-4 Turbo with Vision 모델에서 데이터를 사용하는 내용을 다룹니다. 텍스트 기반 모델에 데이터를 사용하는 내용을 보려면 [텍스트 데이터 사용](#)을 참조하세요.

필수 조건

- Azure 구독 – [체험 구독을 만듭니다](#).
- 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한.
현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다.
<https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 엽니다.
- GPT-4 Turbo with Vision 모델이 배포된 Azure OpenAI 리소스. 모델 배포에 대한 자세한 내용은 [리소스 배포 가이드](#)를 참조하세요.
- 적어도 Azure OpenAI 리소스에 대한 [Cognitive Services 기여자 역할](#)이 할당되어야 합니다.

데이터 원본 추가

[Azure OpenAI Studio](#)로 이동한 다음, Azure OpenAI 리소스에 액세스할 수 있는 자격 증명으로 로그인합니다. 로그인 워크플로 도중 또는 이후에 적절한 디렉터리, Azure 구독 및 Azure OpenAI 리소스를 선택합니다.

Azure AI | Azure AI Studio

Azure AI Studio

Welcome to Azure OpenAI service

Explore the generative AI models, craft unique prompts for your use cases, and fine-tune select models.

Get started

Chat playground

Design a customized AI assistant using ChatGPT. Experiment with GPT-3.5-Turbo and GPT-4 models.

Try it now

Completions playground

Experiment with completions models for use cases such as summarization, content generation, and classification.

Try it now

DALL-E playground PREVIEW

Generate unique images by writing descriptions in natural language.

Try it now

도우미 설정 타일에서 데이터 추가(미리 보기)>+ 데이터 원본 추가를 선택합니다.

Azure AI | Azure OpenAI Studio > Chat playground

Chat playground

Assistant setup

+ Add a data source

System message Add your data (preview)

Ask questions about your own data. Your data is stored securely in your Azure subscription. [Learn more about how your data is protected.](#)

Chat session

Start chatting

The chat playground can now see, hear, and speak. Select the microphone in the chat window and start speaking to prompt the model without manually entering text. You can also hear the model's output by selecting the speaker icon.

Type user query here. (Shift + Enter for new line)

Configuration

Deployment Parameters

Deployment * your GPT-4 Turbo with Vision model

Enhancements

Vision Azure AI Services

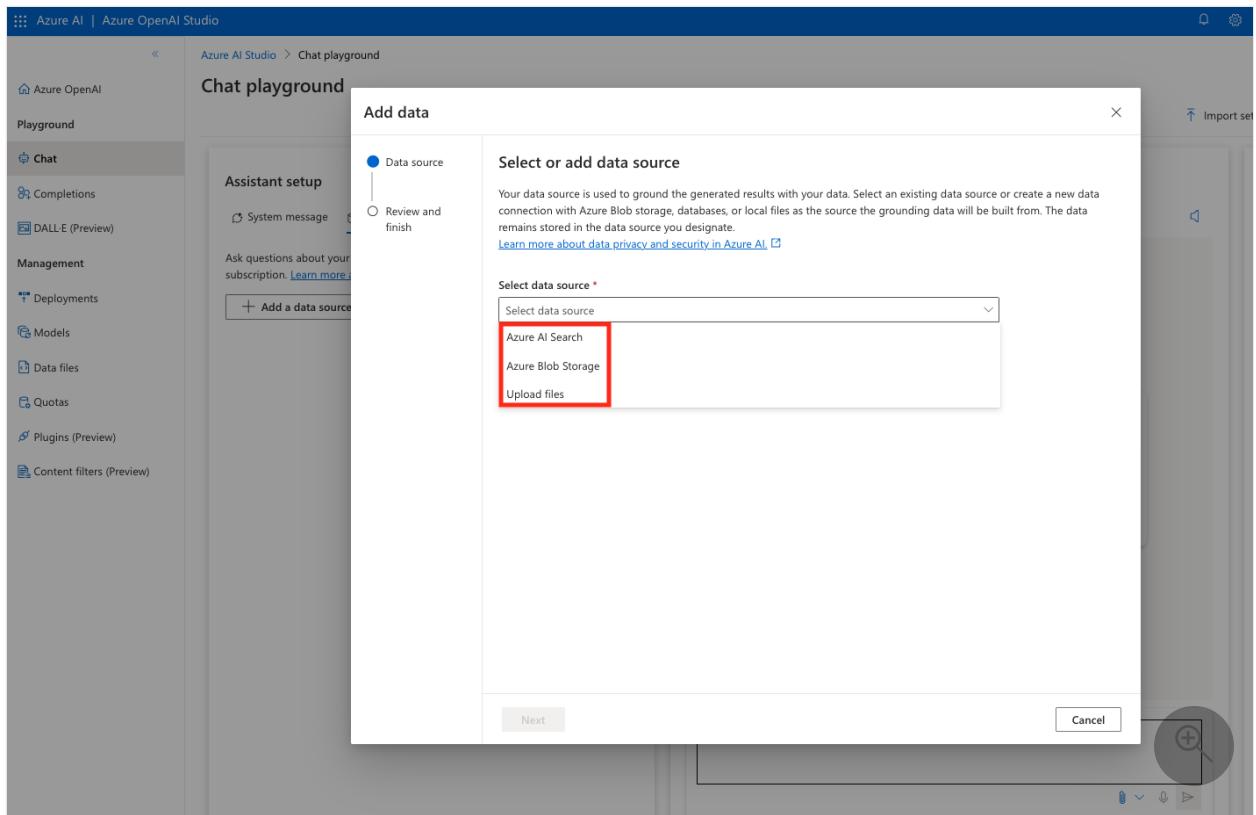
Session settings

Past messages included 10

Current token count

Input tokens progress indicator 11/128000

데이터 원본 추가를 선택하면 나타나는 창에는 데이터 원본을 선택할 수 있는 여러 옵션이 있습니다.



GPT-4 Turbo with Vision의 데이터 원본에 데이터를 추가하는 세 가지 옵션이 있습니다.

- 사용자 고유의 이미지 파일 및 이미지 메타데이터 사용
- Azure AI 검색 사용
- Azure Blob Storage 사용

세 가지 옵션은 Azure AI 검색 인덱스를 사용하여 이미지 간 검색을 수행하고 입력 프롬프트 이미지에 대한 상위 검색 결과를 검색합니다. Azure Blob Storage 및 파일 업로드 옵션의 경우 Azure OpenAI가 자동으로 이미지 검색 인덱스를 생성합니다. Azure AI 검색의 경우 이미지 검색 인덱스가 있어야 합니다. 다음 섹션에서는 검색 인덱스를 만드는 방법을 자세히 설명합니다.

이 세 가지 옵션을 처음으로 사용하는 경우 CORS(원본 간 리소스 공유)를 켜라는 내용의 빨간색 알림이 표시될 수 있습니다. 이 알림은 CORS를 사용하도록 설정할 것을 요청하는 알림입니다. 그래야만 Azure OpenAI가 Blob Storage 계정에 액세스할 수 있습니다. 경고를 해결하려면 CORS 켜기를 선택합니다.

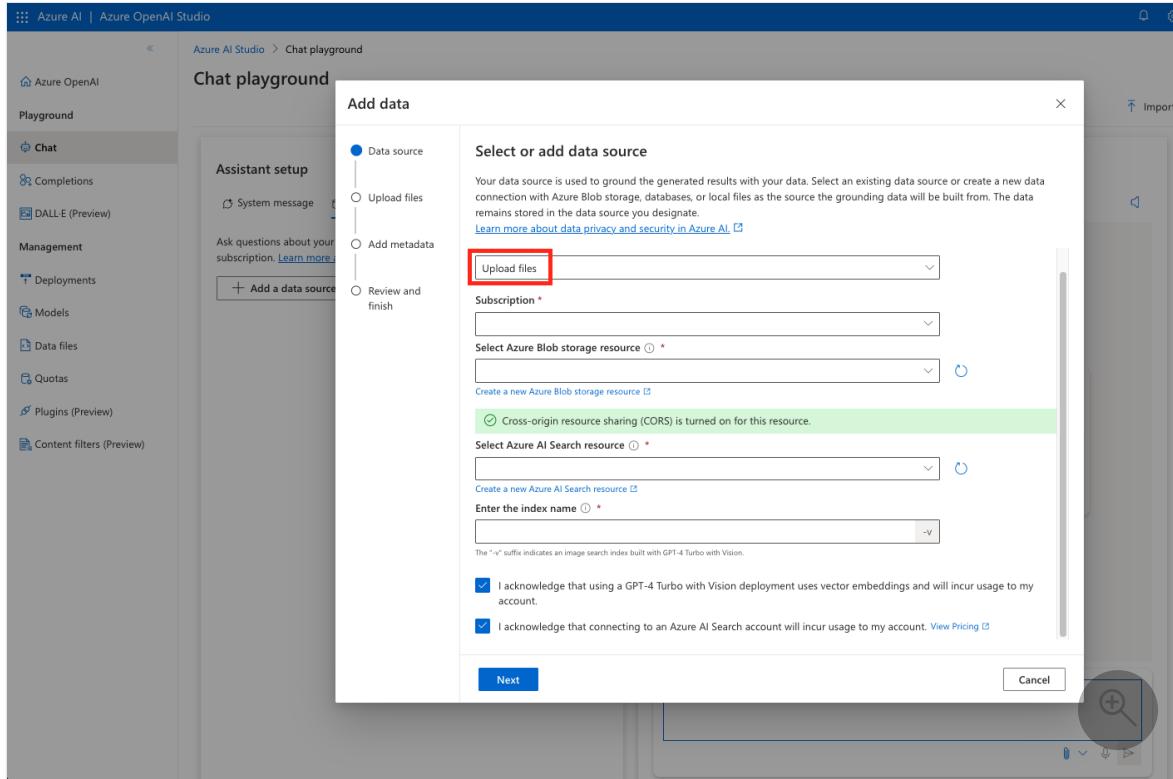
파일을 업로드하여 데이터 추가

Azure OpenAI를 사용하여 이미지 파일을 수동으로 업로드하고 메타데이터를 수동으로 입력할 수 있습니다. 이 방법은 소규모 이미지 집합을 실험하고 데이터 원본을 빌드하려는 경우에 특히 유용합니다.

1. 위에서 설명한 대로 Azure OpenAI에서 데이터 원본 선택 단추로 이동합니다. 파일 업로드를 선택합니다.

2. 구독을 선택합니다. 업로드된 이미지 파일이 저장될 Azure Blob Storage를 선택합니다. 새 이미지 검색 인덱스가 만들어질 Azure AI 검색 리소스를 선택합니다. 선택한 이미지 검색 인덱스의 이름을 입력합니다.

모든 필드를 채운 후, 하단에 있는 2개의 확인란을 선택하여 발생하는 사용량을 확인하고, **다음**을 선택합니다.

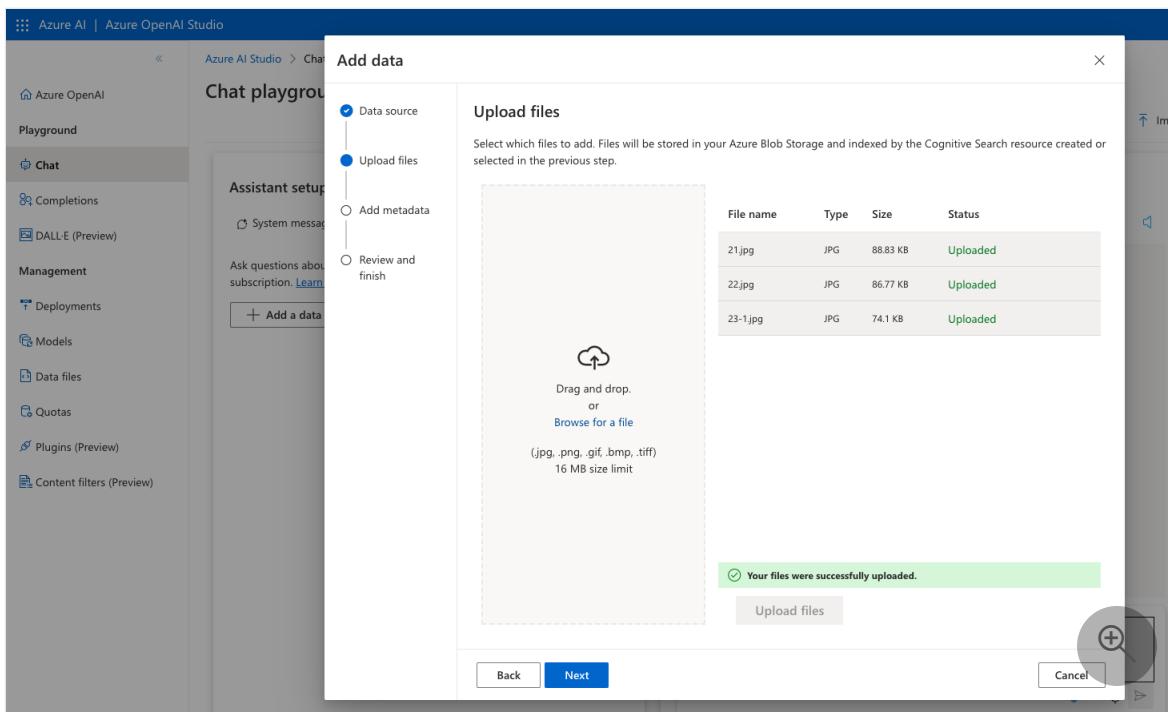


다음은 이미지 파일을 지원하는 파일 형식입니다.

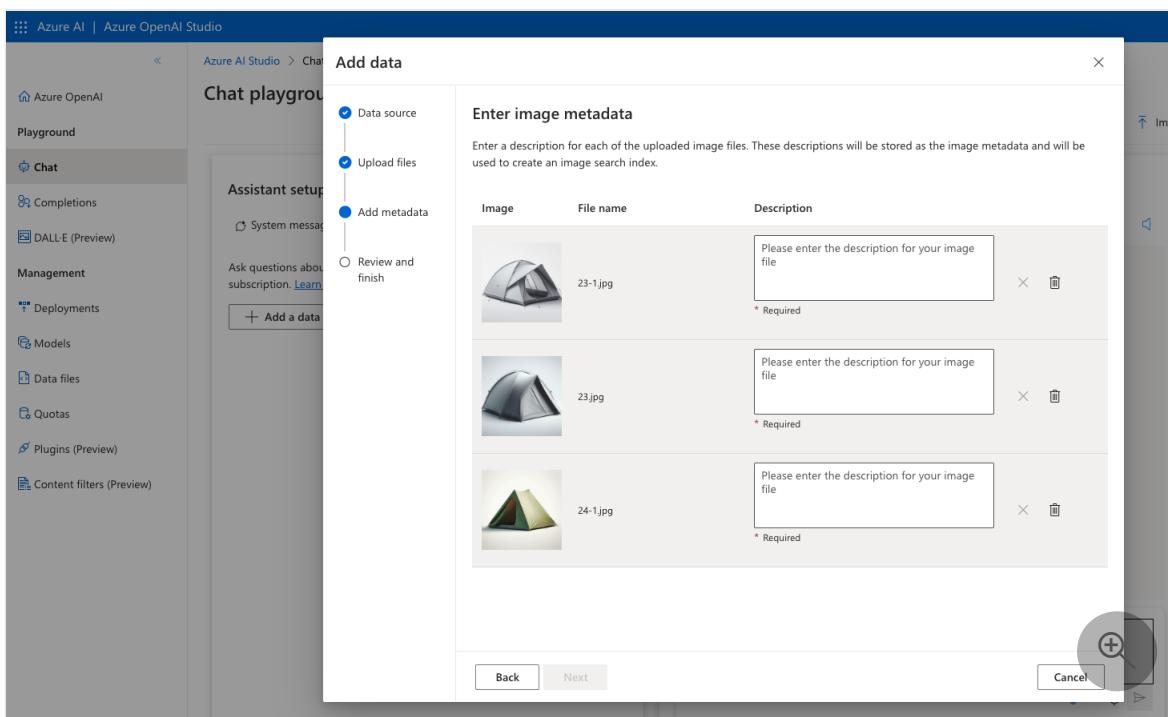
- .jpg
- .png
- .gif
- .bmp
- .tiff

3. 파일 찾아보기를 선택하여 로컬 디렉터리에서 사용할 이미지 파일을 선택합니다.

4. 이미지 파일을 선택하면 선택한 이미지 파일이 오른쪽 테이블에 표시됩니다. 파일 업로드를 선택합니다. 파일을 업로드하면 각 파일의 상태가 업로드됨으로 표시됩니다. **다음**을 선택합니다.



5. 이미지 파일마다 제공된 설명 필드에 메타데이터를 입력합니다. 각 이미지에 대한 설명이 있으면 다음을 선택합니다.



6. 모든 정보가 정확한지 검토합니다. 저장 후 닫기를 선택합니다.

Azure AI 검색을 사용하여 데이터 추가

기존 [Azure AI 검색](#) 인덱스가 있는 경우 데이터 원본으로 사용할 수 있습니다. 이미지의 검색 인덱스가 아직 없는 경우 [GitHub의 AI 검색 벡터 검색 리포지토리](#) 를 사용하여 만들 수 있습니다. 이 리포지토리는 이미지 파일로 인덱스를 만드는 스크립트를 제공합니다. 이 옵션은 위의 옵션처럼 사용자 고유의 파일을 사용하여 데이터 원본을 만든 다음, 플

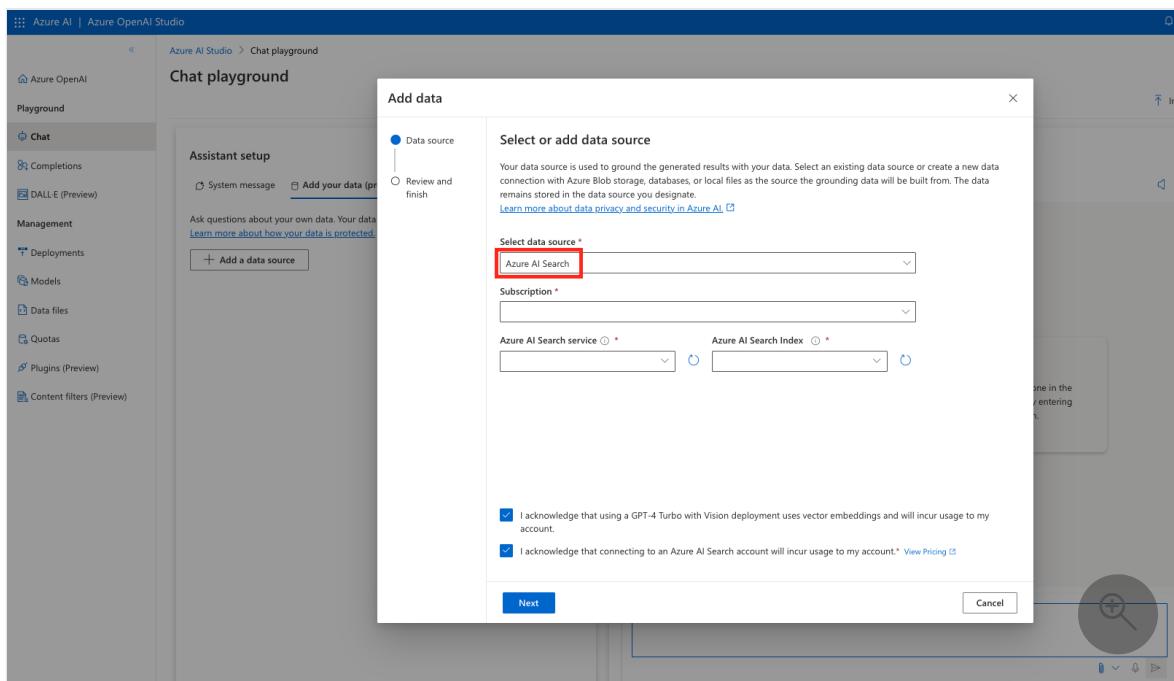
레이그라운드 환경으로 돌아와 이미 만들었지만 아직 추가하지 않은 데이터 원본을 선택하려는 경우에도 유용합니다.

1. 위에서 설명한 대로 Azure OpenAI에서 데이터 원본 선택 단추로 이동합니다. Azure AI 검색을 선택합니다.

💡 팁

Azure Blob Storage 또는 파일 업로드 옵션을 사용하여 만든 이미지 검색 인덱스를 선택해도 됩니다.

2. 구독을 선택하고, 이미지 검색 인덱스를 만드는데 사용한 Azure AI 검색 서비스를 선택합니다.
3. 이미지를 사용하여 Azure AI 검색 인덱스를 선택합니다.
4. 모든 필드를 채운 후, 페이지 하단에서 GPT-4 Turbo with Vision 및 Azure AI 검색을 사용하여 발생한 요금의 승인을 요청하는 확인란 2개를 선택합니다. 다음을 선택합니다. CORS가 아직 AI 검색 리소스에 대해 켜지지 않은 경우 경고가 표시됩니다. 경고를 해결하려면 CORS 켜기를 선택합니다.



5. 세부 정보를 검토함 후 저장하고 닫기를 선택합니다.

Azure Blob Storage를 사용하여 데이터 추가

기존 [Azure Blob Storage](#) 컨테이너가 있는 경우 이를 사용하여 이미지 검색 인덱스를 만들 수 있습니다. 새 Blob Storage를 만들려면 [Azure Blob Storage 빠른 시작](#) 설명서를 참

조하세요.

Blob Storage에는 이미지 파일과 이미지 파일 경로 및 메타데이터가 들어 있는 JSON 파일이 있어야 합니다. 이 옵션은 이미지 파일이 많고 각 파일을 수동으로 업로드하지 않으려는 경우에 특히 유용합니다.

이러한 파일로 채워진 Blob Storage가 아직 없고 파일을 하나씩 업로드하려는 경우 Azure OpenAI 스튜디오를 대신 사용하여 파일을 업로드할 수 있습니다.

Azure Blob Storage 컨테이너를 데이터 원본으로 추가하기 전에, 수집하려는 모든 이미지와 이미지 파일 경로 및 메타데이터가 포함된 JSON 파일이 Blob Storage에 있는지 확인해야 합니다.

① 중요

메타데이터 JSON 파일은 다음 조건을 충족해야 합니다.

- 파일 이름이 “metadata”라는 단어로 시작하고 공백이 없으며 모두 소문자여야 합니다.
- 허용되는 이미지 파일은 최대 10,000개입니다. 컨테이너에 이보다 많은 파일이 있는 경우 10,000개 이하의 이미지 파일을 포함하고 있는 JSON 파일을 여러 개 만들면 됩니다.

JSON

```
[  
  {  
    "image_blob_path": "image1.jpg",  
    "description": "description of image1"  
  },  
  {  
    "image_blob_path": "image2.jpg",  
    "description": "description of image2"  
  },  
  ...  
  {  
    "image_blob_path": "image50.jpg",  
    "description": "description of image50"  
  }  
]
```

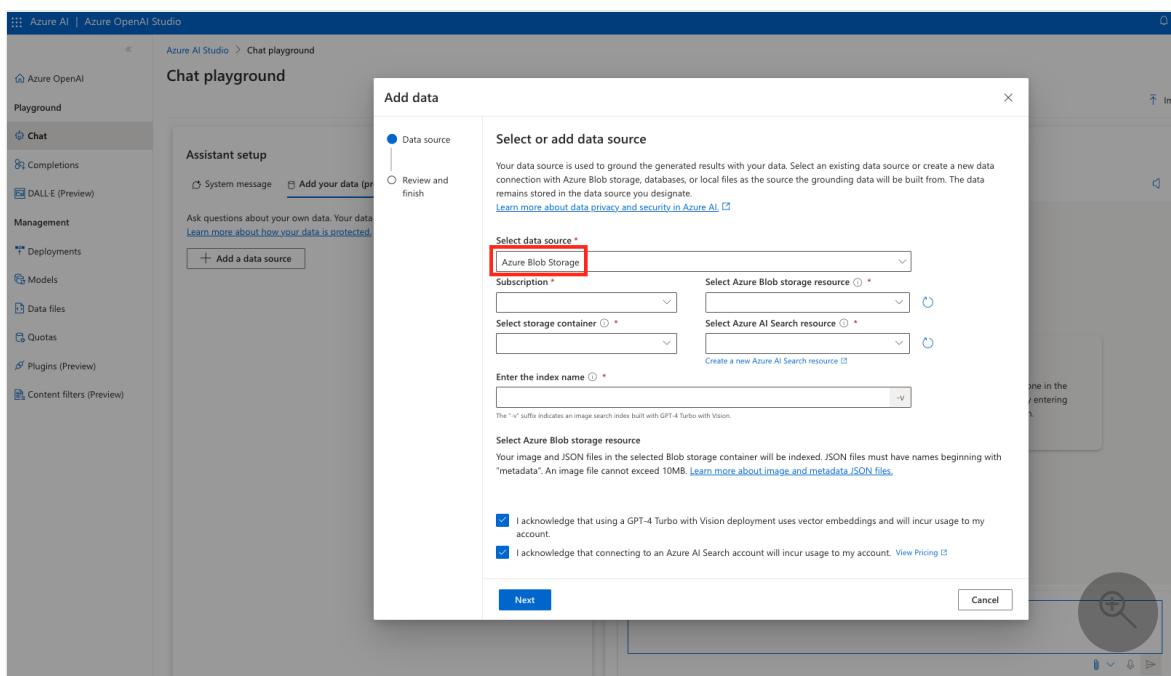
이미지 파일과 하나 이상의 메타데이터 JSON 파일로 채워진 Blob Storage가 있으면 Blob Storage를 데이터 원본으로 추가할 준비가 된 것입니다.

- 위에서 설명한 대로 Azure OpenAI에서 데이터 원본 선택 단추로 이동합니다. Azure Blob Storage를 선택합니다.
- 구독, Azure Blob Storage 및 스토리지 컨테이너를 선택합니다. 또한 이 리소스 그룹에 새 이미지 검색 인덱스가 만들어지므로 Azure AI 검색 리소스를 선택해야 합니다. Azure AI 검색 리소스가 없으면 드롭다운 아래의 링크를 사용하여 새로 만들 수 있습니다. Azure Blob Storage 리소스에 대해 CORS가 아직 켜져 있지 않은 경우 경고가 표시됩니다. 경고를 해결하려면 CORS 켜기를 선택합니다.
- Azure AI 검색 리소스를 선택한 후, 인덱스 이름 필드에 검색 인덱스의 이름을 입력합니다.

① 참고

제공된 이미지에서 추출한 이미지 벡터를 사용하는 인덱스라는 것을 나타내기 위해 인덱스 이름에 `-v` 접미사가 추가됩니다. `metadata.json`의 설명 필드는 인덱스의 텍스트 메타데이터로 추가됩니다.

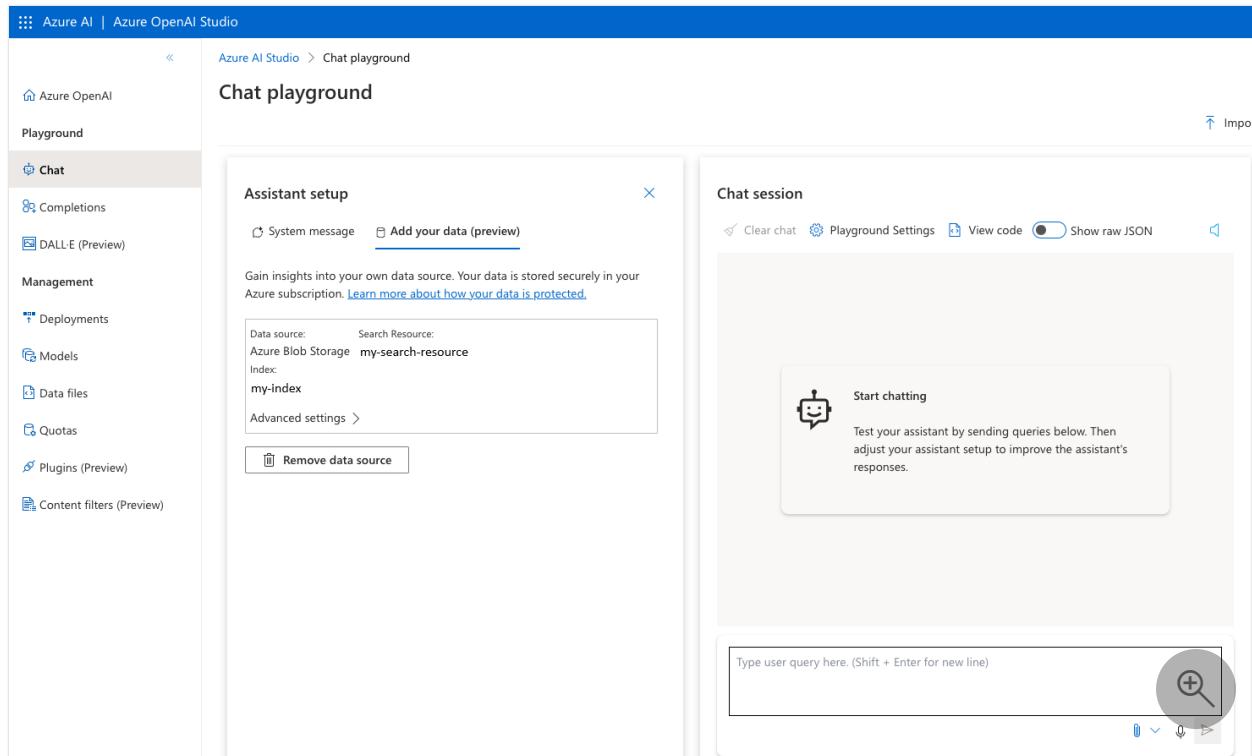
- 모든 필드를 채운 후, 페이지 하단에서 GPT-4 Turbo with Vision 및 Azure AI 검색을 사용하여 발생한 요금의 승인을 요청하는 확인란 2개를 선택합니다. 다음을 선택합니다.



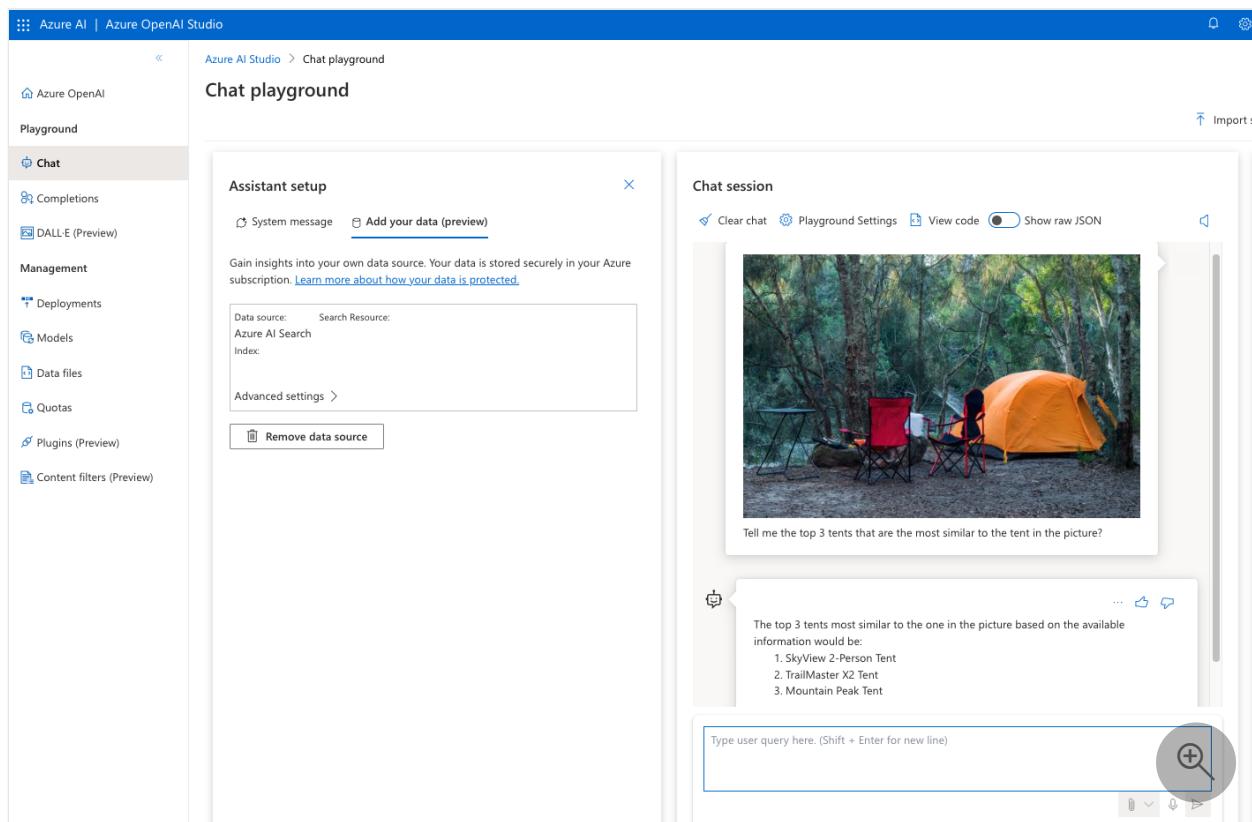
- 세부 정보를 검토한 후 저장하고 닫기를 선택합니다.

GPT-4 Turbo with Vision 모델에서 수집된 데이터 사용

위에서 설명한 세 가지 방법 중 하나로 데이터 원본을 연결한 후, 데이터 수집 프로세스가 완료될 때까지 다소 시간이 걸립니다. 프로세스가 진행되면서 아이콘과 **수집 진행 중** 메시지가 표시됩니다. 수집이 완료되면 데이터 원본이 만들어진 것을 볼 수 있습니다.

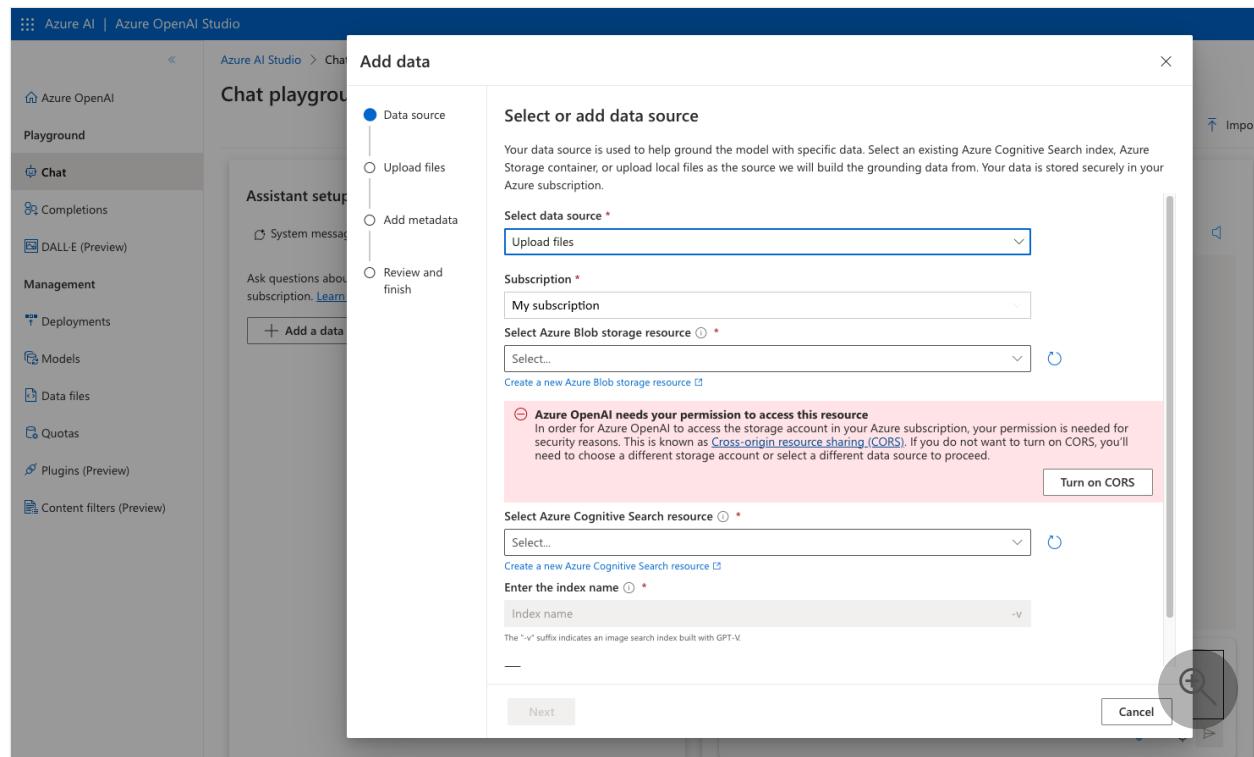


데이터 원본 수집이 완료되면 데이터 원본 세부 정보와 이미지 검색 인덱스 이름이 표시됩니다. 이제 이 수집된 데이터를 배포된 GPT-4 Turbo with Vision 모델의 그라운딩 데이터로 사용할 수 있습니다. 모델은 이미지 검색 인덱스의 상위 검색 데이터를 사용하고, 수집된 데이터를 구체적으로 준수하는 응답을 생성합니다.



CORS 켜기

데이터 원본에 대해 CORS를 아직 켜지 않은 경우 다음 메시지가 표시됩니다.



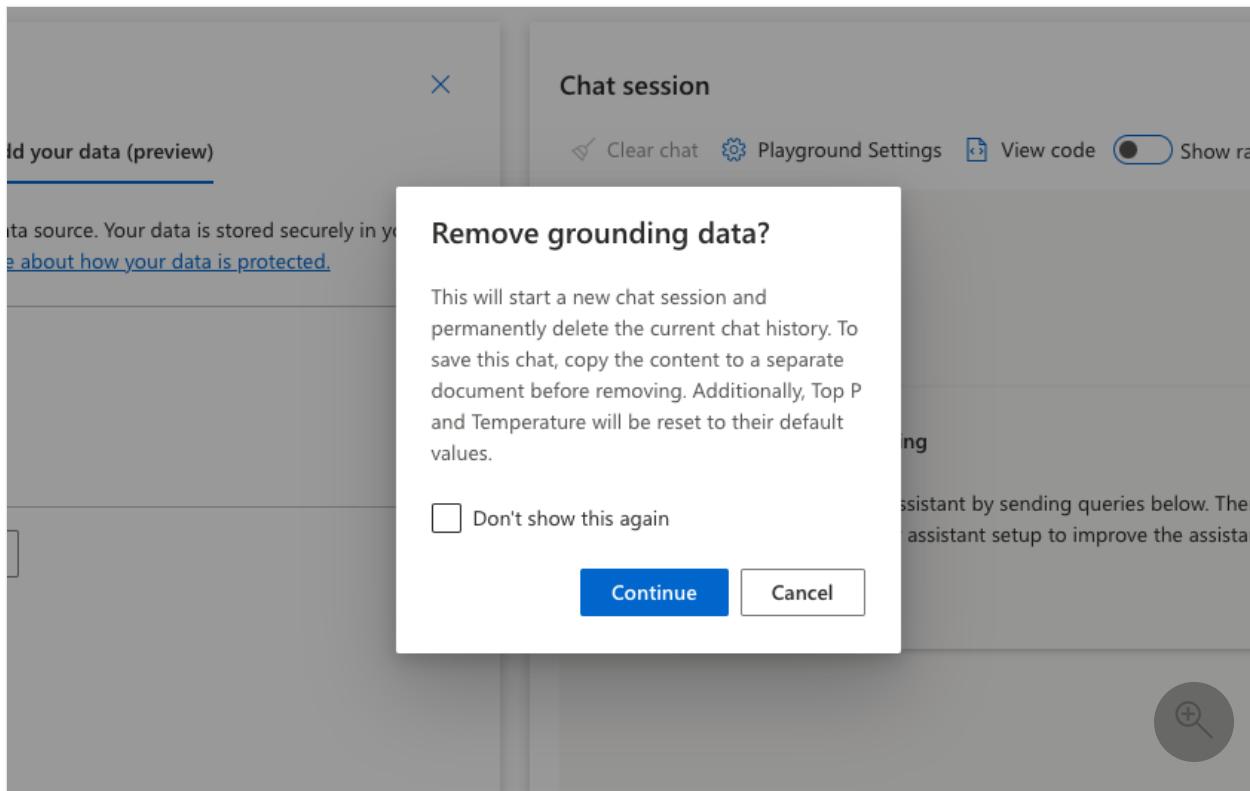
이 메시지가 표시되면 데이터 원본을 연결할 때 CORS 켜기를 선택합니다.

추가 팁

데이터 원본 추가 및 제거

Azure OpenAI는 현재 채팅 세션당 하나의 데이터 원본만 사용할 수 있습니다. 새 데이터 원본을 추가하려면 먼저 기존 데이터 원본을 제거해야 합니다. 이 작업은 데이터 원본 정보 아래에서 **데이터 원본 제거**를 선택하여 수행할 수 있습니다.

데이터 원본을 제거하면 경고 메시지가 표시됩니다. 데이터 원본을 제거하면 채팅 세션이 지워지고 모든 플레이그라운드 설정이 초기화됩니다.



ⓘ 중요

GPT-4 Turbo with Vision 모델을 사용하지 않는 모델 배포로 전환하면 데이터 원본 제거에 대한 경고 메시지가 표시됩니다. 데이터 원본을 제거하면 채팅 세션이 지워지고 모든 플레이그라운드 설정이 초기화됩니다.

다음 단계

- Azure OpenAI 텍스트 모델에서도 채팅할 수 있습니다. 자세한 내용은 [텍스트 데이터 사용](#)을 참조하세요.
- 또는 [빠른 시작](#)에 따라 채팅 시나리오에서 GPT-4 Turbo with Vision을 사용합니다.
- [GPT-4 Turbo with Vision FAQ\(질문과 대답\)](#)
- [GPT-4 Turbo with Vision API 참조](#)

데이터에 대한 Azure OpenAI를 안전하게 사용

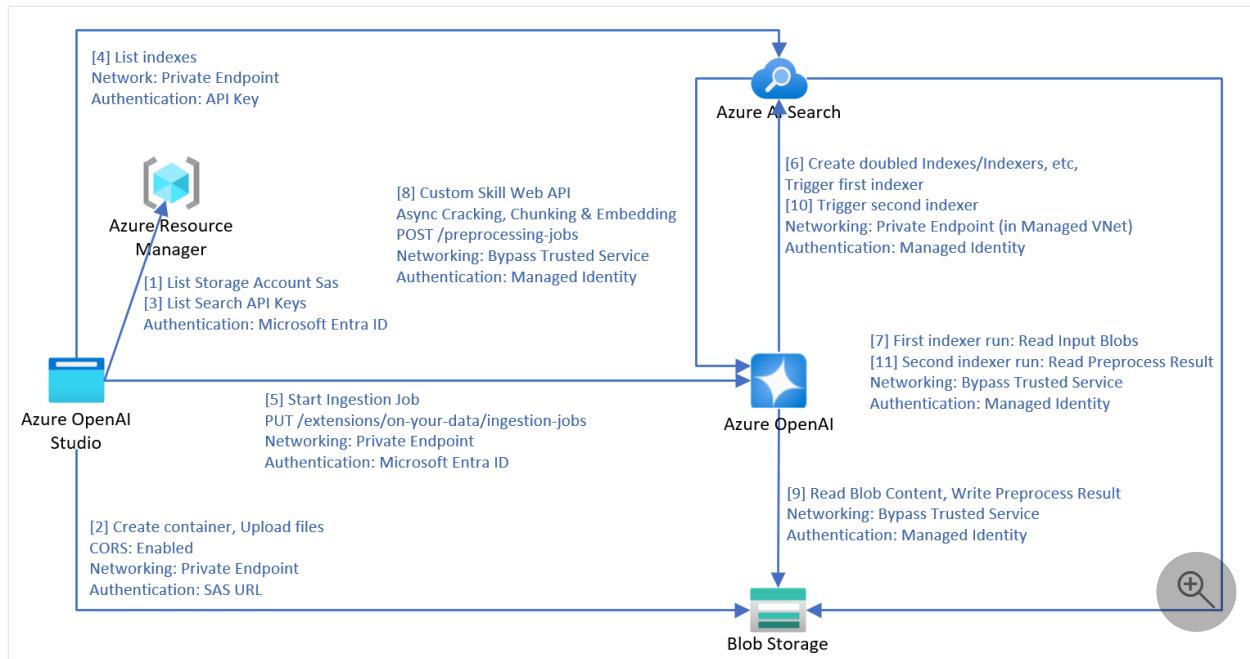
아티클 • 2024. 04. 05.

이 문서를 통해 Microsoft Entra ID 역할 기반 액세스 제어, 가상 네트워크 및 프라이빗 엔드포인트로 데이터와 리소스를 보호하여 Azure OpenAI On Your Data를 안전하게 사용하는 방법을 알아봅니다.

이 문서는 [텍스트가 포함된 Azure OpenAI On Your Data](#)를 사용하는 경우에만 적용됩니다. [이미지가 포함된 Azure OpenAI On Your Data](#)에는 적용되지 않습니다.

데이터 수집 아키텍처

Azure OpenAI On Your Data를 사용하여 Azure Blob Storage, 로컬 파일 또는 URL의 데이터를 Azure AI 검색으로 수집하는 경우 다음 프로세스를 사용하여 데이터를 처리합니다.

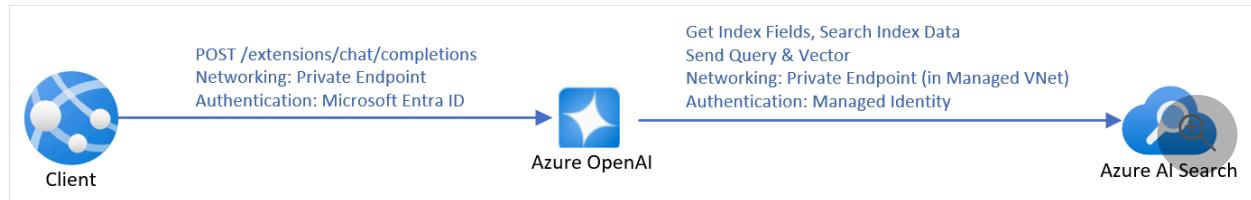


- 1단계와 2단계는 파일 업로드에만 사용됩니다.
- Blob Storage에 대한 URL 다운로드는 이 디아이어그램에 나와 있지 않습니다. 웹 페이지를 인터넷에서 다운로드하고 Blob Storage에 업로드한 후 3단계 이후는 동일합니다.
- 두 개의 인덱서, 두 개의 인덱스, 두 개의 데이터 원본 및 하나의 [사용자 지정 기술](#)이 Azure AI 검색 리소스에 만들어집니다.
- 청크 컨테이너는 Blob Storage에 만들어집니다.
- [예약된 새로 고침](#)에 의해 수집이 트리거되면 수집 프로세스가 7단계부터 시작됩니다.

- Azure OpenAI의 `preprocessing-jobs` API는 Azure AI 검색 고객 기술 웹 API 프로토콜을 구현하고 큐의 문서를 처리합니다.
- Azure OpenAI:
 1. 내부적으로는 이전에 만들어진 첫 번째 인덱서를 사용하여 문서를 크래ップ니다.
 2. 휴리스틱 기반 알고리즘을 사용하여 청크를 수행하고 청크 경계의 테이블 레이아웃과 기타 서식 지정 요소를 준수하여 최고의 청크 품질을 보장합니다.
 3. 벡터 검색을 사용하도록 설정하도록 선택한 경우 Azure OpenAI는 선택한 포함 배포를 사용하여 청크를 내부적으로 벡터화합니다.
- 서비스가 모니터링하는 모든 데이터가 처리되면 Azure OpenAI는 두 번째 인덱서를 트리거합니다.
- 인덱서는 처리된 데이터를 Azure AI 검색 서비스에 저장합니다.

서비스 호출에 사용되는 관리 ID의 경우 시스템 할당 관리 ID만 지원됩니다. 사용자 할당 관리 ID는 지원되지 않습니다.

유추 아키텍처



데이터에 대한 Azure OpenAI 모델과 채팅하기 위해 API 호출을 보내는 경우 서비스는 필드 매핑이 요청에 명시적으로 설정되지 않은 경우 자동으로 필드 매핑을 수행하기 위해 유추 중에 인덱스 필드를 검색해야 합니다. 따라서 유추 중에도 검색 서비스에 대한 `Search Service Contributor` 역할을 가지려면 서비스에서 Azure OpenAI ID가 필요합니다.

유추 요청에 포함 배포가 제공되는 경우 다시 작성된 쿼리는 Azure OpenAI에 의해 벡터화되고 쿼리와 벡터 모두 벡터 쿼리를 위해 Azure AI 검색으로 전송됩니다.

문서 수준 액세스 제어

① 참고

문서 수준 액세스 제어는 Azure AI 검색에 대해서만 지원됩니다.

Azure OpenAI On Your Data를 사용하면 Azure AI Search [보안 필터](#)를 사용하여 다른 사용자에 대한 응답에 사용할 수 있는 문서를 제한할 수 있습니다. 문서 수준 액세스를 사용하도록 설정하면 Azure AI Search에서 반환되고 응답을 생성하는 데 사용되는 검색 결과가 사용자 Microsoft Entra 그룹 멤버 자격에 따라 잘립니다. 기존 Azure AI Search 인덱스에서만 문서 수준 액세스를 사용하도록 설정할 수 있습니다. 문서 수준 액세스를 사용하도록 설정하려면 다음을 수행합니다.

1. [Azure AI Search 설명서](#)의 단계에 따라 애플리케이션을 등록하고 사용자 및 그룹을 만듭니다.
2. [허용된 그룹으로 문서를 인덱싱합니다.](#) 새 [보안 필드](#)에 아래의 스키마가 있는지 확인합니다.

JSON

```
{"name": "group_ids", "type": "Collection(Edm.String)", "filterable": true }
```

`group_ids`는 기본 필드 이름입니다. `my_group_ids` 등의 다른 필드 이름을 사용하는 경우 [인덱스 필드 매핑](#)에서 필드를 매핑할 수 있습니다.

3. 인덱스의 각 중요한 문서에 대해 이 보안 필드에 올바르게 설정된 값이 있는지 확인하여 문서의 허용된 그룹을 나타냅니다.
4. [Azure OpenAI Studio](#)에서 데이터 원본을 추가합니다. [인덱스 필드 매핑](#) 섹션에서 스키마가 호환되는 한, [허용된 그룹](#) 필드에 0개 또는 1개의 값을 매핑할 수 있습니다. [허용된 그룹](#) 필드가 매핑되지 않으면 문서 수준 액세스가 사용하도록 설정되지 않습니다.

Azure OpenAI Studio

Azure AI Search 인덱스가 연결되면 스튜디오의 응답은 로그인한 사용자의 Microsoft Entra 권한에 따라 문서 액세스 권한을 갖습니다.

웹 앱

게시된 [웹 앱](#)을 사용하는 경우 최신 버전으로 업그레이드하려면 다시 배포해야 합니다. 최신 버전의 웹 앱에는 로그인한 사용자의 Microsoft Entra 계정 그룹을 검색하고, 캐시하고, 각 API 요청에 그룹 ID를 포함하는 기능이 포함되어 있습니다.

API

API를 사용하는 경우 각 API 요청에 `filter` 매개 변수를 전달합니다. 예시:

JSON

```
{
  "messages": [
    {
      "role": "user",
      "content": "who is my manager?"
    }
  ],
  "dataSources": [
    {
      "type": "AzureCognitiveSearch",
      "parameters": {
        "endpoint": "'$AZURE_AI_SEARCH_ENDPOINT'",
        "key": "'$AZURE_AI_SEARCH_API_KEY'",
        "indexName": "'$AZURE_AI_SEARCH_INDEX'",
        "filter": "my_group_ids/any(g:search.in(g, 'group_id1,
group_id2'))"
      }
    }
  ]
}
```

- `my_group_ids`는 [필드 매팅](#) 중에 허용된 그룹에 대해 선택한 필드 이름입니다.
- `group_id1, group_id2`는 로그인한 사용자에서 기인한 그룹입니다. 클라이언트 애플리케이션은 사용자 그룹을 검색하고 캐시할 수 있습니다.

리소스 구성

다음 섹션을 사용하여 최적의 보안 사용을 위해 리소스를 구성합니다. 리소스의 일부만 보호하려는 경우에도 아래 단계를 모두 수행해야 합니다.

이 문서에서는 Azure OpenAI 리소스, Azure AI 검색 리소스 및 스토리지 계정에 대한 공용 네트워크를 사용하지 않도록 설정하는 데 관련된 네트워크 설정을 설명합니다. 서비스의 IP 주소가 동적이므로 선택한 네트워크를 IP 규칙과 함께 사용하는 것은 지원되지 않습니다.

💡 팁

[GitHub](#)에서 사용할 수 있는 bash 스크립트를 사용하여 설치 유효성을 검사하고, 여기에 나열된 모든 요구 사항이 충족되고 있는지 확인할 수 있습니다.

리소스 그룹 만들기

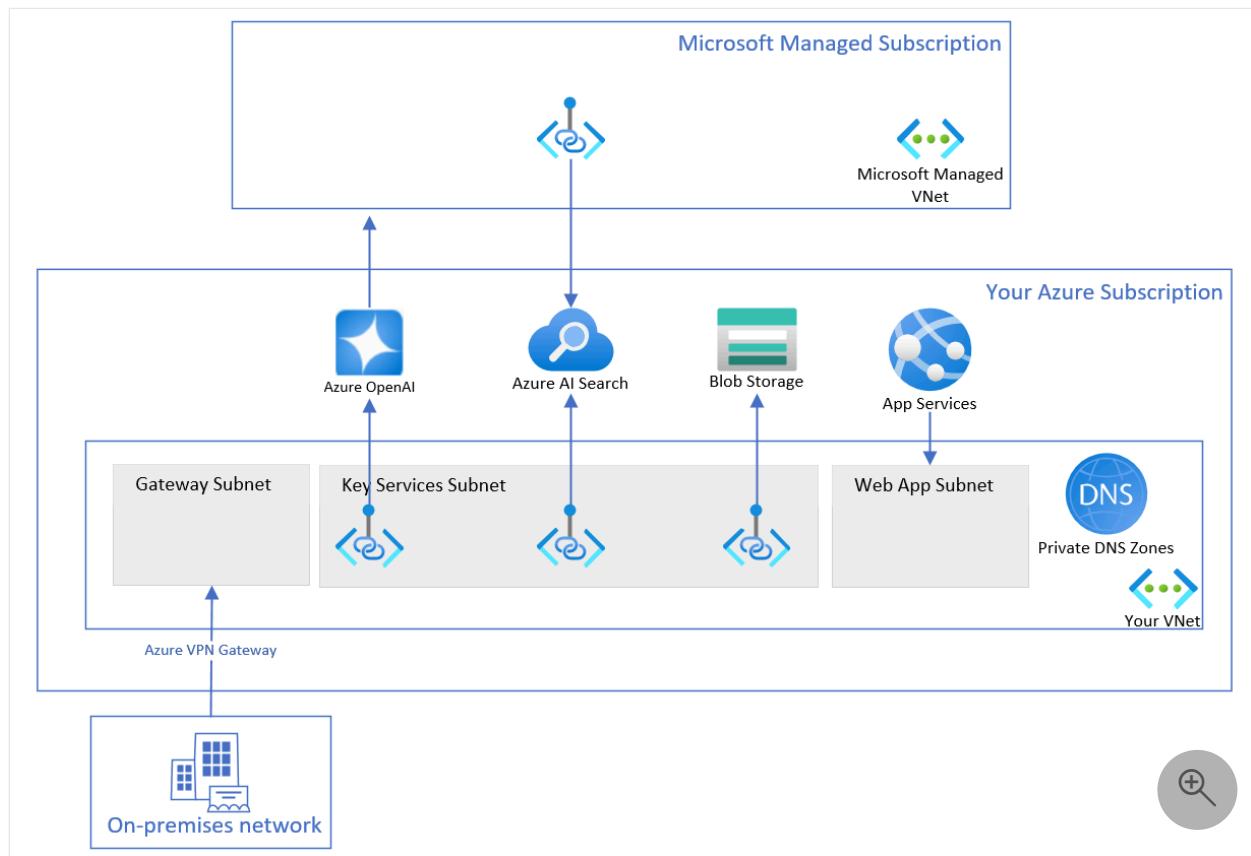
모든 관련 리소스를 구성할 수 있도록 리소스 그룹을 만듭니다. 리소스 그룹의 리소스에는 다음이 포함되지만 이에 국한되지는 않습니다.

- 가상 네트워크 1개
- 세 가지 주요 서비스: Azure OpenAI 1개, Azure AI 검색 1개, 스토리지 계정 1개
- 3개의 프라이빗 엔드포인트, 각각 하나의 키 서비스에 연결됨
- 3개의 네트워크 인터페이스(각각은 하나의 프라이빗 엔드포인트와 연결됨)
- 온-프레미스 클라이언트 컴퓨터에서 액세스하기 위한 가상 네트워크 게이트웨이 1개
- 가상 네트워크가 통합된 하나의 웹앱
- 하나의 프라이빗 DNS 영역(웹앱이 Azure OpenAI의 IP를 찾을 수 있도록)

가상 네트워크 만들기

가상 네트워크에는 3개의 서브넷이 있습니다.

1. 첫 번째 서브넷은 세 프라이빗 엔드포인트의 개인 IP에 사용됩니다.
2. 두 번째 서브넷은 가상 네트워크 게이트웨이를 만들 때 자동으로 만들어집니다.
3. 세 번째 서브넷은 비어 있으며 웹앱 아웃바운드 가상 네트워크 통합에 사용됩니다.



Microsoft 관리되는 가상 네트워크는 Microsoft에서 만들어졌으므로 볼 수 없습니다.
Microsoft 관리되는 가상 네트워크는 Azure OpenAI에서 Azure AI 검색에 안전하게 액세스하는 데 사용됩니다.

Azure OpenAI 구성

사용자 지정 하위 도메인을 사용하도록 설정했습니다.

Azure Portal을 통해 Azure OpenAI를 만든 경우 [사용자 지정 하위 도메인](#)이 이미 만들어져 있어야 합니다. Microsoft Entra ID 기반 인증 및 프라이빗 DNS 영역에는 사용자 지정 하위 도메인이 필요합니다.

관리 ID 사용

Azure AI 검색 및 스토리지 계정이 Microsoft Entra ID 인증을 통해 Azure OpenAI 서비스를 인식할 수 있도록 하려면 Azure OpenAI 서비스에 대한 관리 ID를 할당해야 합니다. 가장 쉬운 방법은 Azure Portal에서 시스템이 할당한 관리 ID를 켜는 것입니다.

The screenshot shows the Azure OpenAI service configuration page under the 'Identity' section. The 'System assigned' tab is selected. The status is set to 'On'. There is a placeholder for the object (principal) ID and a 'Permissions' section with a 'Azure role assignments' button.

관리 API를 통해 관리 ID를 설정하려면 [관리 API 참조 설명서](#)를 확인합니다.

```
JSON
{
  "identity": {
    "principalId": "12345678-abcd-1234-5678-abc123def",
    "tenantId": "1234567-abcd-1234-1234-abcd1234",
    "type": "SystemAssigned, UserAssigned",
    "userAssignedIdentities": {
      "/subscriptions/1234-5678-abcd-1234-1234abcd/resourceGroups/my-resource-group",
      "principalId": "12345678-abcd-1234-5678-abcdefg1234",
      "clientId": "12345678-abcd-efgh-1234-12345678"
    }
  }
}
```

신뢰할 수 있는 서비스 사용

Azure AI 검색이 Azure OpenAI `preprocessing-jobs`를 사용자 지정 기술 웹 API로 호출하도록 허용하려면 Azure OpenAI에는 공용 네트워크 액세스가 없지만 Azure AI 검색을 관리 ID 기반의 신뢰할 수 있는 서비스로 무시하도록 Azure OpenAI를 설정해야 합니다.

Azure OpenAI는 JWT(JSON Web Token)의 클레임을 확인하여 Azure AI 검색의 트래픽을 식별합니다. Azure AI 검색은 사용자 지정 기술 웹 API를 호출하려면 시스템이 할당한 관리 ID 인증을 사용해야 합니다.

관리 API에서 `networkAccls.bypass`를 `AzureServices`로 설정합니다. 자세한 내용은 [가상 네트워크 문서](#)를 참조하세요.

Azure AI 검색 리소스에 대한 [공유 프라이빗 링크](#)가 있는 경우에만 이 단계를 건너뛸 수 있습니다.

공용 네트워크 액세스 사용 안 함

Azure Portal에서 Azure OpenAI 리소스의 공용 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다.

Azure OpenAI Studio를 사용하는 것과 같이 클라이언트 컴퓨터에서 Azure OpenAI 서비스에 대한 액세스를 허용하려면 Azure OpenAI 리소스에 연결하는 [프라이빗 엔드포인트 연결](#)을 만들어야 합니다.

Azure AI 검색 구성

아래 구성에는 기본 가격 책정 계층 이상을 사용할 수 있습니다. 필수는 아니지만 S2 가격 책정 계층을 사용하는 경우 선택 가능한 [추가 옵션](#)이 표시됩니다.

관리 ID 사용

다른 리소스가 Microsoft Entra ID 인증을 사용하여 Azure AI 검색을 인식할 수 있도록 하려면 Azure AI 검색에 대한 관리 ID를 할당해야 합니다. 가장 쉬운 방법은 Azure Portal에서 시스템이 할당한 관리 ID를 켜는 것입니다.

Identity

System assigned

A system assigned managed identity is restricted to one per resource and is authenticated with Microsoft Entra ID, so you don't have to store secrets.

Status: On

Object (principal) ID:

Azure role assignments

역할 기반 액세스 제어 사용

Azure OpenAI는 관리 ID를 사용하여 Azure AI 검색에 액세스하므로 Azure AI 검색에서 역할 기반 액세스 제어를 사용하도록 설정해야 합니다. Azure Portal에서 이 작업을 수행 하려면 Azure Portal의 키 탭에서 **모두**를 선택합니다.

Keys

API access control

API keys

Role-based access control

Both

Manage admin keys

REST API를 통해 역할 기반 액세스 제어를 사용하도록 설정하려면 `authOptions`를 `aadOrApiKey`로 설정합니다. 자세한 내용은 [Azure AI 검색 RBAC 문서](#)를 참조하세요.

JSON

```
"disableLocalAuth": false,  
"authOptions": {  
    "aadOrApiKey": {  
        "aadAuthFailureMode": "http401WithBearerChallenge"  
    }  
}
```

Azure OpenAI Studio를 사용하려면 Azure AI 검색에 대한 API 키 기반 인증을 사용하지 않도록 설정할 수 없습니다. Azure OpenAI Studio는 API 키를 사용하여 브라우저에서 Azure AI 검색 API를 호출하기 때문입니다.

💡 팁

최상의 보안을 위해 프로덕션 준비가 되었고 더 이상 테스트를 위해 Azure OpenAI Studio를 사용할 필요가 없으면 API 키를 사용하지 않도록 설정하는 것이 좋습니다. 자세한 내용은 [Azure AI 검색 RBAC 문서](#)를 참조하세요.

공용 네트워크 액세스 사용 안 함

Azure Portal에서 Azure AI 검색 리소스의 공용 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다.

Azure OpenAI Studio를 사용하는 것과 같이 클라이언트 컴퓨터에서 Azure AI 검색 리소스에 대한 액세스를 허용하려면 Azure AI 검색 리소스에 연결하는 [프라이빗 엔드포인트 연결](#)을 만들어야 합니다.

ⓘ 참고

Azure OpenAI 리소스에서 Azure AI 검색 리소스에 대한 액세스를 허용하려면 [애플리케이션 양식](#)을 제출해야 합니다. 신청서는 영업일 기준 10일 이내에 검토되며 결과는 이메일을 통해 연락드립니다. 자격이 있는 경우 Microsoft 관리되는 가상 네트워크에서 프라이빗 엔드포인트를 프로비전하고 검색 서비스에 프라이빗 엔드포인트 연결 요청을 보내며, 사용자는 요청을 승인해야 합니다.

The screenshot shows the Azure AI services | Cognitive search test Networking page. On the left, there's a sidebar with options like Search, Search management, Settings, Semantic search (Preview), Knowledge Center, Keys, Scale, Search traffic analytics, Identity, Networking (which is selected and highlighted in grey), and Properties. The main area has tabs for Public access, Private access (which is selected), and Shared private access. A note says: "Private endpoints allow access to this resource using a private IP address from a virtual network, effectively bringing the service into your network. Learn more." Below this is a section titled "Private endpoint connections" with a note: "Allow selected virtual networks to connect to your resource using private endpoints." It includes buttons for "+ Create a private endpoint", "Refresh", "Approve" (with a checkmark), "Reject", and "Remove". There's also a "Filter by name..." input field and a search icon. A table lists one connection: "test" (Private endpoint), "test" (Connection name), "searchService" (Sub-resource), "Pending" (Connection state), and "Azure OpenAI on your data" (Description). A magnifying glass icon is at the bottom right of the table.

프라이빗 엔드포인트 리소스는 연결된 리소스가 테넌트에 있는 동안 Microsoft 관리 테넌트에 프로비전됩니다. 네트워킹 페이지의 **프라이빗 액세스** 탭에서 **프라이빗 엔드포인트** 링크(파란색 글꼴)를 그냥 클릭하는 것으로는 프라이빗 엔드포인트 리소스에 액세스할 수 없습니다. 대신 행의 다른 곳을 클릭하면 위의 **승인** 단추를 클릭할 수 있습니다.

[수동 승인 작업 흐름](#)에 대해 자세히 알아봅니다.

공유 프라이빗 링크 만들기

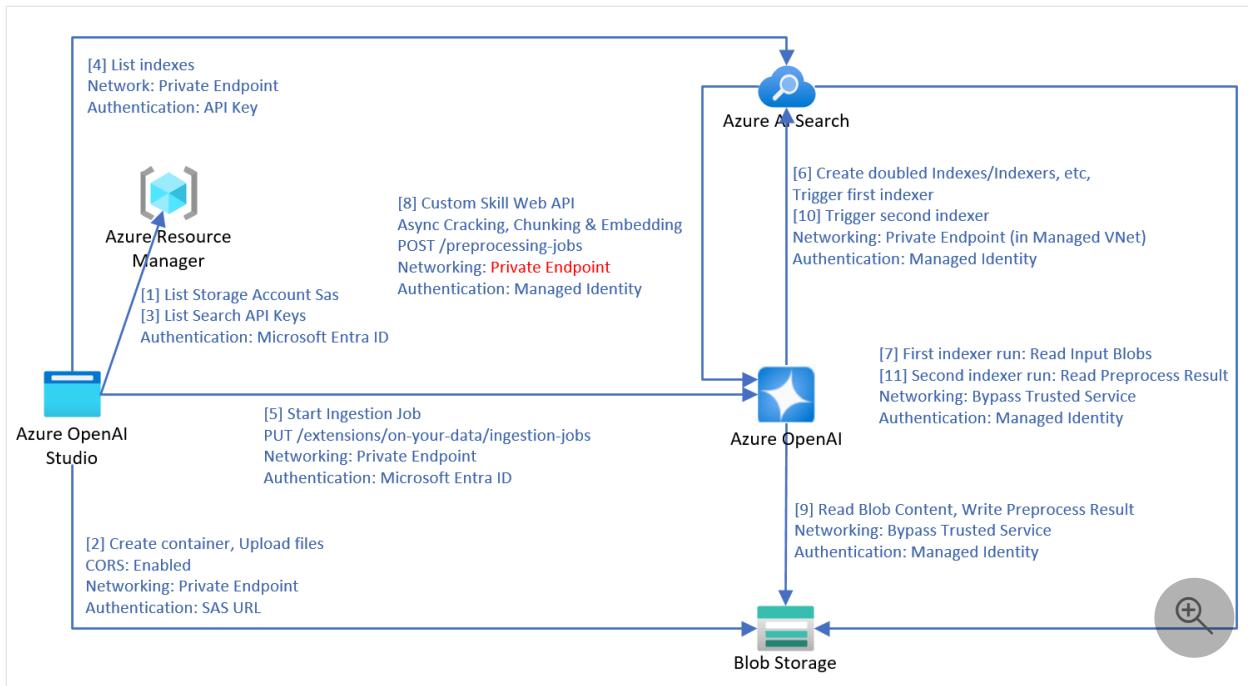
💡 팁

기본 또는 표준 가격 책정 계층을 사용하거나 모든 리소스를 안전하게 설정하는 것 이 처음인 경우 이 고급 항목을 건너뛰어야 합니다.

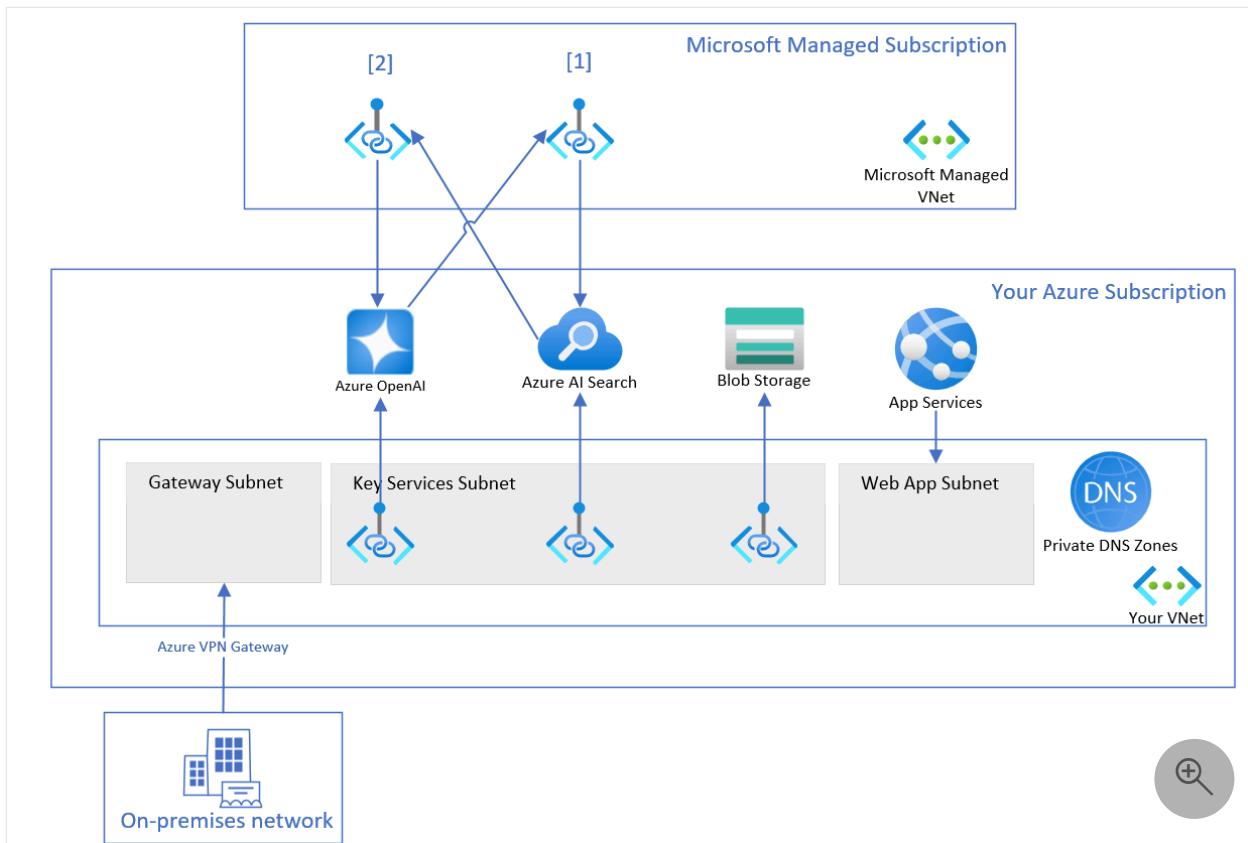
이 섹션은 [기술 집합이 있는 인덱서에 대한 프라이빗 엔드포인트 지원](#)이 필요하기 때문에 S2 가격 책정 계층 검색 리소스에만 적용됩니다.

Azure OpenAI 리소스에 연결하는 검색 리소스에서 공유 프라이빗 링크를 만들려면 [검색 설명서](#)를 참조하세요. 리소스 종류를 `Microsoft.CognitiveServices/accounts`로 그룹 ID 를 `openai_account`로 선택합니다.

공유 프라이빗 링크를 사용하면 데이터 수집 아키텍처 디아이어그램의 [8단계](#)가 신뢰할 수 있는 서비스 우회에서 **프라이빗 엔드포인트**로 변경됩니다.



만든 Azure AI 검색 공유 프라이빗 링크는 가상 네트워크가 아닌 Microsoft 관리형 가상 네트워크에도 있습니다. 앞서 만든 다른 관리형 프라이빗 엔드포인트와의 차이점은 Azure OpenAI에서 Azure Search까지의 관리형 프라이빗 엔드포인트 [1] 가 **양식 애플리케이션**을 통해 프로비전되는 반면, Azure Cognitive Search에서 Azure OpenAI로의 관리형 프라이빗 엔드포인트 [2] 는 Azure Portal 또는 Azure Cognitive Search의 REST API를 통해 프로비전된다는 점입니다.



스토리지 계정 구성

신뢰할 수 있는 서비스 사용

Azure OpenAI 및 Azure AI Search에서 스토리지 계정에 대한 액세스를 허용하려면 스토리지 계정에는 공용 네트워크 액세스가 없지만 Azure OpenAI 및 Azure AI Search를 관리 ID 기반의 신뢰할 수 있는 서비스로 우회하도록 스토리지 계정을 설정해야 합니다.

Azure Portal에서 스토리지 계정 네트워킹 탭으로 이동하여 "선택한 네트워크"를 선택한 다음 신뢰할 수 있는 서비스 목록의 Azure 서비스가 이 스토리지 계정에 액세스하도록 허용을 선택하고 저장을 클릭합니다.

① 참고

신뢰할 수 있는 서비스 기능은 위에서 설명한 명령줄에서만 사용할 수 있으며 Azure Portal에서는 수행할 수 없습니다.

공용 네트워크 액세스 사용 안 함

Azure Portal에서 스토리지 계정의 공용 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다.

Azure OpenAI Studio를 사용하는 것과 같이 클라이언트 컴퓨터에서 스토리지 계정에 대한 액세스를 허용하려면 Blob Storage에 연결하는 [프라이빗 엔드포인트 연결](#)을 만들어야 합니다.

역할 할당

지금까지는 이미 각 리소스 작업을 독립적으로 설정했습니다. 다음으로 서비스가 서로 권한 부여할 수 있도록 허용해야 합니다.

[] 테이블 확장

역할	담당자	리소스	설명
Search Index Data Reader	Azure OpenAI	Azure AI 검색	유추 서비스는 인덱스에서 데이터를 쿼리합니다.
Search Service Contributor	Azure OpenAI	Azure AI 검색	유추 서비스는 자동 필드 매핑을 위해 인덱스 스키마를 쿼리합니다. 데이터 수집 서비스는 인덱스, 데이터 원본, 기술 집합, 인덱서를 만들고 인덱서 상태를 쿼리합니다.
Storage Blob Data Contributor	Azure OpenAI	스토리지 계정	입력 컨테이너에서 읽고 전처리 결과를 출력 컨테이너에 씁니다.

역할	담당자	리소스	설명
Cognitive Services Contributor	Azure AI 검색	Azure OpenAI	사용자 지정 기술
Storage Blob Data Contributor	Azure AI 검색	스토리지 계정	BLOB을 읽고 지식 저장소를 씁니다.

위 표에서 **Assignee**는 해당 리소스의 시스템이 할당한 관리 ID를 의미합니다.

역할 할당을 추가하려면 관리자에게 이러한 리소스에 대한 **Owner** 역할이 있어야 합니다.

Azure Portal에서 이러한 역할을 설정하는 방법에 대한 자침은 [Azure RBAC 설명서](#)를 참조하세요. [GitHub에서 사용 가능한 스크립트](#)를 사용하여 프로그래밍 방식으로 역할 할당을 추가할 수 있습니다.

개발자가 이러한 리소스를 사용하여 애플리케이션을 빌드할 수 있도록 하려면 관리자는 다음 역할 할당을 사용하여 개발자의 ID를 리소스에 추가해야 합니다.

[+] 테이블 확장

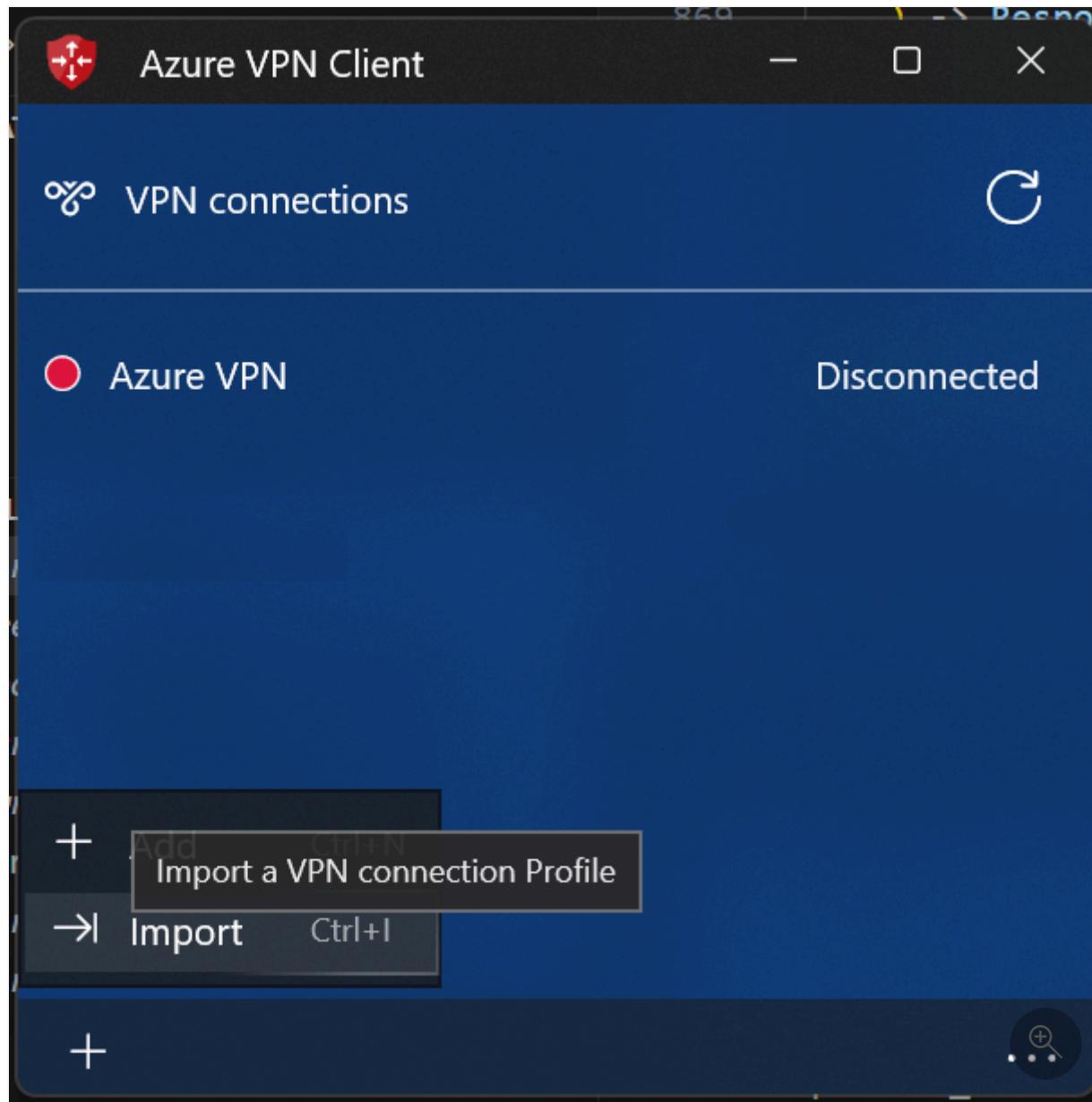
역할	리소스	설명
Cognitive Services Contributor	Azure OpenAI	Azure OpenAI Studio에서 공용 수집 API를 호출합니다. Contributor 역할만으로는 충분하지 않습니다. 왜냐하면 Contributor 역할만 있으면 Microsoft Entra ID 인증을 통해 데이터 평면 API를 호출할 수 없고 이 문서에 설명된 보안 설정에는 Microsoft Entra ID 인증이 필요하기 때문입니다.
Cognitive Services User	Azure OpenAI	Azure OpenAI Studio의 API 키를 나열합니다.
Contributor	Azure AI 검색	Azure OpenAI Studio의 인덱스를 나열하려면 API 키를 나열합니다.
Contributor	스토리지 계정	Azure OpenAI Studio에서 파일을 업로드하려면 계정 SAS를 나열합니다.
Contributor	개발자가 웹앱을 배포해야 하는 리소스 그룹 또는 Azure 구독	개발자의 Azure 구독에 웹앱을 배포합니다.

게이트웨이 및 클라이언트 구성

온-프레미스 클라이언트 컴퓨터에서 Azure OpenAI 서비스에 액세스하기 위한 방식 중 하나는 Azure VPN Gateway 및 Azure VPN Client를 구성하는 것입니다.

가상 네트워크용 가상 네트워크 게이트웨이를 만들려면 [이 지침](#)을 따릅니다.

지점 및 사이트 간 구성을 추가하고 Microsoft Entra ID 기반 인증을 사용하도록 설정하려면 [이 지침](#)을 따릅니다. Azure VPN Client 프로필 구성 패키지를 다운로드하고, 압축을 풀고, `AzureVPN/azurevpnconfig.xml` 파일을 Azure VPN Client로 가져옵니다.



리소스 호스트 이름이 가상 네트워크의 개인 IP를 가리키도록 로컬 컴퓨터 `hosts` 파일을 구성합니다. `hosts` 파일은 Windows의 경우 `C:\Windows\System32\drivers\etc`에 있고 Linux의 경우 `/etc/hosts`에 있습니다. 예시:

```
10.0.0.5 contoso.openai.azure.com
10.0.0.6 contoso.search.windows.net
```

Azure OpenAI Studio

온-프레미스 클라이언트 컴퓨터에서 수집 및 유추를 포함한 모든 Azure OpenAI Studio 기능을 사용할 수 있어야 합니다.

웹 앱

웹 앱은 Azure OpenAI 리소스와 통신합니다. Azure OpenAI 리소스에는 공용 네트워크가 사용하지 않도록 설정되어 있으므로 Azure OpenAI 리소스에 액세스하려면 가상 네트워크의 프라이빗 엔드포인트를 사용하도록 웹 앱을 설정해야 합니다.

웹 앱은 Azure OpenAI 호스트 이름을 Azure OpenAI용 프라이빗 엔드포인트의 개인 IP로 확인해야 합니다. 따라서 먼저 가상 네트워크에 대한 프라이빗 DNS 영역을 구성해야 합니다.

- 리소스 그룹에 [프라이빗 DNS 영역을 만듭니다.](#)
- [DNS 레코드를 추가합니다.](#) IP는 Azure OpenAI 리소스에 대한 프라이빗 엔드포인트의 개인 IP이며, Azure OpenAI에 대한 프라이빗 엔드포인트와 연결된 네트워크 인터페이스에서 IP 주소를 가져올 수 있습니다.
- [프라이빗 DNS 영역을 가상 네트워크에 연결하면](#) 이 가상 네트워크에 통합된 웹 앱이 이 프라이빗 DNS 영역을 사용할 수 있습니다.

Azure OpenAI Studio에서 웹 앱을 배포할 때 가상 네트워크와 동일한 위치를 선택하고 적절한 SKU를 선택하면 [가상 네트워크 통합 기능](#)을 지원할 수 있습니다.

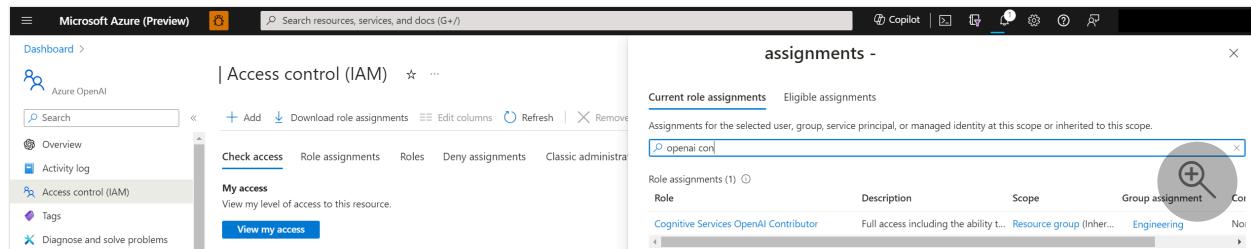
웹 앱이 배포된 후 Azure Portal 네트워킹 탭에서 웹 앱 아웃바운드 트래픽 가상 네트워크 통합을 구성하고 웹 앱용으로 예약한 세 번째 서브넷을 선택합니다.

The screenshot shows the Azure Portal interface for managing a web application named "webapp". The "Networking" tab is currently selected. At the top, there is a search bar with the text "net". Below the search bar, there are several navigation links: "Refresh", "Troubleshoot", and "Send us your feedback". The main content area displays the "Outbound traffic configuration" settings, which include "Virtual network integration" set to "vnet/webapp" and "Hybrid connections" set to "Not configured". On the left side, there is a sidebar with links for "Settings", "Environment variables", "Configuration", and "Networking". The "Networking" link is highlighted with a grey background.

API 사용

로그인 자격 증명에 Azure OpenAI 리소스에 대한 Cognitive Services OpenAI

Contributor 역할이 있는지 확인하고 먼저 az login을 실행합니다.



수집 API

수집 API에서 사용하는 요청 및 응답 개체에 대한 자세한 내용은 [수집 API 참조 문서](#)를 참조하세요.

추가 참고 사항:

- API 경로의 `JOB_NAME`은 Azure AI 검색에서 인덱스 이름으로 사용됩니다.
- api-key 대신 `Authorization` 헤더를 사용합니다.
- `storageEndpoint` 헤더를 명시적으로 설정합니다.
- `storageConnectionString` 헤더에 `ResourceId=` 형식을 사용하므로 Azure OpenAI 및 Azure AI 검색은 관리 ID를 사용하여 네트워크 제한을 무시하는 데 필요한 스토리지 계정을 인증합니다.
- `searchServiceAdminKey` 헤더를 설정하지 **마세요**. Azure OpenAI 리소스의 시스템 할당 ID는 Azure AI 검색을 인증하는 데 사용됩니다.
- `embeddingEndpoint` 또는 `embeddingKey`를 설정하지 **마세요**. 대신 `embeddingDeploymentName` 헤더를 사용하여 텍스트 벡터화를 사용하도록 설정합니다.

작업 제출 예

```
Bash

accessToken=$(az account get-access-token --resource
https://cognitiveservices.azure.com/ --query "accessToken" --output tsv)
curl -i -X PUT https://my-resource.openai.azure.com/openai/extensions/on-
your-data/ingestion-jobs/vpn1025a?api-version=2023-10-01-preview \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $accessToken" \
-H "storageEndpoint: https://mystorage.blob.core.windows.net/" \
-H "storageConnectionString: ResourceId=/subscriptions/1234567-abcd-1234-
5678-1234abcd/resourceGroups/my-
resource/providers/Microsoft.Storage/storageAccounts/mystorage" \
-H "storageContainer: my-container" \
-H "searchServiceEndpoint: https://mysearch.search.windows.net" \
-H "embeddingDeploymentName: ada" \
```

```
-d \
'
{
}
'
```

작업 상태 가져오기 예

Bash

```
accessToken=$(az account get-access-token --resource
https://cognitiveservices.azure.com/ --query "accessToken" --output tsv)
curl -i -X GET https://my-resource.openai.azure.com/openai/extensions/on-
your-data/ingestion-jobs/abc1234?api-version=2023-10-01-preview \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $accessToken"
```

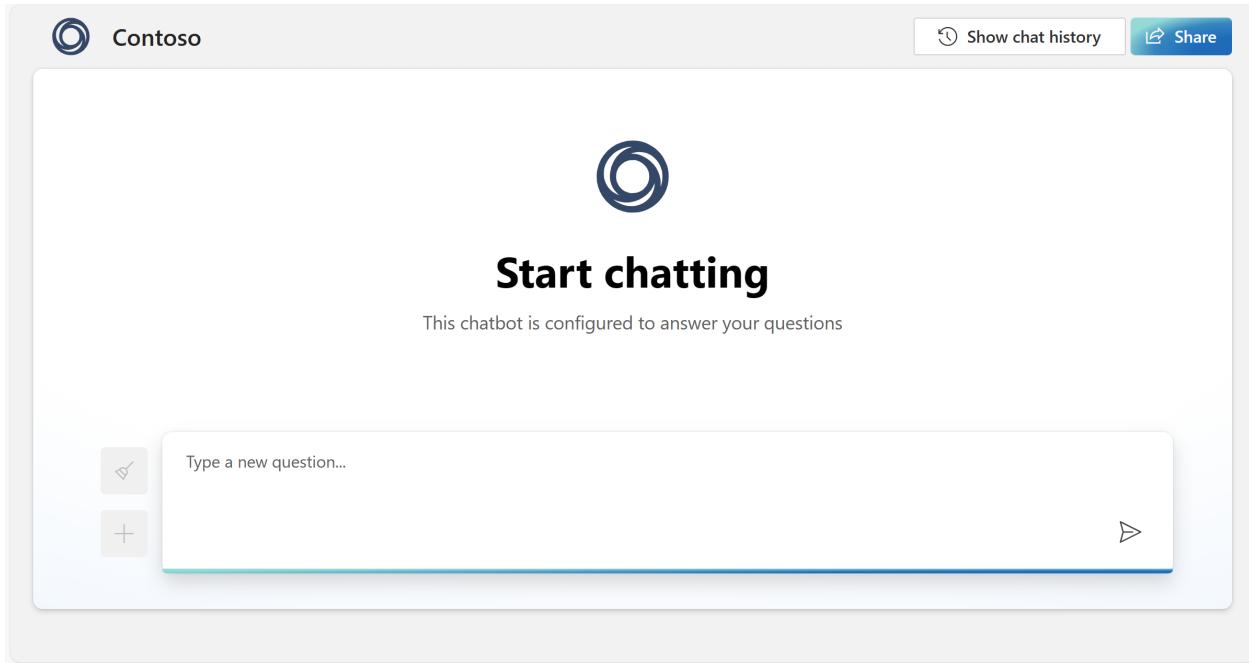
유추 API

유추 API에서 사용하는 요청 및 응답 개체에 대한 자세한 내용은 [유추 API 참조 문서](#)를 참조하세요.

Azure OpenAI 웹앱 사용

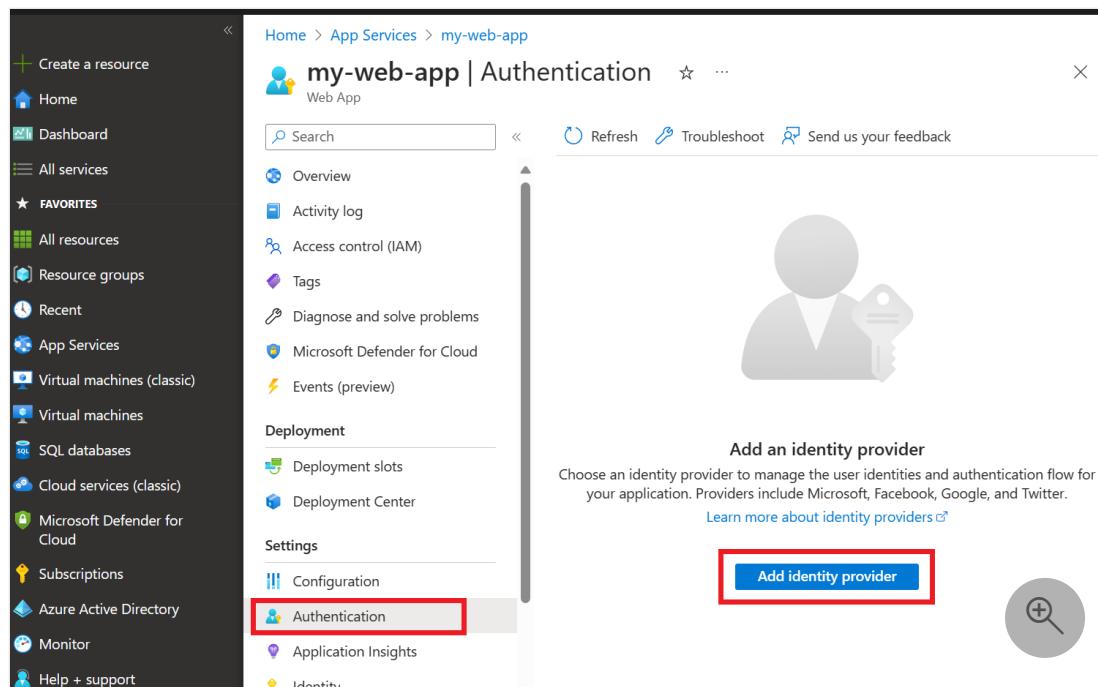
아티클 • 2024. 02. 28.

Azure OpenAI Studio, API 및 SDK와 함께 사용 가능한 독립 실행형 웹앱을 사용하여 Azure OpenAI 스튜디오 또는 [수동 배포](#)를 사용하여 배포할 수 있는 그래픽 사용자 인터페이스를 사용하여 Azure OpenAI 모델과 상호 작용할 수도 있습니다.



중요 사항

- 게시하면 구독에 Azure App Service가 만들어집니다. 선택한 [계획](#)에 따라 비용이 발생할 수 있습니다. 앱 사용이 완료되면 Azure Portal에서 삭제할 수 있습니다.
- 기본적으로 앱은 이미 구성된 Microsoft ID 공급자와 함께 배포되어 앱에 대한 액세스를 Azure 테넌트 멤버로 제한합니다. 인증을 추가하거나 수정하려면 다음을 수행 합니다.
 1. [Azure Portal](#)로 이동하여 게시 중에 지정한 앱 이름을 검색합니다. 웹앱을 선택하고 왼쪽 탐색 메뉴에서 인증 탭으로 이동합니다. 그런 다음 ID 공급자 추가를 선택합니다.



2. Microsoft를 ID 공급자로 선택합니다. 이 페이지의 기본 설정은 앱을 테넌트로만 제한하므로 여기에서 다른 항목을 변경할 필요가 없습니다. 그런 다음 **추가**를 선택합니다.

이제 사용자에게 앱에 액세스할 수 있도록 Microsoft Entra ID 계정으로 로그인하라는 메시지가 표시됩니다. 원하는 경우 유사한 프로세스에 따라 다른 ID 공급자를 추가할 수 있습니다. 앱은 사용자가 테넌트의 멤버인지 확인하는 것 이외의 다른 방법으로 사용자의 로그인 정보를 사용하지 않습니다.

웹앱 사용자 지정

앱의 프런트 엔드 및 백 엔드 논리를 사용자 지정할 수 있습니다. 앱은 앱의 아이콘 변경과 같은 일반적인 사용자 지정 시나리오에 대한 몇 가지 환경 변수를 제공합니다. 웹앱의 소스 코드와 자세한 내용은 [GitHub](#)을 참조하세요.

앱을 사용자 지정할 때 다음을 권장합니다.

- 사용자가 설정을 변경하면 채팅 세션을 초기화합니다(채팅 지우기). 사용자에게 채팅 기록이 손실된다는 사실을 알립니다.
- 구현하는 각 설정이 사용자 환경에 미치는 영향을 명확하게 전달합니다.
- Azure OpenAI 또는 Azure AI Search 리소스에 대한 API 키를 회전하는 경우 새 키를 사용하도록 배포된 각 앱에 대한 앱 설정을 업데이트해야 합니다.

웹앱에 대한 샘플 소스 코드는 [GitHub](#)에서 사용할 수 있습니다. 소스 코드는 "있는 그대로" 샘플로만 제공됩니다. 고객은 웹앱의 모든 사용자 지정 및 구현을 담당합니다.

웹앱 업데이트

웹앱의 소스 코드에 대한 분기에서 `main` 변경 내용을 자주 끌어와 최신 버그 수정, API 버전 및 개선 사항이 있는지 확인하는 것이 좋습니다.

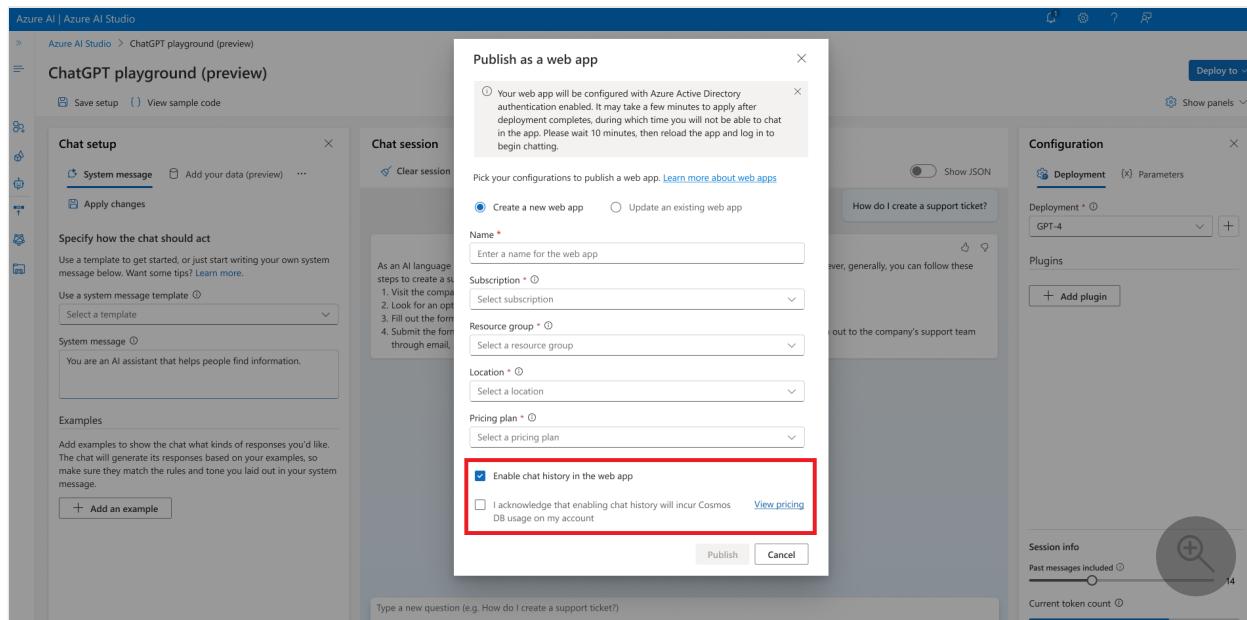
① 참고

2024년 2월 1일 이후에는 웹앱에서 앱 시작 명령을 로 설정해야 합니다 `python3 -m gunicorn app:app`. 2024년 2월 1일 이전에 게시된 앱을 업데이트할 때 App Service 구성 페이지에서 시작 명령을 수동으로 추가해야 합니다.

채팅 기록

웹앱 사용자에 대해 채팅 기록을 사용하도록 설정할 수 있습니다. 이 기능을 사용하도록 설정하면 사용자는 개별 이전 쿼리 및 응답에 액세스할 수 있습니다.

채팅 기록을 사용하도록 설정하려면 [Azure OpenAI Studio](#)를 사용하여 모델을 웹앱으로 배포하거나 다시 배포합니다.

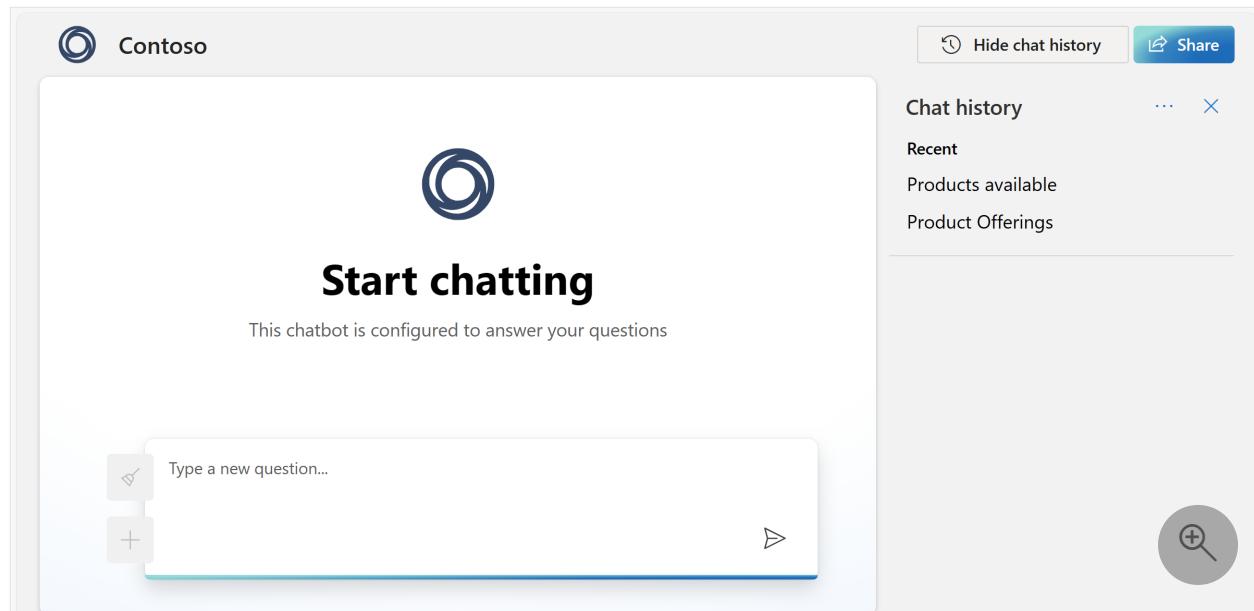


① 중요

채팅 기록을 사용하도록 설정하면 리소스 그룹에 [Cosmos DB](#) 인스턴스가 생성되고 사용된 스토리지에 대한 [추가 요금](#)이 발생합니다.

채팅 기록을 사용하도록 설정하면 앱의 오른쪽 위 모서리에서 채팅 기록을 표시 및 숨길 수 있습니다. 기록이 표시되면 대화의 이름을 바꾸거나 삭제할 수 있습니다. 앱에 로그인

하면 대화가 자동으로 최신에서 가장 오래된 것까지 정렬되고 대화의 첫 번째 쿼리에 따라 이름이 지정됩니다.



Cosmos DB 인스턴스 삭제

웹앱을 삭제해도 Cosmos DB 인스턴스가 자동으로 삭제되지는 않습니다. 모든 저장된 채팅과 함께 Cosmos DB 인스턴스를 삭제하려면 [Azure Portal](#)에서 연결된 리소스로 이동한 후 삭제해야 합니다. Cosmos DB 리소스를 삭제하지만 스튜디오에서 채팅 기록 옵션을 사용하도록 설정한 상태로 유지하면 연결 오류 알림이 표시되지만 채팅 기록에 액세스하지 않고 웹앱을 계속 사용할 수 있습니다.

다음 단계

- [신속한 엔지니어링](#)
- [데이터에 대한 Azure openAI](#)

Use the Azure Developer CLI to deploy resources for Azure OpenAI On Your Data

Article • 04/11/2024

Use this article to learn how to automate resource deployment for Azure OpenAI On Your Data. The Azure Developer CLI (`azd`) is an open-source, command-line tool that streamlines provisioning and deploying resources to Azure using a template system. The template contains infrastructure files to provision the necessary Azure OpenAI resources and configurations and includes the completed sample app code.

Prerequisites

- An Azure subscription - [Create one for free ↗](#).
- Access granted to Azure OpenAI in the desired Azure subscription.

Azure OpenAI requires registration and is currently only available to approved enterprise customers and partners. [See Limited access to Azure OpenAI Service](#) for more information. You can apply for access to Azure OpenAI by completing the form at <https://aka.ms/oai/access> ↗. Open an issue on this repo to contact us if you have an issue.

- The Azure Developer CLI [installed](#) on your machine

Clone and initialize the Azure Developer CLI template

1. For the steps ahead, clone and initialize the template.

```
Bash
```

```
azd init --template openai-chat-your-own-data
```

2. The `azd init` command prompts you for the following information:

- Environment name: This value is used as a prefix for all Azure resources created by Azure Developer CLI. The name must be unique across all Azure

subscriptions and must be between 3 and 24 characters long. The name can contain numbers and lowercase letters only.

Use the template to deploy resources

1. Sign-in to Azure:

```
Bash  
azd auth login
```

2. Provision and deploy the OpenAI resource to Azure:

```
Bash  
azd up
```

`azd` prompts you for the following information:

- Subscription: The Azure subscription that your resources are deployed to.
- Location: The Azure region where your resources are deployed.

ⓘ Note

The sample `azd` template uses the `gpt-35-turbo-16k` model. A recommended region for this template is East US, since different Azure regions support different OpenAI models. You can visit the [Azure OpenAI Service Models](#) support page for more details about model support by region.

ⓘ Note

The provisioning process may take several minutes to complete. Wait for the task to finish before you proceed to the next steps.

3. Click the link `azd` outputs to navigate to the new resource group in the Azure portal. You should see the following top level resources:

- An Azure OpenAI service with a deployed model
- An Azure Storage account you can use to upload your own data files
- An Azure AI Search service configured with the proper indexes and data sources

Upload data to the storage account

`azd` provisioned all of the required resources for you to chat with your own data, but you still need to upload the data files you want to make available to your AI service.

1. Navigate to the new storage account in the Azure portal.
2. On the left navigation, select **Storage browser**.
3. Select **Blob containers** and then navigate into the **File uploads** container.
4. Click the **Upload** button at the top of the screen.
5. In the flyout menu that opens, upload your data.

Note

The search indexer is set to run every 5 minutes to index the data in the storage account. You can either wait a few minutes for the uploaded data to be indexed, or you can manually run the indexer from the search service page.

Connect or create an application

After running the `azd` template and uploading your data, you're ready to start using Azure OpenAI on Your Data. See the [quickstart article](#) for code samples you can use to build your applications.

OpenAI Python API 라이브러리 1.x로 마이그레이션

아티클 • 2024. 02. 26.

OpenAI는 최근 [OpenAI Python API 라이브러리](#)의 새 버전을 릴리스했습니다. 이 가이드는 [OpenAI 마이그레이션 가이드](#)를 보완하며 Azure OpenAI와 관련된 변경 내용을 빠르게 파악하는 데 도움이 됩니다.

업데이트

- OpenAI Python API 라이브러리의 새 버전입니다.
- 2023년 11월 6일부터 `pip install openai` 및 `pip install openai --upgrade`는 OpenAI Python 라이브러리의 `version 1.x`를 설치합니다.
- `version 0.28.1`에서 `version 1.x`로 업그레이드하는 것은 호환성이 손상되는 변경이므로 코드를 테스트하고 업데이트해야 합니다.
- 오류가 있는 경우 백오프로 자동 다시 시도
- 적절한 형식(mypy/pyright/editors용)
- 이제 전역 기본값을 사용하는 대신 클라이언트를 인스턴스화할 수 있습니다.
- 명시적 클라이언트 인스턴스화로 전환
- [이름 변경](#)

알려진 문제

- `DALL-E3`은 [최신 1.x 릴리스에서 완벽하게 지원됩니다](#). `DALL-E2`는 [코드를 다음과 같이 이 수정](#)하여 1.x와 함께 사용할 수 있습니다.
- 의미 체계 텍스트 검색을 위한 코사인 유사성과 같은 기능을 제공하는 데 사용된 `embeddings_utils.py`는 [더 이상 OpenAI Python API 라이브러리의 일부가 아닙니다](#).
- OpenAI Python 라이브러리에 대해 활성화된 [GitHub 문제](#)도 확인해야 합니다.

마이그레이션하기 전에 테스트

ⓘ 중요

Azure OpenAI에서는 `openai migrate`를 사용한 코드 자동 마이그레이션이 지원되지 않습니다.

이는 호환성이 손상되는 변경이 포함된 새 버전의 라이브러리이므로 버전 1.x를 사용하도록 프로덕션 애플리케이션을 마이그레이션하기 전에 새 릴리스에 대해 코드를 광범위하게 테스트해야 합니다. 또한 코드와 내부 프로세스를 검토하여 모범 사례를 따르고 완전히 테스트한 버전에만 프로덕션 코드를 고정하고 있는지 확인해야 합니다.

マイグレートプロセスをより簡単に作成するため、Python用ドキュメントの基準コード例をタブ環境で更新しています。

OpenAI Python 0.28.1

콘솔

```
pip install openai==0.28.1
```

이를 통해 변경된 사항에 대한 컨텍스트를 제공하고 버전 0.28.1에 대한 지원을 계속 제공하면서 새 라이브러리를 병렬로 테스트할 수 있습니다. 1.x로 업그레이드하고 일시적으로 이전 버전으로 되돌려야 한다는 것을 깨닫는 경우 언제든지 pip uninstall openai 한 다음 pip install openai==0.28.1을 사용하여 0.28.1 대상으로 다시 설치할 수 있습니다.

채팅 완료

OpenAI Python 0.28.1

GPT-35-Turbo 또는 GPT-4 모델을 배포할 때 선택한 배포 이름으로 engine 변수를 설정해야 합니다. 기본 모델 이름과 동일한 배포 이름을 선택하지 않으면 모델 이름을 입력할 때 오류가 발생합니다.

Python

```
import os
import openai
openai.api_type = "azure"
openai.api_base = os.getenv("AZURE_OPENAI_ENDPOINT")
openai.api_key = os.getenv("AZURE_OPENAI_API_KEY")
openai.api_version = "2023-05-15"

response = openai.ChatCompletion.create(
    engine="gpt-35-turbo", # engine = "deployment_name".
    messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Does Azure OpenAI support customer managed keys?"},
```

```
        {"role": "assistant", "content": "Yes, customer managed keys are supported by Azure OpenAI."},  
        {"role": "user", "content": "Do other Azure AI services support this too?"}  
    ]  
)  
  
print(response)  
print(response['choices'][0]['message']['content'])
```

완성

OpenAI Python 0.28.1

Python

```
import os  
import openai  
  
openai.api_key = os.getenv("AZURE_OPENAI_API_KEY")  
openai.api_base = os.getenv("AZURE_OPENAI_ENDPOINT") # your endpoint  
should look like the following  
https://YOUR_RESOURCE_NAME.openai.azure.com/  
openai.api_type = 'azure'  
openai.api_version = '2023-05-15' # this might change in the future  
  
deployment_name='REPLACE_WITH_YOUR_DEPLOYMENT_NAME' #This will  
correspond to the custom name you chose for your deployment when you  
deployed a model.  
  
# Send a completion call to generate an answer  
print('Sending a test completion job')  
start_phrase = 'Write a tagline for an ice cream shop. '  
response = openai.Completion.create(engine=deployment_name,  
prompt=start_phrase, max_tokens=10)  
text = response['choices'][0]['text'].replace('\n', '').replace('. .',  
'.').strip()  
print(start_phrase+text)
```

포함

OpenAI Python 0.28.1

Python

```

import openai

openai.api_type = "azure"
openai.api_key = YOUR_API_KEY
openai.api_base = "https://YOUR_RESOURCE_NAME.openai.azure.com"
openai.api_version = "2023-05-15"

response = openai.Embedding.create(
    input="Your text string goes here",
    engine="YOUR_DEPLOYMENT_NAME"
)
embeddings = response[ 'data'][0][ 'embedding']
print(embeddings)

```

Async

OpenAI는 모듈 수준 클라이언트에서 비동기 메서드 호출을 지원하지 않습니다. 대신 비동기 클라이언트를 인스턴스화해야 합니다.

Python

```

import os
import asyncio
from openai import AsyncAzureOpenAI

async def main():
    client = AsyncAzureOpenAI(
        api_key = os.getenv("AZURE_OPENAI_API_KEY"),
        api_version = "2023-12-01-preview",
        azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
    )
    response = await client.chat.completions.create(model="gpt-35-turbo",
messages=[{"role": "user", "content": "Hello world"}])

    print(response.model_dump_json(indent=2))

asyncio.run(main())

```

인증

Python

```

from azure.identity import DefaultAzureCredential, get_bearer_token_provider
from openai import AzureOpenAI

token_provider = get_bearer_token_provider(DefaultAzureCredential(),

```

```

"https://cognitiveservices.azure.com/.default")

api_version = "2023-12-01-preview"
endpoint = "https://my-resource.openai.azure.com"

client = AzureOpenAI(
    api_version=api_version,
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
)

completion = client.chat.completions.create(
    model="deployment-name", # gpt-35-instant
    messages=[
        {
            "role": "user",
            "content": "How do I output all files in a directory using
Python?",
        },
    ],
)
print(completion.model_dump_json(indent=2))

```

데이터 사용

이러한 코드 예제가 작동하도록 하는 데 필요한 전체 구성 단계는 데이터 사용 빠른 시작을 [참조하세요](#).

OpenAI Python 0.28.1

Python

```

import os
import openai
import dotenv
import requests

dotenv.load_dotenv()

openai.api_base = os.environ.get("AZURE_OPENAI_ENDPOINT")
openai.api_version = "2023-08-01-preview"
openai.api_type = 'azure'
openai.api_key = os.environ.get("AZURE_OPENAI_API_KEY")

def setup_byod(deployment_id: str) -> None:
    """Sets up the OpenAI Python SDK to use your own data for the chat
    endpoint.

    :param deployment_id: The deployment ID for the model to use with
    your own data.

```

```

To remove this configuration, simply set openai.requestssession to
None.

"""

class BringYourOwnDataAdapter(requests.adapters.HTTPAdapter):

    def send(self, request, **kwargs):
        request.url = f"
{openai.api_base}/openai/deployments/{deployment_id}/extensions/chat/com
pletions?api-version={openai.api_version}"
        return super().send(request, **kwargs)

session = requests.Session()

# Mount a custom adapter which will use the extensions endpoint for
any call using the given `deployment_id`
session.mount(
    prefix=f"{openai.api_base}/openai/deployments/{deployment_id}",
    adapter=BringYourOwnDataAdapter()
)

openai.requestssession = session

aoai_deployment_id = os.environ.get("AZURE_OPEN_AI_DEPLOYMENT_ID")
setup_byod(aoai_deployment_id)

completion = openai.ChatCompletion.create(
    messages=[{"role": "user", "content": "What are the differences
between Azure Machine Learning and Azure AI services?"}],
    deployment_id=os.environ.get("AZURE_OPEN_AI_DEPLOYMENT_ID"),
    dataSources=[ # camelCase is intentional, as this is the format the
API expects
    {
        "type": "AzureCognitiveSearch",
        "parameters": {
            "endpoint": os.environ.get("AZURE_AI_SEARCH_ENDPOINT"),
            "key": os.environ.get("AZURE_AI_SEARCH_API_KEY"),
            "indexName": os.environ.get("AZURE_AI_SEARCH_INDEX"),
        }
    }
]
)
print(completion)

```

DALL-E 퍽스

DALLE-Fix

Python

```
import time
import json
import httpx
import openai

class CustomHTTPTransport(httpx.HTTPTransport):
    def handle_request(
        self,
        request: httpx.Request,
    ) -> httpx.Response:
        if "images/generations" in request.url.path and
request.url.params[
            "api-version"
        ] in [
            "2023-06-01-preview",
            "2023-07-01-preview",
            "2023-08-01-preview",
            "2023-09-01-preview",
            "2023-10-01-preview",
        ]:
            request.url =
request.url.copy_with(path="/openai/images/generations:submit")
            response = super().handle_request(request)
            operation_location_url = response.headers["operation-
location"]
            request.url = httpx.URL(operation_location_url)
            request.method = "GET"
            response = super().handle_request(request)
            response.read()

            timeout_secs: int = 120
            start_time = time.time()
            while response.json()["status"] not in ["succeeded",
"failed"]:
                if time.time() - start_time > timeout_secs:
                    timeout = {"error": {"code": "Timeout", "message":
"Operation polling timed out."}}
                    return httpx.Response(
                        status_code=400,
                        headers=response.headers,
                        content=json.dumps(timeout).encode("utf-8"),
                        request=request,
                    )

                time.sleep(int(response.headers.get("retry-after")) or
10)
                response = super().handle_request(request)
                response.read()

                if response.json()["status"] == "failed":
                    error_data = response.json()
                    return httpx.Response(
                        status_code=400,
```

```

        headers=response.headers,
        content=json.dumps(error_data).encode("utf-8"),
        request=request,
    )

    result = response.json()["result"]
    return httpx.Response(
        status_code=200,
        headers=response.headers,
        content=json.dumps(result).encode("utf-8"),
        request=request,
    )
return super().handle_request(request)

client = openai.AzureOpenAI(
    azure_endpoint="",
    api_key="",
    api_version="",
    http_client=httpx.Client(
        transport=CustomHTTPTransport(),
    ),
)
image = client.images.generate(prompt="a cute baby seal")

print(image.data[0].url)

```

이름 변경

① 참고

모든 a* 메서드가 제거되었습니다. 대신 비동기 클라이언트를 사용해야 합니다.

 테이블 확장

OpenAI Python 0.28.1	OpenAI Python 1.x
openai.api_base	openai.base_url
openai.proxy	openai.proxies
openai.InvalidRequestError	openai.BadRequestError
openai.Audio.transcribe()	client.audio.transcriptions.create()
openai.Audio.translate()	client.audio.translations.create()

OpenAI Python 0.28.1	OpenAI Python 1.x
openai.ChatCompletion.create()	client.chat.completions.create()
openai.Completion.create()	client.completions.create()
openai.Edit.create()	client.edits.create()
openai.Embedding.create()	client.embeddings.create()
openai.File.create()	client.files.create()
openai.File.list()	client.files.list()
openai.File.retrieve()	client.files.retrieve()
openai.File.download()	client.files.retrieve_content()
openai.FineTune.cancel()	client.fine_tunes.cancel()
openai.FineTune.list()	client.fine_tunes.list()
openai.FineTune.list_events()	client.fine_tunes.list_events()
openai.FineTune.stream_events()	client.fine_tunes.list_events(stream=True)
openai.FineTune.retrieve()	client.fine_tunes.retrieve()
openai.FineTune.delete()	client.fine_tunes.delete()
openai.FineTune.create()	client.fine_tunes.create()
openai.FineTuningJob.create()	client.fine_tuning.jobs.create()
openai.FineTuningJob.cancel()	client.fine_tuning.jobs.cancel()
openai.FineTuningJob.delete()	client.fine_tuning.jobs.create()
openai.FineTuningJob.retrieve()	client.fine_tuning.jobs.retrieve()
openai.FineTuningJob.list()	client.fine_tuning.jobs.list()
openai.FineTuningJob.list_events()	client.fine_tuning.jobs.list_events()
openai.Image.create()	client.images.generate()
openai.Image.create_variation()	client.images.create_variation()
openai.Image.create_edit()	client.images.edit()
openai.Model.list()	client.models.list()
openai.Model.delete()	client.models.delete()

OpenAI Python 0.28.1	OpenAI Python 1.x
<code>openai.Model.retrieve()</code>	<code>client.models.retrieve()</code>
<code>openai.Moderation.create()</code>	<code>client.moderations.create()</code>
<code>openai.api_resources</code>	<code>openai.resources</code>

제거됨

- `openai.api_key_path`
- `openai.app_info`
- `openai.debug`
- `openai.log`
- `openai.OpenAIError`
- `openai.Audio.transcribe_raw()`
- `openai.Audio.translate_raw()`
- `openai.ErrorObject`
- `openai.Customer`
- `openai.api_version`
- `openai.verify_ssl_certs`
- `openai.api_type`
- `openai.enable_telemetry`
- `openai.ca_bundle_path`
- `openai.requestssession`(OpenAI는 이제 `httpx`를 사용함)
- `openai.aiosession`(OpenAI는 이제 `httpx`를 사용함)
- `openai.Deployment`(이전에는 Azure OpenAI에 사용됨)
- `openai.Engine`
- `openai.File.find_matching_files()`

Azure OpenAI 모델 작업

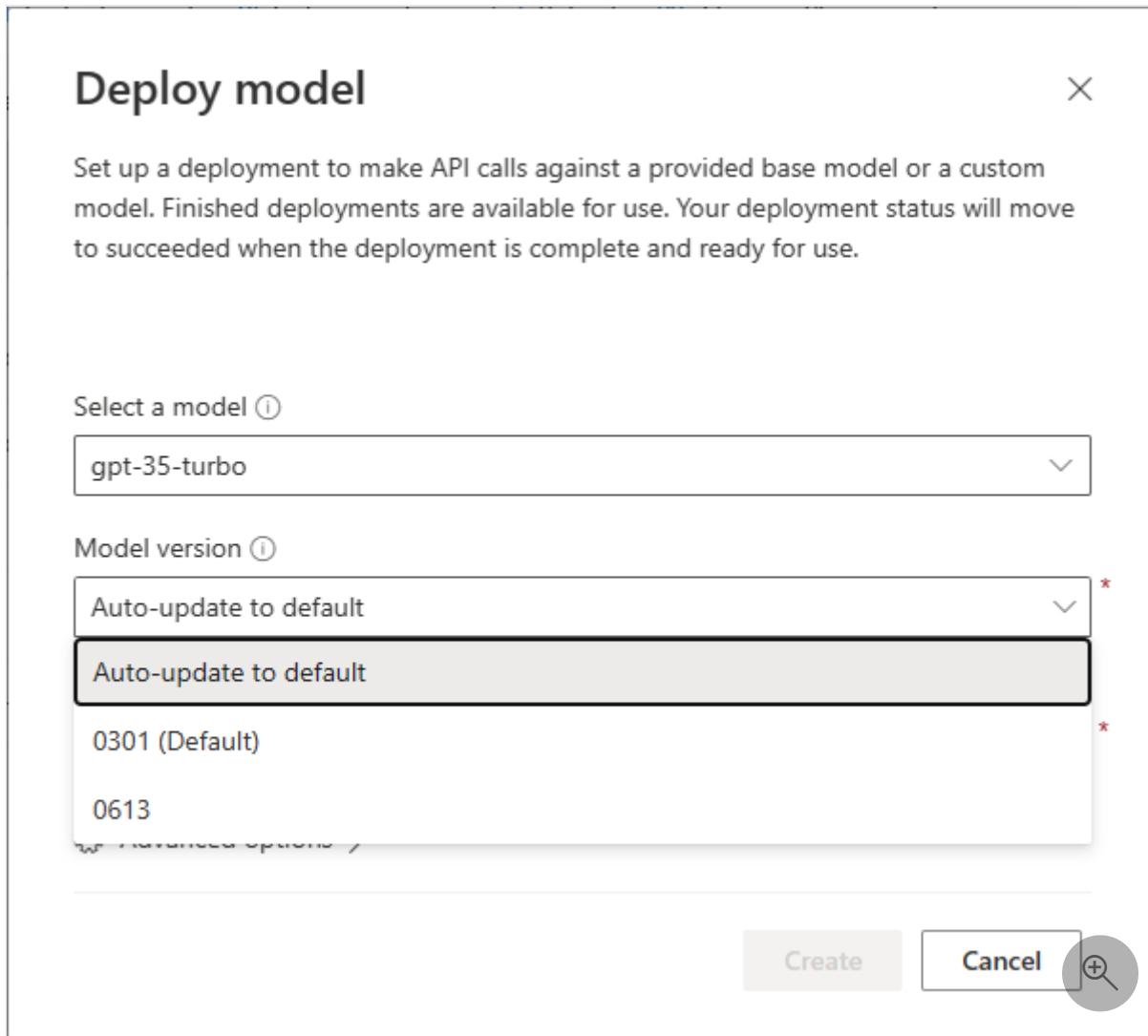
아티클 • 2024. 03. 14.

Azure OpenAI 서비스는 다양한 기능과 가격대를 갖춘 다양한 모델 집합으로 구동됩니다. 모델 가용성은 지역에 따라 다릅니다.

모델 목록 API를 사용하여 Azure OpenAI 리소스에서 유추 및 미세 조정에 사용할 수 있는 모델 목록을 가져올 수 있습니다.

모델 업데이트

이제 Azure OpenAI는 선택 모델 배포에 대한 자동 업데이트를 지원합니다. 자동 업데이트 지원을 사용할 수 있는 모델에서는 새 배포 만들기 및 배포 편집 아래의 Azure OpenAI Studio에 모델 버전 드롭다운이 표시됩니다.



Azure OpenAI 모델 버전 및 작동 방식은 [Azure OpenAI 모델 버전](#) 문서에서 자세히 알아볼 수 있습니다.

기본값으로 자동 업데이트

배포를 기본값으로 자동 업데이트로 설정하면 기본 버전이 변경된 후 2주 이내에 모델 배포가 자동으로 업데이트됩니다. 미리 보기 버전의 경우 새 미리 보기 버전이 릴리스된 후 2주 후에 새 미리 보기 버전을 사용할 수 있게 되면 자동으로 업데이트됩니다.

유추 모델에 대한 초기 테스트 단계에 있는 경우 가능할 때마다 **기본값으로 자동 업데이트**가 설정된 모델을 배포하는 것이 좋습니다.

특정 모델 버전

Azure OpenAI 사용이 진화하고 애플리케이션을 빌드하고 통합하기 시작하면 모델 업데이트를 수동으로 제어할 수 있습니다. 먼저 업그레이드하기 전에 애플리케이션 동작이 사용 사례에 대해 일관된지 테스트하고 유효성을 검사할 수 있습니다.

배포에 대한 특정 모델 버전을 선택하면 수동으로 업데이트하도록 선택하거나 모델의 사용 중지 날짜에 도달할 때까지 이 버전을 다시 기본 선택합니다. 사용 중지 날짜에 도달하면 모델은 사용 중지 시점에 기본 버전으로 자동 업그레이드됩니다.

사용 중지 날짜 보기

현재 배포된 모델의 경우 Azure OpenAI Studio에서 **배포**를 선택합니다.

The screenshot shows the Azure OpenAI Studio interface with the 'Deployments' section selected. It displays a table of deployed models with columns for Deployment name, Model name, Model version, Deployment type, Capacity, Status, and Model deprecation date. A red box highlights the 'Model deprecation ...' column, which lists dates like 2/1/2025, 2/29/2024, 9/30/2023, 7/10/2024, and 9/29/2024. To the right of the table is a sidebar with 'Content Filter' and 'Rate limit (Tokens per...)' settings, and a search bar.

Deployment name	Model name	Model version	Deployme...	Capacity	Status	Model deprecation ...	Content Filter	Rate limit (Tokens pe...
text-embedding-ada-002	text-embedding-ada-002	1	Standard	120K TPM	Succeeded	2/1/2025	Default	120000
text-ada-001	text-ada-001	1	Standard	120K TPM	Succeeded	2/29/2024	Default	120000
gpt-35-turbo	gpt-35-turbo	0301	Standard	120K TPM	Succeeded	9/30/2023	Default	120000
<input checked="" type="checkbox"/> code-davinci-002	code-davinci-002	1	Standard	120K TPM	Succeeded	7/10/2024	Default	120000
text-davinci-003	text-davinci-003	1	Standard	60K TPM	Succeeded	9/29/2024	Default	60000

Azure OpenAI Studio에서 지정된 지역의 사용 가능한 모든 모델에 대한 사용 중지 날짜를 보려면 모델>열 옵션을> 선택하여 사용 중단 미세 조정 및 사용 중단 유추를 선택합니다.

Azure OpenAI Studio > Models

Models

Azure OpenAI is powered by models with different capabilities and price points. Deploy one of the provided base models to try it out in [Playground](#) or train a custom model to your specific use case and data for better performance and more accurate results. [Learn more about the different types of base models](#)

Base models

Deploy Create a custom model Column options Refresh Search

Model name	Model version	Created at	Status	Deployable	Deprecation fine tune	Deprecation inference
code-davinci-002	1	7/10/2022 8:00 PM	Succeeded	No	-	7/10/2024 8:00 PM
gpt-35-turbo	0301	3/8/2023 7:00 PM	Succeeded	No	-	9/30/2023 8:00 PM
text-ada-001	1	2/28/2022 7:00 PM	Succeeded	No	2/29/2024 7:00 PM	2/29/2024 7:00 PM
text-babbage-001	1	2/28/2022 7:00 PM	Succeeded	Yes	2/29/2024 7:00 PM	2/29/2024 7:00 PM
text-curious-001	1	2/28/2022 7:00 PM	Succeeded	Yes	2/29/2024 7:00 PM	2/29/2024 7:00 PM
text-davinci-002	1	1/21/2022 7:00 PM	Succeeded	Yes	-	1/14/2024 7:00 PM
text-davinci-003	1	9/29/2022 8:00 PM	Succeeded	No	-	9/29/2024 8:00 PM
text-embedding-ada-002	2	4/2/2023 8:00 PM	Succeeded	Yes	-	4/2/2025 8:00 PM
text-embedding-ada-002	1	2/1/2023 7:00 PM	Succeeded	No	-	2/1/2025 7:00 PM
text-similarity-ada-001	1	5/19/2022 8:00 PM	Succeeded	Yes	-	5/19/2024 8:00 PM
text-similarity-curie-001	1	5/19/2022 8:00 PM	Succeeded	Yes	-	5/19/2024 8:00 PM

모델 배포 업그레이드 구성

Azure OpenAI Studio [Azure OpenAI Studio](#)에서 이전에 배포된 모델에 대해 설정된 모델 업그레이드 옵션을 확인할 수 있습니다. **배포**>를 선택하고 배포 이름 열 아래에서 파란색으로 강조 표시된 배포 이름 중 하나를 선택합니다.

Azure AI | Azure OpenAI Studio

Azure AI Studio > Deployments

Deployments

Deployments provide endpoints to the Azure OpenAI base models, or your fine-tuned models, configured with settings to meet your needs. You can view your deployments, edit them, and create new deployments here.

Create new deployment Edit deployment Delete deployment Column options Refresh Open

Deployment name	Model name	Model version	Deployment type	Capacity	Status
gpt-35-turbo	gpt-35-turbo	0301	Standard	80K TPM	Succeeded

배포 이름을 선택하면 모델 배포의 속성이 열립니다. **버전 업데이트 정책**에서 배포에 대해 설정된 업그레이드 옵션을 볼 수 있습니다.

gpt-35-turbo

[Edit deployment](#) [Delete deployment](#) [Refresh](#) [Open in Playground](#)

Status: Deployment succeeded

Created by: docs@contoso.com
Created at: 7/31/2023 12:45 PM
Last updated by: docs@contoso.com
Last updated at: 10/31/2023 9:59 AM

Properties:

Model name: gpt-35-turbo
Model version: 0301
Version update policy: Once a new default version is available.
Deployment type: Standard
Content Filter: Default
Tokens per Minute Rate Limit (thousands): 80
Rate limit (Tokens per minute): 80000
Rate limit (Requests per minute): 480



해당 속성은 REST, Azure PowerShell 및 Azure CLI를 통해 액세스할 수도 있습니다.

[] 테이블 확장

옵션	읽음	업데이트
REST	예. <code>versionUpgradeOption</code> 반환되지 않으면 <code>null</code>	예
Azure PowerShell	예. <code>VersionUpgradeOption</code> 이 <code>\$null</code> 인지 확인 할 수 있습니다.	예
Azure CLI	예. <code>versionUpgradeOption</code> 이 설정되지 않으면 <code>null</code> 이 표시됩니다.	아니요. 현재 버전 업그레이드 옵션을 업데이트할 수 없습니다.

세 가지 고유한 모델 배포 업그레이드 옵션이 있습니다.

[] 테이블 확장

이름	설명
<code>OnceNewDefaultVersionAvailable</code>	새 버전이 기본값으로 지정되면 모델 배포는 해당 지정이 변경된 후 2주 이내에 자동으로 기본 버전으로 업그레이드됩니다.
<code>OnceCurrentVersionExpired</code>	사용 중지 날짜에 도달하면 모델 배포가 자동으로 현재 기본 버전으로 업그레이드됩니다.
<code>NoAutoUpgrade</code>	모델 배포는 자동으로 업그레이드되지 않습니다. 사용 중지 날짜에 도달하면 모델 배포가 작동을 중지합니다. 존재하지 않는 모델 배포를 가리키도록 해당 배포를 참조하는 코드를 업데이트해야 합니다.

① 참고

`null` 는 `AutoUpgradeWhenExpired`와 같습니다. 모델의 속성에 모델 업그레이드를 지원하는 **버전 업데이트 정책** 옵션이 없으면 값이 현재 `null` 인 것입니다. 이 값을 명시적으로 수정하면 REST API뿐만 아니라 스튜디오 속성 페이지에도 속성이 표시됩니다.

예제

PowerShell

Azure PowerShell [시작 가이드](#)를 검토하여 Azure PowerShell을 로컬로 설치하거나 [Azure Cloud Shell](#)을 사용할 수 있습니다.

아래 단계에서는 `VersionUpgradeOption` 옵션 속성을 확인하고 업데이트하는 방법을 보여 줍니다.

PowerShell

```
// Step 1: Get Deployment
$deployment = Get-AzCognitiveServicesAccountDeployment -ResourceGroupName {ResourceGroupName} -AccountName {AccountName} -Name {DeploymentName}

// Step 2: Show Deployment VersionUpgradeOption
$deployment.Properties.VersionUpgradeOption

// VersionUpgradeOption can be null - one way to check is
IsNull -eq $deployment.Properties.VersionUpgradeOption

// Step 3: Update Deployment VersionUpgradeOption
$deployment.Properties.VersionUpgradeOption = "NoAutoUpgrade"
New-AzCognitiveServicesAccountDeployment -ResourceGroupName {ResourceGroupName} -AccountName {AccountName} -Name {DeploymentName} -Properties $deployment.Properties -Sku $deployment.Sku

// repeat step 1 and 2 to confirm the change.
// If not sure about deployment name, use this command to show all
deployments under an account
Get-AzCognitiveServicesAccountDeployment -ResourceGroupName {ResourceGroupName} -AccountName {AccountName}
```

PowerShell

```
// To update to a new model version

// Step 1: Get Deployment
$deployment = Get-AzCognitiveServicesAccountDeployment -ResourceGroupName {ResourceGroupName} -AccountName {AccountName} -Name
```

```

{DeploymentName}

// Step 2: Show Deployment Model properties
$deployment.Properties.Model.Version

// Step 3: Update Deployed Model Version
$deployment.Properties.Model.Version = "0613"
New-AzCognitiveServicesAccountDeployment -ResourceGroupName
{ResourceGroupName} -AccountName {AccountName} -Name {DeploymentName} -
Properties $deployment.Properties -Sku $deployment.Sku

// repeat step 1 and 2 to confirm the change.

```

API를 통해 모델 업데이트 및 배포

HTTP

PUT

<https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.CognitiveServices/accounts/{accountName}/deployments/{deploymentName}?api-version=2023-05-01>

경로 매개 변수

[+] 테이블 확장

매개 변수	형식	필수 여부	설명
accountname	string	Required	Azure OpenAI 리소스의 이름입니다.
deploymentName	string	Required	기존 모델을 배포할 때 선택한 배포 이름 또는 새 모델 배포에 사용하려는 이름입니다.
resourceGroupName	string	Required	이 모델 배포에 연결된 리소스 그룹의 이름입니다.
subscriptionId	string	Required	연결된 구독의 구독 ID입니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-05-01 Swagger 사양 ↗

요청 본문

이는 사용할 수 있는 요청 본문 매개 변수의 하위 집합일 뿐입니다. 매개 변수의 전체 목록을 보려면 [REST API 참조 설명서](#)를 참조하세요.

테이블 확장

매개 변수	형식	설명
버전업그레이드 옵션	문자열	배포 모델 버전 업그레이드 옵션: <code>OnceNewDefaultVersionAvailable</code> <code>OnceCurrentVersionExpired</code> <code>NoAutoUpgrade</code>
capacity	정수	이는 이 배포에 할당하는 활당량 의 양을 나타냅니다. 값 1은 분당 토큰 (TPM) 1,000개와 같습니다.

예제 요청

Bash

```
curl -X PUT https://management.azure.com/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/resource-group-temp/providers/Microsoft.CognitiveServices/accounts/docs-openai-test-001/deployments/gpt-35-turbo?api-version=2023-05-01 \
-H "Content-Type: application/json" \
-H 'Authorization: Bearer YOUR_AUTH_TOKEN' \
-d '{"sku": {"name": "Standard", "capacity": 120}, "properties": {"model": {"format": "OpenAI", "name": "gpt-35-turbo", "version": "0613"}, "versionUpgradeOption": "OnceCurrentVersionExpired"} }'
```

① 참고

권한 부여 토큰을 생성하는 방법에는 여러 가지가 있습니다. 초기 테스트를 위한 가장 쉬운 방법은 [Azure Portal](#)에서 Cloud Shell을 시작하는 것입니다. 그런 다음 `az account get-access-token`를 실행합니다. 이 토큰을 API 테스트를 위한 임시 권한 부여 토큰으로 사용할 수 있습니다.

예제 응답

JSON

```
{
  "id": "/subscriptions/{subscription-id}/resourceGroups/resource-group-temp/providers/Microsoft.CognitiveServices/accounts/docs-openai-test-001/deployments/gpt-35-turbo",
  "type": "Microsoft.CognitiveServices/accounts/deployments",
```

```
"name": "gpt-35-turbo",
"sku": {
    "name": "Standard",
    "capacity": 120
},
"properties": {
    "model": {
        "format": "OpenAI",
        "name": "gpt-35-turbo",
        "version": "0613"
    },
    "versionUpgradeOption": "OnceCurrentVersionExpired",
    "capabilities": {
        "chatCompletion": "true"
    },
    "provisioningState": "Succeeded",
    "rateLimits": [
        {
            "key": "request",
            "renewalPeriod": 10,
            "count": 120
        },
        {
            "key": "token",
            "renewalPeriod": 60,
            "count": 120000
        }
    ]
},
"systemData": {
    "createdBy": "docs@contoso.com",
    "createdByType": "User",
    "createdAt": "2023-02-28T02:57:15.8951706Z",
    "lastModifiedBy": "docs@contoso.com",
    "lastModifiedByType": "User",
    "lastModifiedAt": "2023-10-31T15:35:53.082912Z"
},
"etag": "\"GUID\""
}
```

다음 단계

- Azure OpenAI 모델의 지역별 가용성에 대해 자세히 알아보기
- Azure OpenAI에 대해 자세히 알아보기

Azure AI 서비스 가상 네트워크 구성

아티클 • 2024. 04. 05.

Azure AI 서비스는 계층화된 보안 모델을 제공합니다. 이 모델을 사용하여 Azure AI 서비스 계정을 특정 네트워크 하위 집합으로 보호할 수 있습니다. 네트워크 규칙이 구성되면 지정된 네트워크 세트를 통해 데이터를 요청하는 애플리케이션만 계정에 액세스할 수 있습니다. 지정된 IP 주소, IP 범위 또는 [Azure Virtual Networks](#)의 서브넷 목록에서 시작하는 요청만을 허용하는 [요청 필터링](#)을 통해 리소스에 대한 액세스를 제한할 수 있습니다.

네트워크 규칙이 적용될 때 Azure AI 서비스에 액세스하는 애플리케이션에는 권한 부여가 필요합니다. 권한 부여는 [Microsoft Entra ID](#) 자격 증명 또는 유효한 API 키를 사용하여 지원됩니다.

ⓘ 중요

Azure AI 서비스 계정에 대한 방화벽 규칙을 설정하면 기본적으로 데이터에 대해 들어오는 요청이 차단됩니다. 요청을 허용하려면 다음 조건 중 하나를 충족해야 합니다.

- 요청은 대상 Azure AI 서비스 계정의 허용된 서브넷 목록에 있는 Azure VNet 내에서 작동하는 서비스로부터 시작됩니다. 가상 네트워크에서 시작된 요청의 엔드포인트는 Azure AI 서비스 계정의 [사용자 지정 하위 도메인](#)으로 설정되어야 합니다.
- 요청은 허용되는 IP 주소 목록에서 시작됩니다.

차단되는 요청에는 다른 Azure 서비스로부터의 요청, Azure Portal의 요청, 로깅 및 메트릭 서비스로부터의 요청이 포함됩니다.

ⓘ 참고

Azure Az PowerShell 모듈을 사용하여 Azure와 상호 작용하는 것이 좋습니다. 시작 하려면 [Azure PowerShell 설치](#)를 참조하세요. Az PowerShell 모듈로 마이그레이션 하는 방법에 대한 자세한 내용은 [Azure PowerShell을 AzureRM에서 Azure로 마이그레이션](#)을 참조하세요.

시나리오

Azure AI 서비스 리소스를 보호하려면 먼저 인터넷 트래픽을 비롯한 모든 네트워크의 트래픽에 대한 액세스를 거부하도록 규칙을 구성해야 합니다. 그런 다음, 특정 가상 네트워크에 대한 접근을 허용하는 규칙을 추가해야 합니다.

크의 트래픽에 대한 액세스를 허가하는 규칙을 구성합니다. 이 구성을 사용하면 애플리케이션에 대한 보안 네트워크 경계를 구축할 수 있습니다. 또한 특정 공용 인터넷 IP 주소 범위의 트래픽에 대한 액세스를 허가하도록 규칙을 구성하고 특정 인터넷 또는 온-프레미스 클라이언트의 연결을 사용하도록 설정할 수도 있습니다.

네트워크 규칙은 REST 및 WebSocket을 포함하여 Azure AI 서비스에 대한 모든 네트워크 프로토콜에 적용됩니다. Azure 테스트 콘솔 등의 도구를 사용하여 데이터에 액세스하려면 명시적 네트워크 규칙을 구성해야 합니다. 네트워크 규칙을 기준 Azure AI 서비스 리소스에 적용하거나 새 Azure AI 서비스 리소스를 만들 때 적용할 수 있습니다. 네트워크 규칙이 적용된 이후에는 모든 요청에 적용됩니다.

지원되는 지역 및 서비스 제공 사항

가상 네트워크는 [Azure AI 서비스를 사용할 수 있는 지역](#)에서 지원됩니다. Azure AI 서비스는 네트워크 규칙 구성에 대한 서비스 태그를 지원합니다. 여기에 나열된 서비스는 `CognitiveServicesManagement` 서비스 태그에 포함되어 있습니다.

- ✓ Anomaly Detector
- ✓ Azure OpenAI
- ✓ Content Moderator
- ✓ Custom Vision
- ✓ Face
- ✓ 언어 이해(LUIS)
- ✓ Personalizer
- ✓ Speech Service
- ✓ 언어
- ✓ QnA Maker
- ✓ Translator

① 참고

Azure OpenAI, LUIS, Speech Services 또는 언어 서비스를 사용하는 경우 `CognitiveServicesManagement` 태그를 통해서는 SDK 또는 REST API를 사용하는 서비스만 사용할 수 있습니다. 가상 네트워크에서 Azure OpenAI 스튜디오, LUIS 포털, Speech Studio 또는 Language Studio에 액세스하고 이를 사용하려면 다음 태그를 사용해야 합니다.

- `AzureActiveDirectory`
- `AzureFrontDoor.Frontend`
- `AzureResourceManager`
- `CognitiveServicesManagement`

- CognitiveServicesFrontEnd
- Storage (Speech Studio에만 해당)

Azure AI Studio 구성에 대한 자세한 내용은 [Azure AI Studio 설명서](#)를 참조하세요.

기본 네트워크 액세스 규칙 변경

기본적으로 Azure AI 서비스 리소스는 네트워크에 있는 클라이언트로부터의 연결을 허용합니다. 선택한 네트워크에 대한 액세스를 제한하려면 먼저 기본 동작을 변경해야 합니다.

⚠ 경고

네트워크 규칙을 변경하면 Azure AI 서비스에 연결하는 애플리케이션의 기능에 영향을 미칠 수 있습니다. 기본 네트워크 규칙을 거부로 설정하면 액세스를 허용하는 특정 네트워크 규칙이 적용되지 않는 한 데이터에 대한 모든 액세스가 차단됩니다.

액세스를 거부하도록 기본 규칙을 변경하기 전에 네트워크 규칙을 사용하여 허용된 모든 네트워크에 대한 액세스를 허가해야 합니다. 온-프레미스 네트워크에 대한 IP 주소를 나열하도록 허용하는 경우 온-프레미스 네트워크에서 사용 가능한 모든 나가는 공용 IP 주소를 추가해야 합니다.

기본 네트워크 액세스 규칙 관리

Azure Portal, PowerShell 또는 Azure CLI를 통해 Azure AI 서비스 리소스에 대한 기본 네트워크 액세스 규칙을 관리할 수 있습니다.

Azure Portal

1. 보안을 유지하려는 Azure AI 서비스 리소스로 이동합니다.
2. 리소스 관리를 선택하여 확장한 다음 **네트워킹**을 선택합니다.

The screenshot shows the Azure portal interface for managing networking settings. The left sidebar has a 'Networking' section highlighted with a red box. Under 'Allow access from', the 'Selected Networks and Private Endpoints' radio button is selected, also highlighted with a red box. A note at the top states: 'Access control settings allowing access to Azure AI services account will remain in effect for up to three minutes after saving updated settings restricting access.'

3. 기본으로 액세스를 거부하려면 방화벽 및 가상 네트워크에서 선택한 네트워크 및 프라이빗 엔드포인트를 선택합니다.

구성된 가상 네트워크 또는 주소 범위를 같이 사용하지 않고 이 설정만 단독으로 사용하면 모든 액세스가 사실상 거부됩니다. 모든 액세스가 거부되면 Azure AI 서비스 리소스를 사용하려는 요청이 허용되지 않습니다. 계속 Azure Portal, Azure PowerShell 또는 Azure CLI를 사용하여 Azure AI 서비스 리소스를 구성할 수 있습니다.

4. 모든 네트워크에서 트래픽을 허용하려면 모든 네트워크를 선택합니다.

The screenshot shows the Azure portal interface for managing networking settings. The 'All networks' radio button under 'Allow access from' is selected, highlighted with a red box. A note below states: 'All networks, including the internet, can access this resource.' The left sidebar has a 'Networking' section highlighted with a red box.

5. 저장을 선택하여 변경 내용을 적용합니다.

가상 네트워크의 액세스 허가

특정 서브넷에서만 액세스를 허용하도록 Azure AI 서비스 리소스를 구성할 수 있습니다. 허용된 서브넷은 동일하거나 다른 구독 내의 가상 네트워크에 속할 수 있습니다. 다른 구독은 다른 Microsoft Entra 테넌트에 속할 수 있습니다. 서브넷이 다른 구독에 속하는 경우 Microsoft.CognitiveServices 리소스 공급자도 해당 구독에 등록해야 합니다.

가상 네트워크 내에서 Azure AI 서비스에 서비스 엔드포인트를 사용하도록 설정합니다. 서비스 엔드포인트는 가상 네트워크의 트래픽을 Azure AI 서비스에 대한 최적의 경로를 통해 라우팅합니다. 자세한 내용은 [가상 네트워크 서비스 엔드포인트](#)를 참조하세요.

서브넷 및 가상 네트워크의 ID 또한 각 요청과 함께 전송됩니다. 그러면 관리자가 가상 네트워크의 특정 서브넷 요청을 허용하는 Azure AI 서비스 리소스에 대한 네트워크 규칙을 구성할 수 있습니다. 이러한 네트워크 규칙을 통해 액세스가 허가된 클라이언트는 데이터에 액세스하기 위해 Azure AI 서비스 리소스의 인증 요구 사항을 계속 충족해야 합니다.

각 Azure AI 서비스 리소스는 IP 네트워크 규칙과 결합될 수 있는 최대 100개의 가상 네트워크 규칙을 지원합니다. 자세한 내용은 이 문서의 뒷부분에 나오는 [인터넷 IP 범위에서 액세스 권한 부여](#)를 참조하세요.

필요한 권한 설정

가상 네트워크 규칙을 Azure AI 서비스 리소스에 적용하려면 추가할 서브넷에 대한 적절한 권한이 있어야 합니다. 필요한 권한은 기본 기여자 역할 또는 *Cognitive Services* 기여자 역할입니다. 필요한 사용 권한을 사용자 지정 역할 정의에 추가할 수도 있습니다.

액세스 권한이 허용된 Azure AI 서비스 리소스 및 가상 네트워크는 다른 Microsoft Entra 테넌트에 속하는 구독을 포함한 다른 구독에 있을 수 있습니다.

① 참고

다른 Microsoft Entra 테넌트의 일부인 가상 네트워크의 서브넷에 대한 액세스 권한을 부여하는 규칙 구성은 현재 PowerShell, Azure CLI 및 REST API를 통해서만 지원됩니다. Azure Portal에서 이러한 규칙을 볼 수 있지만 구성할 수는 없습니다.

가상 네트워크 규칙 구성

Azure Portal, PowerShell 또는 Azure CLI를 통해 Azure AI 서비스 리소스에 대한 가상 네트워크 규칙을 관리할 수 있습니다.

Azure Portal

기존 네트워크 규칙을 이용해 가상 네트워크에 액세스 권한을 부여하려면 다음을 수행합니다.

1. 보안을 유지하려는 Azure AI 서비스 리소스로 이동합니다.
2. 리소스 관리를 선택하여 확장한 다음 **네트워킹**을 선택합니다.
3. 선택한 **네트워크 및 프라이빗 엔드포인트**를 선택했는지 확인합니다.
4. 다음에서 **액세스 허용**에서 **기존 가상 네트워크 추가**를 선택합니다.

The screenshot shows the 'Networking' section of the Azure portal for the 'contoso-custom-vision' resource group. On the left, there's a sidebar with various service links like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource Management, Keys and Endpoint, Encryption, Pricing tier, and Networking (which is currently selected). The main area is titled 'Firewalls and virtual networks' and shows settings for allowing access from 'All networks', 'Selected Networks and Private Endpoints' (which is selected and highlighted with a red box), or 'Disabled'. It also includes sections for 'Virtual networks' (with '+ Add existing virtual network' and '+ Add new virtual network' buttons) and 'Firewall' (with an 'Address range' input field and a search icon).

5. **가상 네트워크 및 서브넷 옵션을 선택한 다음, 사용을 선택합니다.**

Add networks

X

Subscription *

Contoso Subscription

Virtual networks *

contoso-rg

Subnets *

default (Service endpoint required)

i The following networks don't have service endpoints enabled for 'Microsoft.CognitiveServices'. Enabling access will take up to 15 minutes to complete. After starting this operation, it is safe to leave and return later if you do not wish to wait.

Virtual network	Service endpoint status	
contoso-rg	Not enabled	...
default	Not enabled	...

Enable

ⓘ 참고

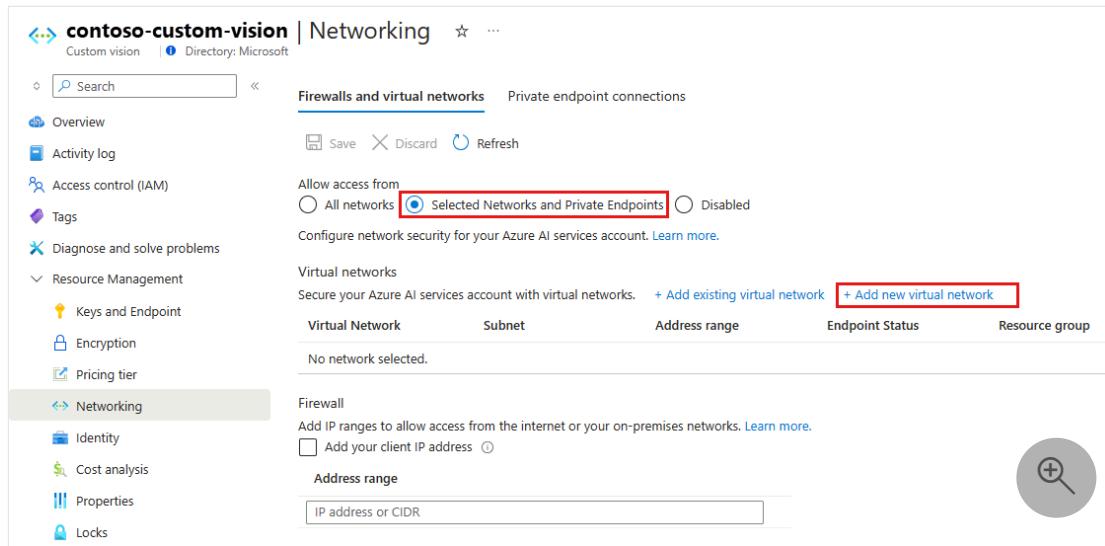
이전에 Azure AI 서비스에 대한 서비스 엔드포인트가 선택한 가상 네트워크 및 서브넷에 대해 구성되지 않은 경우 이 작업의 일환으로 구성할 수 있습니다.

현재는 동일한 Microsoft Entra 테넌트에 속한 가상 네트워크만 규칙을 만드는 동안 선택할 수 있습니다. 다른 테넌트에 속하는 가상 네트워크 내의 서브넷에 대한 액세스 권한을 부여하려면 PowerShell이나 Azure CLI 또는 REST API를 사용하세요.

6. 저장을 선택하여 변경 내용을 적용합니다.

새 가상 네트워크를 만들어 액세스 권한을 부여하려면 다음을 수행합니다.

1. 이전 절차와 동일한 페이지에서 새 가상 네트워크 추가를 선택합니다.



The screenshot shows the 'Networking' page for the 'contoso-custom-vision' resource in the Azure portal. The left sidebar has 'Networking' selected. The main area shows 'Firewalls and virtual networks' settings. Under 'Allow access from', the 'Selected Networks and Private Endpoints' radio button is selected. In the 'Virtual networks' section, there is a table with columns: Virtual Network, Subnet, Address range, Endpoint Status, and Resource group. Below the table, it says 'No network selected.' To the right, there is a 'Firewall' section with an 'Address range' input field containing 'IP address or CIDR' and a search icon. A red box highlights the '+ Add new virtual network' button.

2. 새 가상 네트워크를 만드는 데 필요한 정보를 입력하고 만들기를 선택합니다.

Create virtual network

* Name
widgets-vnet

* Address space ⓘ
10.1.0.0/16
10.1.0.0 - 10.1.255.255 (65536 addresses)

* Subscription
widgets-subscription

* Resource group
widgets-resource-group
[Create new](#)

* Location
(US) West US 2

Subnet

* Name
default

* Address range ⓘ
10.1.0.0/24
10.1.0.0 - 10.1.0.255 (256 addresses)

DDoS protection ⓘ
 Basic Standard

Service endpoint ⓘ
Microsoft.CognitiveServices

Firewall ⓘ
 Disabled Enabled

Create

3. 저장을 선택하여 변경 내용을 적용합니다.

가상 네트워크나 서브넷 규칙을 제거하려면 다음을 수행합니다.

1. 이전 절차와 동일한 페이지에서 ...(**추가 옵션**)을 선택하여 가상 네트워크와 서브넷용 바로 가기 메뉴를 열고 **제거**를 선택합니다.

Firewalls and virtual networks Private endpoint connections

Save Discard Refresh

Allow access from
All networks Selected Networks and Private Endpoints Disabled

Configure network security for your Azure AI services account. [Learn more.](#)

Virtual networks
Secure your Azure AI services account with virtual networks. + Add existing virtual network + Add new virtual network

Virtual Network	Subnet	Address range	Endpoint Status	Resource group	Subscription
contoso-01-vnet	1			contoso-rg	Remove ...

Firewall
Add IP ranges to allow access from the internet or your on-premises networks. [Learn more.](#)

Add your client IP address [?](#)

Address range
IP address or CIDR

2. 저장을 선택하여 변경 내용을 적용합니다.

① 중요

반드시 [기본 규칙](#)을 거부로 설정해야 합니다. 그렇지 않으면 네트워크 규칙이 적용되지 않습니다.

인터넷 IP 범위의 액세스 허가

특정 공용 인터넷 IP 주소 범위에서 액세스할 수 있도록 Azure AI 서비스 리소스를 구성할 수 있습니다. 이 구성은 특정 서비스와 온-프레미스 네트워크에 대한 액세스 권한을 부여하고 일반 인터넷 트래픽을 효과적으로 차단합니다.

192.168.0.0/16 양식의 [CIDR 형식\(RFC 4632\)](#)을 사용하거나 192.168.0.1 같은 개별 IP 주소로 허용된 인터넷 주소 범위를 지정할 수 있습니다.

💡 팁

접두사 크기가 /31 또는 /32인 작은 주소 범위는 지원하지 않습니다. 해당 범위는 개별 IP 주소 규칙을 사용하여 구성합니다.

IP 네트워크 규칙은 [공용 인터넷](#) IP 주소에 대해서만 허용됩니다. 프라이빗 네트워크에 예약된 IP 주소 범위는 IP 규칙에서 허용되지 않습니다. 사설망에는 10.*, 172.16.* - 172.31.* 및 192.168.*로 시작하는 주소가 포함됩니다. 자세한 내용은 [프라이빗 주소 공간\(RFC 1918\)](#)을 참조하세요.

현재는 IPv4 주소만 지원합니다. 각 Azure AI 서비스 리소스는 [가상 네트워크 규칙](#)과 결합될 수 있는 최대 100개의 IP 네트워크 규칙을 지원합니다.

온-프레미스 네트워크에서의 액세스 구성

IP 네트워크 규칙을 사용하여 온-프레미스 네트워크에서 Azure AI 서비스 리소스로의 액세스를 허가하려면 네트워크에서 사용되는 인터넷 연결 IP 주소를 식별합니다. 네트워크 관리자에게 도움을 요청합니다.

공용 피어링 또는 Microsoft 피어링을 위해 Azure ExpressRoute 온-프레미스를 사용하는 경우 NAT IP 주소를 식별해야 합니다. 자세한 내용은 [Azure ExpressRoute란?](#)을 참조하세요.

공용 피어링의 경우 기본적으로 각 ExpressRoute 회로에서 두 개의 NAT IP 주소를 사용합니다. 각각은 트래픽이 Microsoft Azure 네트워크 백본으로 들어갈 때 Azure 서비스 트래픽에 적용됩니다. Microsoft 피어링의 경우 사용되는 NAT IP 주소는 고객이 제공하거나 서비스 공급자가 제공합니다. 서비스 리소스에 대한 액세스를 허용하려면 리소스 IP 방화벽 설정에서 이러한 공용 IP 주소를 허용해야 합니다.

공용 피어링 ExpressRoute 회로 IP 주소를 찾으려면 Azure Portal을 통해 [ExpressRoute에서 지원 티켓을 엽니다](#). 자세한 내용은 [Azure 공용 피어링에 대한 NAT 요구 사항](#)을 참조하세요.

IP 네트워크 규칙 관리

Azure Portal, PowerShell 또는 Azure CLI를 통해 Azure AI 서비스 리소스에 대한 IP 네트워크 규칙을 관리할 수 있습니다.

Azure Portal

- 보안을 유지하려는 Azure AI 서비스 리소스로 이동합니다.
- 리소스 관리를 선택하여 확장한 다음 **네트워킹**을 선택합니다.
- 선택한 **네트워크 및 프라이빗 엔드포인트**를 선택했는지 확인합니다.
- 방화벽 및 가상 네트워크에서 주소 범위** 옵션을 찾습니다. 인터넷 IP 범위에 대한 액세스 권한을 부여하려면 [CIDR 형식](#)으로 된 IP 주소나 주소 범위를 입력합니다. 유효한 공용 IP(예약되지 않음) 주소만 허용됩니다.

Allow access from

All networks Selected Networks and Private Endpoints Disabled

Configure network security for your Azure AI services account. [Learn more](#).

Virtual networks

Secure your Azure AI services account with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

Virtual Network	Subnet	Address range	Endpoint Status	Resource group
No network selected.				

Firewall

Add IP ranges to allow access from the internet or your on-premises networks. [Learn more](#).

Add your client IP address ⓘ

Address range

IP address or CIDR

IP 네트워크 규칙을 제거하려면 주소 범위 옆에 있는 휴지통 ✖ 아이콘을 선택합니다.

5. 저장을 선택하여 변경 내용을 적용합니다.

ⓘ 중요

반드시 [기본 규칙을 거부로 설정해야 합니다](#). 그렇지 않으면 네트워크 규칙이 적용되지 않습니다.

프라이빗 엔드포인트 사용

Azure AI 서비스 리소스에서 [프라이빗 엔드포인트](#)를 사용하면 가상 네트워크의 클라이언트가 [Azure Private Link](#)를 통해 안전하게 데이터에 액세스하도록 할 수 있습니다. 프라이빗 엔드포인트는 Azure AI 서비스 리소스에 가상 네트워크 주소 공간의 IP 주소를 사용합니다. 가상 네트워크의 클라이언트와 리소스 간의 네트워크 트래픽이 Microsoft Azure 백본 네트워크에서 가상 네트워크와 프라이빗 링크를 통과하며 공용 인터넷에서의 노출을 제거합니다.

Azure AI 서비스 리소스의 프라이빗 엔드포인트를 사용하면 다음과 같은 작업을 할 수 있습니다.

- Azure AI 서비스에 대한 퍼블릭 엔드포인트의 모든 연결을 차단하도록 방화벽을 구성하여 Azure AI 서비스를 보호합니다.
- 가상 네트워크에서의 데이터 반출을 차단하여 가상 네트워크 보안을 강화합니다.

- Azure VPN 게이트웨이 또는 ExpressRoute의 개인 피어링을 사용하여 가상 네트워크에 연결하는 온-프레미스 네트워크에서 Azure AI 서비스 리소스로 안전하게 연결합니다.

프라이빗 엔드포인트 이해하기

프라이빗 엔드포인트는 [가상 네트워크](#) 내부의 Azure 리소스를 위한 특별한 네트워크 인터페이스입니다. Azure AI 서비스 리소스에 대한 프라이빗 엔드포인트를 만들면 가상 네트워크 내 클라이언트와 리소스 간에 보안 연결을 제공합니다. 프라이빗 엔드포인트에는 가상 네트워크의 IP 주소 범위에서 IP 주소가 할당됩니다. 프라이빗 엔드포인트와 Azure AI 서비스 간의 연결은 보안 프라이빗 링크를 사용합니다.

가상 네트워크 내의 애플리케이션은 프라이빗 엔드포인트를 통해 서비스에 원활하게 연결할 수 있습니다. 연결은 달리 사용할 수도 있는 동일한 연결 문자열 및 권한 부여 메커니즘을 사용합니다. 별도의 엔드포인트가 필요한 Speech Services는 예외입니다. 자세한 내용은 이 문서의 [Speech Services를 사용하는 프라이빗 엔드포인트](#)를 참조하세요. 프라이빗 엔드포인트는 REST를 포함하여 Azure AI 서비스 리소스에서 지원하는 모든 프로토콜과 함께 사용할 수 있습니다.

서비스 엔드포인트를 사용하는 서브넷에서 프라이빗 엔드포인트를 만들 수 있습니다. 서브넷의 클라이언트는 서비스 엔드포인트를 사용해 Azure AI 서비스 리소스에 액세스하는 동안 프라이빗 엔드포인트를 사용하여 다른 Azure AI 서비스 리소스에 연결할 수 있습니다. 자세한 내용은 [가상 네트워크 서비스 엔드포인트](#)를 참조하세요.

가상 네트워크에서 Azure AI 서비스 리소스에 대한 프라이빗 엔드포인트를 만들 때 Azure는 Azure AI 서비스 리소스 소유자에게 승인 동의 요청을 보냅니다. 프라이빗 엔드포인트 만들기를 요청한 사용자가 리소스의 소유자인 경우 이 동의 요청이 자동으로 승인됩니다.

Azure AI 서비스 리소스 소유자는 [Azure Portal](#)의 Azure AI 서비스 리소스 관련 [프라이빗 엔드포인트](#) 연결 탭을 통해 동의 요청 및 프라이빗 엔드포인트를 관리할 수 있습니다.

프라이빗 엔드포인트 지정

프라이빗 엔드포인트를 만들 때 연결할 Azure AI 서비스 리소스를 지정합니다. 프라이빗 엔드포인트를 만드는 방법에 대한 자세한 내용은 다음을 참조하세요.

- [Azure Portal](#)을 사용하여 프라이빗 엔드포인트 만들기
- [Azure PowerShell](#)을 사용하여 프라이빗 엔드포인트 만들기
- [Azure CLI](#)를 사용하여 프라이빗 엔드포인트 만들기.

프라이빗 엔드포인트에 연결

① 참고

Azure OpenAI Service는 다른 Azure AI 서비스와는 다른 프라이빗 DNS 영역 및 공용 DNS 영역 전달자를 사용합니다. 올바른 영역 및 전달자 이름은 [Azure 서비스 DNS 영역 구성](#)을 참조하세요.

프라이빗 엔드포인트를 사용하는 가상 네트워크의 클라이언트는 퍼블릭 엔드포인트에 연결하는 클라이언트와 동일한 Azure AI 서비스 리소스 연결 문자열을 사용합니다. 별도의 엔드포인트가 필요한 Speech Services는 예외입니다. 자세한 내용은 이 문서의 [Speech Services를 이용해 프라이빗 엔드포인트 사용하기](#)를 참조하세요. DNS 확인은 자동으로 프라이빗 링크를 통해 가상 네트워크에서 Azure AI 서비스 리소스로의 연결을 라우팅합니다.

기본적으로 Azure는 프라이빗 엔드포인트에 필요한 업데이트를 사용하여 가상 네트워크에 연결된 [프라이빗 DNS 영역](#)을 만듭니다. 자체 DNS 서버를 사용하는 경우 DNS 구성을 추가로 변경해야 할 수 있습니다. 프라이빗 엔드포인트에 필요할 수도 있는 업데이트는 이 문서의 [프라이빗 엔드포인트에 대한 DNS 변경 내용 적용](#)을 참조하세요.

Speech Service를 이용해 프라이빗 엔드포인트 사용하기

프라이빗 엔드포인트를 통해 [Speech service 사용하기](#)를 참조하세요.

프라이빗 엔드포인트에 대한 DNS 변경 내용 적용

프라이빗 엔드포인트를 만들 때 Azure AI 서비스 리소스에 대한 DNS `CNAME` 리소스 레코드는 `privatelink` 접두사가 있는 하위 도메인의 별칭으로 업데이트됩니다. 또한, 기본적으로 Azure는 프라이빗 엔드포인트에 대한 DNS A 리소스 레코드를 사용하여 `privatelink` 하위 도메인에 해당하는 프라이빗 DNS 영역을 만듭니다. 자세한 내용은 [Azure 프라이빗 DNS란?](#)을 참조하세요.

프라이빗 엔드포인트를 사용하여 가상 네트워크 외부에서 엔드포인트 URL을 확인하는 경우 Azure AI 서비스 리소스의 퍼블릭 엔드포인트로 확인됩니다. 프라이빗 엔드포인트를 호스트하는 가상 네트워크에서 확인하는 경우 엔드포인트 URL은 프라이빗 엔드포인트의 IP 주소로 확인됩니다.

이 접근 방식을 사용하면 프라이빗 엔드포인트를 호스트하는 가상 네트워크의 클라이언트와 가상 네트워크 외부의 클라이언트에 동일한 연결 문자열을 사용하여 Azure AI 서비스 리소스에 액세스할 수 있습니다.

네트워크에서 사용자 지정 DNS 서버를 사용하는 경우 클라이언트는 프라이빗 엔드포인트 IP 주소에 대한 Azure AI 서비스 리소스 엔드포인트의 FQDN(정규화된 도메인 이름)을

확인할 수 있어야 합니다. 프라이빗 링크 하위 도메인을 가상 네트워크의 개인 DNS 영역에 위임하도록 DNS 서버를 구성합니다.

💡 팁

사용자 지정 또는 온-프레미스 DNS 서버를 사용하는 경우 `privatelink` 하위 도메인의 Azure AI 서비스 리소스 이름을 프라이빗 엔드포인트 IP 주소로 확인하도록 DNS 서버를 구성해야 합니다. `privatelink` 하위 도메인을 가상 네트워크의 프라이빗 DNS 영역에 위임합니다. 또는 DNS 서버에서 DNS 영역을 구성하고 DNS A 레코드를 추가합니다.

프라이빗 엔드포인트를 지원하기 위해 자체 DNS 서버를 구성하는 방법에 대한 자세한 내용은 다음 리소스를 참조하세요.

- 자체 DNS 서버를 사용하는 이름 확인
- DNS 구성

신뢰할 수 있는 Azure 서비스에 Azure OpenAI에 대한 액세스 권한 부여

다른 앱의 네트워크 규칙을 유지하면서 신뢰할 수 있는 Azure 서비스의 하위 집합에 Azure OpenAI에 대한 액세스 권한을 부여할 수 있습니다. 그러면 이러한 신뢰할 수 있는 서비스는 관리 ID를 사용하여 Azure OpenAI 서비스를 인증합니다. 다음 표에는 해당 서비스의 관리 ID에 적절한 역할 할당이 있는 경우 Azure OpenAI에 액세스할 수 있는 서비스가 나와 있습니다.

[+] 테이블 확장

서비스	리소스 공급자 이름
Azure AI 서비스	<code>Microsoft.CognitiveServices</code>
Azure Machine Learning	<code>Microsoft.MachineLearningServices</code>
Azure AI 검색	<code>Microsoft.Search</code>

REST API를 사용하여 네트워크 규칙 예외를 만들면 신뢰할 수 있는 Azure 서비스에 네트워킹 액세스 권한을 부여할 수 있습니다.

Bash

```
accessToken=$(az account get-access-token --resource
```

```

https://management.azure.com --query "accessToken" --output tsv
$rid="/subscriptions/<your subscription id>/resourceGroups/<your resource
group>/providers/Microsoft.CognitiveServices/accounts/<your Azure AI
resource name>"

curl -i -X PATCH https://management.azure.com$rid?api-version=2023-10-01-
preview \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $accessToken" \
-d \
'
{
  "properties": {
    {
      "networkAcls": {
        "bypass": "AzureServices"
      }
    }
  }
'

```

① 참고

신뢰할 수 있는 서비스 기능은 위에서 설명한 명령줄에서만 사용할 수 있으며 Azure Portal에서는 수행할 수 없습니다.

예외를 철회하려면 `networkAcls.bypass` 를 `None` 으로 설정합니다.

Azure Portal에서 신뢰할 수 있는 서비스를 사용하도록 설정했는지 확인하려면

1. Azure OpenAI 리소스 개요 페이지에서 JSON 보기 사용

The screenshot shows the Azure OpenAI service overview page for a resource named 'my-openai-resource'. The left sidebar includes links for Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Resource Management (Keys and Endpoint, Model deployments, Encryption), Overview, and Essentials. The main content area displays resource details under 'Essentials': Resource group (move) : my-resource-group, Status : Active, Location : East US 2, Subscription (move) : My_subscription, Subscription ID : abcd1234, and Tags (edit) : Add tags. To the right of these details are API Kind : OpenAI, Pricing tier : Standard, Endpoints : Click here to view endpoints, and Manage keys : Click here to manage keys. At the bottom of the main content area, there is a call-to-action: 'Build your own secure copilot and generative AI applications with Azure OpenAI Service' and a note: 'Deploy an Azure OpenAI model and start making API calls. Connect your own data, call functions, and improve workflow with Azure OpenAI language, image and speech'. In the top right corner of the main content area, there is a 'View Cost' button and a 'JSON View' button, which is highlighted with a red box.

2. API 버전에서 최신 API 버전을 선택합니다. 최신 API 버전인 2023-10-01-preview 만 지원됩니다.

Resource JSON

X

Resource ID

/subscriptions/

/resourceGroups/

/providers/Microsoft/

API Versions

2023-10-01-preview



```
75     "networkAcls": {  
76       "bypass": "AzureServices",  
77       "defaultAction": "Deny",  
78       "virtualNetworkRules": [],  
79       "ipRules": []  
80     },
```

가격 책정

가격 책정에 대한 자세한 내용은 [Azure Private Link 가격 책정](#)을 참조하세요.

다음 단계

- 다양한 [Azure AI 서비스](#) 살펴보기
- [가상 네트워크 서비스 엔드포인트](#)에 대한 자세한 내용

미사용 데이터의 Azure OpenAI 서비스 암호화

아티클 • 2024. 02. 28.

Azure OpenAI는 클라우드에 유지되면 데이터를 자동으로 암호화합니다. 암호화는 데이터를 보호하고 조직의 보안 및 규정 준수 약정을 충족하는 데 도움이 됩니다. 이 문서에서는 Azure OpenAI가 미사용 데이터 암호화, 특히 학습 데이터 및 미세 조정된 모델을 처리하는 방법을 설명합니다. 서비스에 제공한 데이터가 처리, 사용 및 저장되는 방법에 대한 자세한 내용은 데이터, 개인 정보 및 보안 문서를 참조 [하세요](#).

Azure AI 서비스 암호화 정보

Azure OpenAI는 Azure AI 서비스의 일부입니다. Azure AI 서비스 데이터는 FIPS 140-2 호환 [256비트 AES](#) 암호화를 사용하여 암호화 및 복호화됩니다. 암호화 및 암호 해독은 투명하므로 암호화 및 액세스가 자동으로 관리됩니다. 데이터는 기본적으로 안전하며 암호화를 활용하기 위해 코드 또는 애플리케이션을 수정할 필요가 없습니다.

암호화 키 관리 정보

기본적으로 구독은 Microsoft에서 관리하는 암호화 키를 사용합니다. CMK(고객 관리형 키)라고 하는 사용자 고유의 키를 사용하여 구독을 관리하는 옵션도 있습니다. CMK는 액세스 제어를 만들고, 회전시키고, 사용하지 않도록 설정하고, 철회할 수 있는 훨씬 더 큰 유연성을 제공합니다. 데이터를 보호하는 데 사용되는 암호화 키를 감사할 수도 있습니다.

Azure Key Vault에서 고객 관리형 키 사용

BYOK(Bring Your Own Key)라고도 하는 CMK(고객 관리형 키) 사용하면 훨씬 더 유연하게 액세스 제어를 만들고, 회전하고, 사용하지 않도록 설정하고, 취소할 수 있습니다. 데이터를 보호하는 데 사용되는 암호화 키를 감사할 수도 있습니다.

고객 관리형 키를 저장하려면 Azure Key Vault를 사용해야 합니다. 사용자 고유의 키를 만들어 키 자격 증명 모음에 저장할 수도 있고, Azure Key Vault API를 사용하여 키를 생성할 수도 있습니다. Azure AI 서비스 리소스와 키 자격 증명 모음은 동일한 지역 및 동일한 Microsoft Entra 테넌트에 있어야 하지만 서로 다른 구독에 있을 수 있습니다. Azure Key Vault에 대한 자세한 내용은 [Azure Key Vault란?](#)을 참조하세요.

고객 관리형 키를 사용하도록 설정하려면 키가 포함된 키 자격 증명 모음이 다음 요구 사항을 충족해야 합니다.

- 키 자격 증명 모음에서 일시 삭제 및 제거 안 함 속성을 둘 다 사용하도록 설정해야 합니다.
- Key Vault 방화벽을 사용하는 경우 신뢰할 수 있는 Microsoft 서비스 키 자격 증명 모음에 액세스하도록 허용해야 합니다.
- 키 자격 증명 모음은 레거시 액세스 정책을 사용해야 합니다.
- 키 가져오기, 키 래핑, 키 래핑 해제와 같은 권한을 Azure OpenAI 리소스의 시스템 할당 관리 ID에 부여해야 합니다.

2048 크기의 RSA 및 RSA-HSM 키만 Azure AI 서비스 암호화에서 지원됩니다. 키에 대한 자세한 내용은 [Azure Key Vault 키, 비밀 및 인증서 정보](#)의 Key Vault 키를 참조하세요.

Azure OpenAI 리소스의 관리 ID 사용

1. Azure AI 서비스 리소스로 이동합니다.
2. 왼쪽의 리소스 관리에서 ID를 선택합니다.
3. 시스템 할당 관리 ID 상태 켜기로 전환합니다.
4. 변경 내용을 저장하고 시스템 할당 관리 ID를 사용하도록 설정할지 확인합니다.

키 자격 증명 모음의 액세스 권한 구성

1. Azure Portal에서 키 자격 증명 모음으로 이동합니다.
2. 왼쪽에서 액세스 정책을 선택합니다.
액세스 정책을 사용할 수 없다는 메시지가 표시되면 계속하기 전에 레거시 액세스 정책을 사용하도록 키 자격 증명 모음을 다시 구성합니다.
3. 만들기를 실행합니다.
4. 키 사용 권한에서 키 가져오기, 래핑 및 래핑 해제를 선택합니다. 다시 기본 검사box를 선택하지 않은 상태로 듭니다.

Configure from a template

Select a template

Key permissions	Secret permissions	Certificate permissions
Key Management Operations <input type="checkbox"/> Select all <input checked="" type="checkbox"/> Get <input type="checkbox"/> List <input type="checkbox"/> Update <input type="checkbox"/> Create <input type="checkbox"/> Import <input type="checkbox"/> Delete <input type="checkbox"/> Recover <input type="checkbox"/> Backup <input type="checkbox"/> Restore	Secret Management Operations <input type="checkbox"/> Select all <input type="checkbox"/> Get <input type="checkbox"/> List <input type="checkbox"/> Set <input type="checkbox"/> Delete <input type="checkbox"/> Recover <input type="checkbox"/> Backup <input type="checkbox"/> Restore	Certificate Management Operations <input type="checkbox"/> Select all <input type="checkbox"/> Get <input type="checkbox"/> List <input type="checkbox"/> Update <input type="checkbox"/> Create <input type="checkbox"/> Import <input type="checkbox"/> Delete <input type="checkbox"/> Recover <input type="checkbox"/> Backup <input type="checkbox"/> Restore
Cryptographic Operations <input type="checkbox"/> Select all <input type="checkbox"/> Decrypt <input type="checkbox"/> Encrypt <input checked="" type="checkbox"/> Unwrap Key <input checked="" type="checkbox"/> Wrap Key <input type="checkbox"/> Verify <input type="checkbox"/> Sign	Privileged Secret Operations <input type="checkbox"/> Select all <input type="checkbox"/> Purge	<input type="checkbox"/> Manage Contacts <input type="checkbox"/> Manage Certificate Authorities <input type="checkbox"/> Get Certificate Authorities <input type="checkbox"/> List Certificate Authorities <input type="checkbox"/> Set Certificate Authorities <input type="checkbox"/> Delete Certificate Authorities
Privileged Key Operations <input type="checkbox"/> Select all <input type="checkbox"/> Purge <input type="checkbox"/> Release		Privileged Certificate Operations <input type="checkbox"/> Select all <input type="checkbox"/> Purge
Rotation Policy Operations <input type="checkbox"/> Select all <input type="checkbox"/> Rotate <input type="checkbox"/> Get Rotation Policy <input type="checkbox"/> Set Rotation Policy		

5. 다음을 선택합니다.

6. Azure OpenAI 리소스의 이름을 검색하고 관리 ID를 선택합니다.

7. 다음을 선택합니다.

8. 다음을 선택하여 애플리케이션 설정 구성을 건너뜁니다.

9. 만들기를 실행합니다.

Azure OpenAI 리소스에서 고객 관리형 키 사용

Azure Portal에서 고객 관리형 키를 사용하도록 설정하려면 다음 단계를 수행합니다.

1. Azure AI 서비스 리소스로 이동합니다.
2. 왼쪽의 리소스 관리에서 암호화를 선택합니다.
3. 다음 스크린샷과 같이 암호화 형식에서 고객 관리형 키를 선택합니다.

Demo1234 | Encryption

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource Management

- Keys and Endpoint
- Model deployments
- Encryption**
- Pricing tier
- Networking
- Identity
- Cost analysis
- Properties
- Locks

Encryption

Save Discard

Cognitive services encryption protects your data at rest. Azure Cognitive services encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

By default, data in the Cognitive services account is encrypted using Microsoft Managed Keys. You may choose to bring your own key.

Learn More about Azure Cognitive services Encryption

Encryption type

Microsoft Managed Keys

Customer Managed Keys

The cognitive service account named 'Demo1234' will be granted access to the selected key vault. Both soft delete and purge protection will be enabled on the key vault and cannot be disabled. The selected key vault must be in same location with current resource. The selected key must be an RSA[Supported Json Web Key Types are ['RSA', 'RSA-HSM']] 2048 bit key. No other key-size/asymmetric key-type is supported.

Learn more about customer managed keys

Encryption key

Enter key URI

Select from Key Vault

Key URI

Subscription

OpenAI Enterprise Bug Bash

키 지정

고객 관리형 키를 사용하도록 설정한 후 Azure AI 서비스 리소스와 연결할 키를 지정할 수 있습니다.

키를 URI로 지정

키를 URI로 지정하려면 다음 단계를 수행합니다.

1. Azure Portal에서 키 자격 증명 모음으로 이동합니다.
2. 개체에서 키를 선택합니다.
3. 원하는 키를 선택한 다음 해당 키를 선택하여 버전을 확인합니다. 키 버전을 선택하여 해당 버전의 설정을 봅니다.
4. URI를 제공하는 키 식별자 값을 복사합니다.

Dashboard > Key vaults > storagesamplekvcli - Keys > customstoragekey > 17bf9182bb694f109b8dc6d1e9b69f29

17bf9182bb694f109b8dc6d1e9b69f29

Key Version

Save Discard

Properties

Key Type RSA

RSA Key Size 2048

Created 4/9/2019, 12:50:38 PM

Updated 4/9/2019, 12:50:38 PM

Key Identifier
<key-uri> 

Settings

Set activation date?

Set expiration date?

Enabled? Yes No

Tags
0 tags 

Permitted operations

Encrypt Sign Wrap Key

Decrypt Verify Unwrap Key

5. Azure AI 서비스 리소스로 돌아가서 암호화를 선택합니다.

6. 암호화 키에서 키 URI 입력을 선택합니다.

7. 복사한 URI를 키 URI 상자에 붙여넣습니다.

CMK-Test - Encryption
Cognitive Services

Search (Ctrl+ /)  Overview  Activity log  Access control (IAM)  Tags  Diagnose and solve problems  RESOURCE MANAGEMENT  Quick start  Keys and Endpoint  Encryption  Pricing tier  Virtual network  Identity  Billing By Subscription  Properties  Locks  Export template 

Encryption

Save Discard

Cognitive services encryption protects your data at rest. Azure Cognitive services encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

By default, data in the cognitive service account is encrypted using Microsoft Managed Keys. You may choose to bring your own key.

Please note that after enabling Cognitive Service Encryption, only new data will be encrypted, and any existing files in this cognitive service account will retroactively get encrypted by a background encryption process.

[Learn More about Azure Cognitive services Encryption](#)

Encryption type Microsoft Managed Keys Customer Managed Keys
The cognitive service account named 'CMK-Test' will be granted access to the selected key vault. Both soft delete and purge protection will be enabled on the key vault and cannot be disabled.

[Learn more about customer managed keys](#)

Encryption key Enter key URI Select from Key Vault

Key URI * 

Subscription  AICP-DEV 

8. 구독에서 키 자격 증명 모음이 포함된 구독을 선택합니다.

9. 변경 내용을 저장합니다.

키 자격 증명 모음에서 키 선택

키 자격 증명 모음에서 키를 선택하려면 먼저 키가 포함된 키 자격 증명 모음이 있는지 확인합니다. 그런 다음, 다음 단계를 수행합니다.

1. Azure AI 서비스 리소스로 이동한 다음 **암호화**를 선택합니다.
2. **암호화 키**에서 **Key Vault**에서 **선택**을 선택합니다.
3. 사용하려는 키가 포함된 키 자격 증명 모음을 선택합니다.
4. 사용하려는 키를 선택합니다.

The screenshot shows the Microsoft Azure portal interface. At the top, there's a blue header bar with the Microsoft Azure logo and a search bar that says "Search resources, services, and docs (G+/)". Below the header, the URL path is visible: Home > CMKTest01-SB - Encryption > Select key from Azure Key Vault. The main content area has a title "Select key from Azure Key Vault". It contains four form fields with dropdown menus:

- Subscription ***: A dropdown menu showing "AICP-DEV".
- Key vault ***: A dropdown menu showing "CMKTest-01SB" with a "Create new" link below it.
- Key ***: A dropdown menu showing "CMKTest-01SB" with a "Create new" link below it.
- Version ***: A dropdown menu showing "19fc5cfacbd34e47b373709c1e400902" with a "Create new" link below it.

5. 변경 내용을 저장합니다.

키 버전 업데이트

새 버전의 키를 만들 때 새 버전을 사용하도록 Azure AI 서비스 리소스를 업데이트합니다. 다음 단계를 수행합니다.

1. Azure AI 서비스 리소스로 이동한 다음 **암호화**를 선택합니다.
2. 새 키 버전의 URI를 입력합니다. 또는 키 자격 증명 모음을 선택한 다음 키를 다시 선택하여 버전을 업데이트할 수 있습니다.
3. 변경 내용을 저장합니다.

다른 키 사용

암호화에 사용하는 키를 변경하려면 다음 단계를 따릅니다.

1. Azure AI 서비스 리소스로 이동한 다음 **암호화**를 선택합니다.

- 새 키의 URI를 입력합니다. 또는 키 자격 증명 모음을 선택한 다음 새 키를 선택할 수 있습니다.
- 변경 내용을 저장합니다.

고객 관리형 키 순환

규정 준수 정책에 따라 Key Vault에서 고객 관리형 키를 회전할 수 있습니다. 키가 회전되면 새 키 URI를 사용하도록 Azure AI 서비스 리소스를 업데이트해야 합니다. Azure Portal에서 새 버전의 키를 사용하도록 리소스를 업데이트하는 방법을 알아보려면 [키 버전 업데이트](#)를 참조하세요.

키를 회전해도 리소스의 데이터 재암호화는 트리거되지 않습니다. 사용자는 추가적인 작업을 할 필요가 없습니다.

고객 관리형 키 철회

고객 관리형 암호화 키는 액세스 정책을 변경하거나 Key Vault에 대한 권한을 변경하거나 키를 삭제하여 해지할 수 있습니다.

레지스트리에서 사용하는 관리 ID의 액세스 정책을 변경하려면 [az-keyvault-delete-policy](#) 명령을 실행합니다.

Azure CLI

```
az keyvault delete-policy \
--resource-group <resource-group-name> \
--name <key-vault-name> \
--key_id <key-vault-key-id>
```

키의 개별 버전을 삭제하려면 [az-keyvault-key-delete](#) 명령을 실행합니다. 이 작업에는 '키/삭제' 권한이 필요합니다.

Azure CLI

```
az keyvault key delete \
--vault-name <key-vault-name> \
--id <key-ID>
```

① 중요

CMK를 계속 사용하는 동안 활성 고객 관리형 키에 대한 액세스를 취소하면 학습 데이터 및 결과 파일 다운로드, 새 모델 미세 조정 및 미세 조정된 모델 배포를 방지할

수 있습니다. 그러나 이전에 배포된 미세 조정된 모델은 해당 배포가 삭제될 때까지 계속 작동하고 트래픽을 처리합니다.

학습, 유효성 검사 및 학습 결과 데이터 삭제

Files API를 사용하면 고객이 모델을 미세 조정하기 위해 학습 데이터를 업로드할 수 있습니다. 이 데이터는 리소스와 동일한 지역 내의 Azure Storage에 저장되고 Azure 구독 및 API 자격 증명으로 논리적으로 격리됩니다. 업로드된 파일은 DELETE API 작업을 [통해 사용자가 삭제할](#) 수 있습니다.

미세 조정된 모델 및 배포 삭제

미세 조정 API를 사용하면 고객이 파일 API를 통해 서비스에 업로드한 학습 데이터를 기반으로 자체적으로 미세 조정된 버전의 OpenAI 모델을 만들 수 있습니다. 학습된 미세 조정 모델은 동일한 지역의 Azure Storage에 저장되고 유휴 상태에서 암호화되며(Microsoft 관리형 키 또는 고객 관리형 키) Azure 구독 및 API 자격 증명으로 논리적으로 격리됩니다. 사용자가 DELETE API 작업을 [호출하여 미세 조정된 모델 및 배포를 삭제할](#) 수 있습니다.

고객 관리형 키 사용 안 함

고객 관리형 키를 사용하지 않도록 설정하면 Azure AI 서비스 리소스가 Microsoft 관리형 키로 암호화됩니다. 고객 관리형 키를 사용하지 않도록 설정하려면 다음 단계를 수행합니다.

1. Azure AI 서비스 리소스로 이동한 다음 **암호화**를 선택합니다.
2. Microsoft 관리형 키 저장을>선택합니다.

이전에 고객 관리 키를 사용하도록 설정했을 때 Microsoft Entra ID의 기능인 시스템 할당 관리 ID도 사용하도록 설정되었습니다. 시스템 할당 관리 ID를 사용하도록 설정하면 이 리소스가 Microsoft Entra ID에 등록됩니다. 등록 후 관리 ID에는 고객 관리형 키를 설정하는 동안 선택된 키 자격 증명 모음에 대한 액세스 권한이 부여됩니다. 여기에서 [관리 ID](#)에 대해 자세히 알아볼 수 있습니다.

ⓘ 중요

시스템 할당 관리 ID를 사용하지 않도록 설정하면 키 자격 증명 모음에 대한 액세스 권한이 제거되고 고객 키로 암호화된 데이터에 더 이상 액세스할 수 없게 됩니다. 이 데이터에 의존하는 기능은 작동하지 않습니다.

① 중요

관리 ID는 현재 교차 디렉터리 시나리오를 지원하지 않습니다. Azure Portal에 고객 관리형 키를 구성하는 경우 관리 ID가 내부적으로 자동 할당됩니다. 이후에 구독, 리소스 그룹 또는 리소스를 Microsoft Entra 디렉터리 간에 이동하는 경우, 리소스와 연결된 관리 ID가 새로운 테넌트로 전송되지 않으므로 고객 관리형 키가 더 이상 작동하지 않을 수 있습니다. 자세한 내용은 FAQ에서 [디렉터리 간 구독 전송 및 Azure 리소스에 대한 관리 ID의 알려진 문제](#)를 참조하세요.

다음 단계

- [Azure Key Vault에 대해 자세히 알아보기](#)

관리 ID를 사용하여 Azure OpenAI Service를 구성하는 방법

아티클 • 2024. 04. 11.

더 복잡한 보안 시나리오에는 Azure RBAC(Azure 역할 기반 액세스 제어)가 필요합니다. 이 문서에서는 Microsoft Entra ID를 사용하여 OpenAI 리소스에 인증하는 방법을 설명합니다.

다음 섹션에서는 Azure CLI를 사용하여 로그인하고 전달자 토큰을 가져와 OpenAI 리소스를 호출합니다. 작업하면서 어려움에 처할 경우를 위해 각 세션에서는 Azure Cloud Shell/Azure CLI의 각 명령에 대해 사용 가능한 모든 옵션이 있는 링크가 제공됩니다.

필수 조건

- Azure 구독 - [체험 구독 만들기](#)
- 원하는 Azure 구독의 Azure OpenAI Service에 부여된 액세스 권한
- 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. [Azure OpenAI Service에 대한 액세스 요청 양식](#)을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 엽니다.
- 인증을 위해 Microsoft Entra ID와 같은 기능을 사용하려면 사용자 지정 하위 도메인 이름이 필요합니다.
- Azure CLI - [설치 가이드](#)
- 다음 Python 라이브러리: os, 요청, json, openai, azure-identity

Cognitive Services 사용자 역할에 자신을 할당합니다.

[Cognitive Services OpenAI 사용자](#) 또는 [Cognitive Services OpenAI 기여자](#) 역할을 할당하여 키 기반 인증을 사용하지 않고도 계정을 사용하여 Azure OpenAI 유추 API 호출을 수행할 수 있도록 합니다. 이 변경을 수행한 후 변경 내용이 적용되기까지 최대 5분이 걸릴 수 있습니다.

Azure CLI에 로그인

Azure CLI에 로그인하려면 다음 명령을 실행하고 로그인을 완료합니다. 세션이 너무 오랫동안 유회 상태인 경우 다시 수행해야 할 수 있습니다.

```
Azure CLI
```

```
az login
```

채팅 완료

```
Python
```

```
from azure.identity import DefaultAzureCredential, get_bearer_token_provider
from openai import AzureOpenAI

token_provider = get_bearer_token_provider(
    DefaultAzureCredential(), "https://cognitiveservices.azure.com/.default"
)

client = AzureOpenAI(
    api_version="2024-02-15-preview",
    azure_endpoint="https://{{your-custom-endpoint}}.openai.azure.com/",
    azure_ad_token_provider=token_provider
)

response = client.chat.completions.create(
    model="gpt-35-turbo-0125", # model = "deployment_name".
    messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Does Azure OpenAI support customer managed keys?"},
        {"role": "assistant", "content": "Yes, customer managed keys are supported by Azure OpenAI."},
        {"role": "user", "content": "Do other Azure AI services support this too?"}
    ]
)

print(response.choices[0].message.content)
```

컨트롤 플레인 API를 사용하여 Azure OpenAI 쿼리

```
Python
```

```
import requests
import json
from azure.identity import DefaultAzureCredential
```

```
region = "eastus"
token_credential = DefaultAzureCredential()
subscriptionId = "{YOUR-SUBSCRIPTION-ID}"

token = token_credential.get_token('https://management.azure.com/.default')
headers = {'Authorization': 'Bearer ' + token.token}

url =
f"https://management.azure.com/subscriptions/{subscriptionId}/providers/Microsoft.CognitiveServices/locations/{region}/models?api-version=2023-05-01"

response = requests.get(url, headers=headers)

data = json.loads(response.text)

print(json.dumps(data, indent=4))
```

관리 ID에 대한 액세스 권한 부여

OpenAI는 [Azure 리소스에 대한 관리 ID](#)를 사용하는 Microsoft Entra 인증을 지원합니다. Azure 리소스에 대한 관리 ID는 Azure VM(가상 머신), 함수 앱, 가상 머신 확장 집합 및 기타 서비스에서 실행되는 애플리케이션의 Microsoft Entra 자격 증명을 사용하여 Azure AI 서비스 리소스에 대한 액세스 권한을 부여할 수 있습니다. Microsoft Entra 인증과 함께 Azure 리소스에 대한 관리 ID를 사용하면 클라우드에서 실행되는 응용 프로그램에 자격 증명을 저장할 필요가 없습니다.

VM에서 관리 ID 사용

Azure 리소스에 대한 관리 ID를 사용하여 VM에서 Azure AI 서비스 리소스에 대한 액세스 권한을 부여하려면 먼저 VM에서 Azure 리소스에 대한 관리 ID를 사용하도록 설정해야 합니다. Azure 리소스의 관리 ID를 사용하도록 설정하는 방법을 알아보려면 다음을 참조하세요.

- [Azure Portal](#)
- [Azure PowerShell](#)
- [Azure CLI](#)
- [Azure Resource Manager 템플릿](#)
- [Azure Resource Manager 클라이언트 라이브러리](#)

관리 ID에 대한 자세한 내용은 [Azure 리소스의 관리 ID](#)를 참조하세요.

데이터에 대한 Azure OpenAI를 안전하게 사용

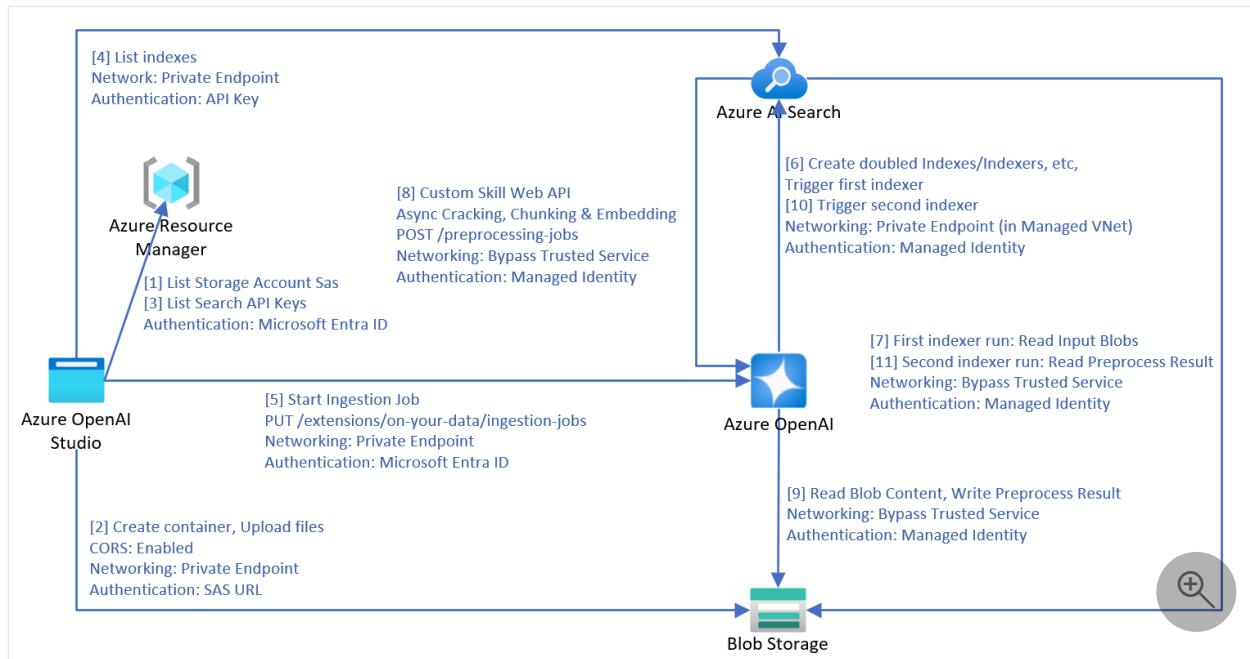
아티클 • 2024. 04. 05.

이 문서를 통해 Microsoft Entra ID 역할 기반 액세스 제어, 가상 네트워크 및 프라이빗 엔드포인트로 데이터와 리소스를 보호하여 Azure OpenAI On Your Data를 안전하게 사용하는 방법을 알아봅니다.

이 문서는 [텍스트가 포함된 Azure OpenAI On Your Data](#)를 사용하는 경우에만 적용됩니다. [이미지가 포함된 Azure OpenAI On Your Data](#)에는 적용되지 않습니다.

데이터 수집 아키텍처

Azure OpenAI On Your Data를 사용하여 Azure Blob Storage, 로컬 파일 또는 URL의 데이터를 Azure AI 검색으로 수집하는 경우 다음 프로세스를 사용하여 데이터를 처리합니다.

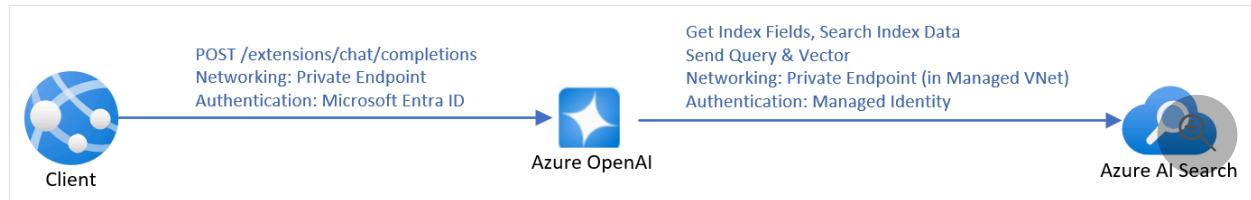


- 1단계와 2단계는 파일 업로드에만 사용됩니다.
- Blob Storage에 대한 URL 다운로드는 이 디아이어그램에 나와 있지 않습니다. 웹 페이지를 인터넷에서 다운로드하고 Blob Storage에 업로드한 후 3단계 이후는 동일합니다.
- 두 개의 인덱서, 두 개의 인덱스, 두 개의 데이터 원본 및 하나의 [사용자 지정 기술](#)이 Azure AI 검색 리소스에 만들어집니다.
- 청크 컨테이너는 Blob Storage에 만들어집니다.
- [예약된 새로 고침](#)에 의해 수집이 트리거되면 수집 프로세스가 7단계부터 시작됩니다.

- Azure OpenAI의 `preprocessing-jobs` API는 Azure AI 검색 고객 기술 웹 API 프로토콜을 구현하고 큐의 문서를 처리합니다.
- Azure OpenAI:
 1. 내부적으로는 이전에 만들어진 첫 번째 인덱서를 사용하여 문서를 크래ップ니다.
 2. 휴리스틱 기반 알고리즘을 사용하여 청크를 수행하고 청크 경계의 테이블 레이아웃과 기타 서식 지정 요소를 준수하여 최고의 청크 품질을 보장합니다.
 3. 벡터 검색을 사용하도록 설정하도록 선택한 경우 Azure OpenAI는 선택한 포함 배포를 사용하여 청크를 내부적으로 벡터화합니다.
- 서비스가 모니터링하는 모든 데이터가 처리되면 Azure OpenAI는 두 번째 인덱서를 트리거합니다.
- 인덱서는 처리된 데이터를 Azure AI 검색 서비스에 저장합니다.

서비스 호출에 사용되는 관리 ID의 경우 시스템 할당 관리 ID만 지원됩니다. 사용자 할당 관리 ID는 지원되지 않습니다.

유추 아키텍처



데이터에 대한 Azure OpenAI 모델과 채팅하기 위해 API 호출을 보내는 경우 서비스는 필드 매핑이 요청에 명시적으로 설정되지 않은 경우 자동으로 필드 매핑을 수행하기 위해 유추 중에 인덱스 필드를 검색해야 합니다. 따라서 유추 중에도 검색 서비스에 대한 `Search Service Contributor` 역할을 가지려면 서비스에서 Azure OpenAI ID가 필요합니다.

유추 요청에 포함 배포가 제공되는 경우 다시 작성된 쿼리는 Azure OpenAI에 의해 벡터화되고 쿼리와 벡터 모두 벡터 쿼리를 위해 Azure AI 검색으로 전송됩니다.

문서 수준 액세스 제어

① 참고

문서 수준 액세스 제어는 Azure AI 검색에 대해서만 지원됩니다.

Azure OpenAI On Your Data를 사용하면 Azure AI Search [보안 필터](#)를 사용하여 다른 사용자에 대한 응답에 사용할 수 있는 문서를 제한할 수 있습니다. 문서 수준 액세스를 사용하도록 설정하면 Azure AI Search에서 반환되고 응답을 생성하는 데 사용되는 검색 결과가 사용자 Microsoft Entra 그룹 멤버 자격에 따라 잘립니다. 기존 Azure AI Search 인덱스에서만 문서 수준 액세스를 사용하도록 설정할 수 있습니다. 문서 수준 액세스를 사용하도록 설정하려면 다음을 수행합니다.

1. [Azure AI Search 설명서](#)의 단계에 따라 애플리케이션을 등록하고 사용자 및 그룹을 만듭니다.
2. [허용된 그룹으로 문서를 인덱싱합니다.](#) 새 [보안 필드](#)에 아래의 스키마가 있는지 확인합니다.

JSON

```
{"name": "group_ids", "type": "Collection(Edm.String)", "filterable": true }
```

`group_ids`는 기본 필드 이름입니다. `my_group_ids` 등의 다른 필드 이름을 사용하는 경우 [인덱스 필드 매핑](#)에서 필드를 매핑할 수 있습니다.

3. 인덱스의 각 중요한 문서에 대해 이 보안 필드에 올바르게 설정된 값이 있는지 확인하여 문서의 허용된 그룹을 나타냅니다.
4. [Azure OpenAI Studio](#)에서 데이터 원본을 추가합니다. [인덱스 필드 매핑](#) 섹션에서 스키마가 호환되는 한, [허용된 그룹](#) 필드에 0개 또는 1개의 값을 매핑할 수 있습니다. [허용된 그룹](#) 필드가 매핑되지 않으면 문서 수준 액세스가 사용하도록 설정되지 않습니다.

Azure OpenAI Studio

Azure AI Search 인덱스가 연결되면 스튜디오의 응답은 로그인한 사용자의 Microsoft Entra 권한에 따라 문서 액세스 권한을 갖습니다.

웹 앱

게시된 [웹 앱](#)을 사용하는 경우 최신 버전으로 업그레이드하려면 다시 배포해야 합니다. 최신 버전의 웹 앱에는 로그인한 사용자의 Microsoft Entra 계정 그룹을 검색하고, 캐시하고, 각 API 요청에 그룹 ID를 포함하는 기능이 포함되어 있습니다.

API

API를 사용하는 경우 각 API 요청에 `filter` 매개 변수를 전달합니다. 예시:

JSON

```
{
  "messages": [
    {
      "role": "user",
      "content": "who is my manager?"
    }
  ],
  "dataSources": [
    {
      "type": "AzureCognitiveSearch",
      "parameters": {
        "endpoint": "'$AZURE_AI_SEARCH_ENDPOINT'",
        "key": "'$AZURE_AI_SEARCH_API_KEY'",
        "indexName": "'$AZURE_AI_SEARCH_INDEX'",
        "filter": "my_group_ids/any(g:search.in(g, 'group_id1,
group_id2'))"
      }
    }
  ]
}
```

- `my_group_ids`는 [필드 매핑](#) 중에 [허용된 그룹](#)에 대해 선택한 필드 이름입니다.
- `group_id1`, `group_id2`는 로그인한 사용자에서 기인한 그룹입니다. 클라이언트 애플리케이션은 사용자 그룹을 검색하고 캐시할 수 있습니다.

리소스 구성

다음 섹션을 사용하여 최적의 보안 사용을 위해 리소스를 구성합니다. 리소스의 일부만 보호하려는 경우에도 아래 단계를 모두 수행해야 합니다.

이 문서에서는 Azure OpenAI 리소스, Azure AI 검색 리소스 및 스토리지 계정에 대한 공용 네트워크를 사용하지 않도록 설정하는 데 관련된 네트워크 설정을 설명합니다. 서비스의 IP 주소가 동적이므로 선택한 네트워크를 IP 규칙과 함께 사용하는 것은 지원되지 않습니다.

💡 팁

[GitHub](#)에서 사용할 수 있는 bash 스크립트를 사용하여 설치 유효성을 검사하고, 여기에 나열된 모든 요구 사항이 충족되고 있는지 확인할 수 있습니다.

리소스 그룹 만들기

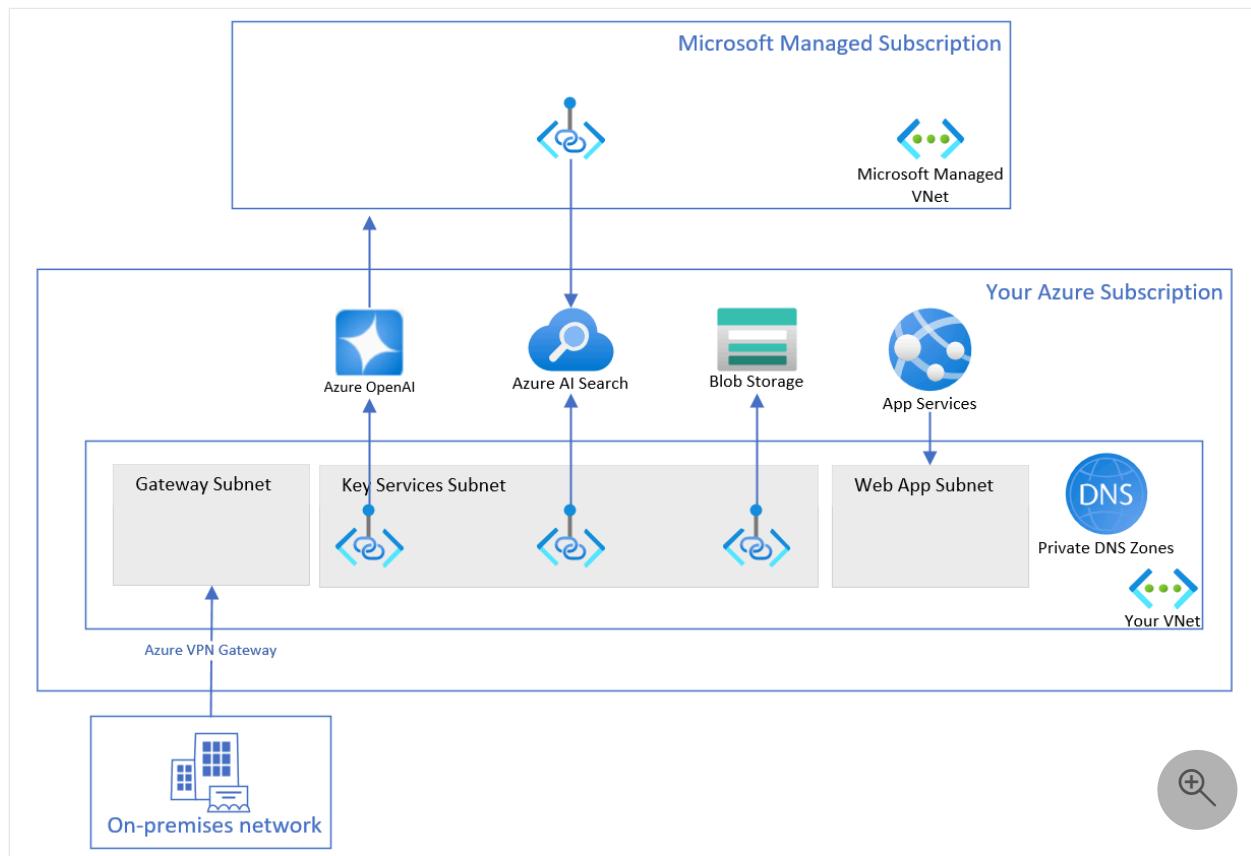
모든 관련 리소스를 구성할 수 있도록 리소스 그룹을 만듭니다. 리소스 그룹의 리소스에는 다음이 포함되지만 이에 국한되지는 않습니다.

- 가상 네트워크 1개
- 세 가지 주요 서비스: Azure OpenAI 1개, Azure AI 검색 1개, 스토리지 계정 1개
- 3개의 프라이빗 엔드포인트, 각각 하나의 키 서비스에 연결됨
- 3개의 네트워크 인터페이스(각각은 하나의 프라이빗 엔드포인트와 연결됨)
- 온-프레미스 클라이언트 컴퓨터에서 액세스하기 위한 가상 네트워크 게이트웨이 1개
- 가상 네트워크가 통합된 하나의 웹앱
- 하나의 프라이빗 DNS 영역(웹앱이 Azure OpenAI의 IP를 찾을 수 있도록)

가상 네트워크 만들기

가상 네트워크에는 3개의 서브넷이 있습니다.

1. 첫 번째 서브넷은 세 프라이빗 엔드포인트의 개인 IP에 사용됩니다.
2. 두 번째 서브넷은 가상 네트워크 게이트웨이를 만들 때 자동으로 만들어집니다.
3. 세 번째 서브넷은 비어 있으며 웹앱 아웃바운드 가상 네트워크 통합에 사용됩니다.



Microsoft 관리되는 가상 네트워크는 Microsoft에서 만들어졌으므로 볼 수 없습니다.
Microsoft 관리되는 가상 네트워크는 Azure OpenAI에서 Azure AI 검색에 안전하게 액세스하는 데 사용됩니다.

Azure OpenAI 구성

사용자 지정 하위 도메인을 사용하도록 설정했습니다.

Azure Portal을 통해 Azure OpenAI를 만든 경우 [사용자 지정 하위 도메인](#)이 이미 만들어져 있어야 합니다. Microsoft Entra ID 기반 인증 및 프라이빗 DNS 영역에는 사용자 지정 하위 도메인이 필요합니다.

관리 ID 사용

Azure AI 검색 및 스토리지 계정이 Microsoft Entra ID 인증을 통해 Azure OpenAI 서비스를 인식할 수 있도록 하려면 Azure OpenAI 서비스에 대한 관리 ID를 할당해야 합니다. 가장 쉬운 방법은 Azure Portal에서 시스템이 할당한 관리 ID를 켜는 것입니다.

The screenshot shows the Azure OpenAI service configuration page under the 'Identity' section. The 'System assigned' tab is selected. The status is set to 'On'. There is a placeholder for the object (principal) ID and a 'Permissions' section with a 'Azure role assignments' button.

관리 API를 통해 관리 ID를 설정하려면 [관리 API 참조 설명서](#)를 확인합니다.

```
JSON
{
  "identity": {
    "principalId": "12345678-abcd-1234-5678-abc123def",
    "tenantId": "1234567-abcd-1234-1234-abcd1234",
    "type": "SystemAssigned, UserAssigned",
    "userAssignedIdentities": {
      "/subscriptions/1234-5678-abcd-1234-1234abcd/resourceGroups/my-resource-group",
      "principalId": "12345678-abcd-1234-5678-abcdefg1234",
      "clientId": "12345678-abcd-efgh-1234-12345678"
    }
  }
}
```

신뢰할 수 있는 서비스 사용

Azure AI 검색이 Azure OpenAI `preprocessing-jobs`를 사용자 지정 기술 웹 API로 호출하도록 허용하려면 Azure OpenAI에는 공용 네트워크 액세스가 없지만 Azure AI 검색을 관리 ID 기반의 신뢰할 수 있는 서비스로 무시하도록 Azure OpenAI를 설정해야 합니다.

Azure OpenAI는 JWT(JSON Web Token)의 클레임을 확인하여 Azure AI 검색의 트래픽을 식별합니다. Azure AI 검색은 사용자 지정 기술 웹 API를 호출하려면 시스템이 할당한 관리 ID 인증을 사용해야 합니다.

관리 API에서 `networkAccls.bypass`를 `AzureServices`로 설정합니다. 자세한 내용은 [가상 네트워크 문서](#)를 참조하세요.

Azure AI 검색 리소스에 대한 [공유 프라이빗 링크](#)가 있는 경우에만 이 단계를 건너뛸 수 있습니다.

공용 네트워크 액세스 사용 안 함

Azure Portal에서 Azure OpenAI 리소스의 공용 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다.

Azure OpenAI Studio를 사용하는 것과 같이 클라이언트 컴퓨터에서 Azure OpenAI 서비스에 대한 액세스를 허용하려면 Azure OpenAI 리소스에 연결하는 [프라이빗 엔드포인트 연결](#)을 만들어야 합니다.

Azure AI 검색 구성

아래 구성에는 기본 가격 책정 계층 이상을 사용할 수 있습니다. 필수는 아니지만 S2 가격 책정 계층을 사용하는 경우 선택 가능한 [추가 옵션](#)이 표시됩니다.

관리 ID 사용

다른 리소스가 Microsoft Entra ID 인증을 사용하여 Azure AI 검색을 인식할 수 있도록 하려면 Azure AI 검색에 대한 관리 ID를 할당해야 합니다. 가장 쉬운 방법은 Azure Portal에서 시스템이 할당한 관리 ID를 켜는 것입니다.

Identity

System assigned

A system assigned managed identity is restricted to one per resource and is authenticated with Microsoft Entra ID, so you don't have to store secrets.

Status: On

Object (principal) ID:

Azure role assignments

역할 기반 액세스 제어 사용

Azure OpenAI는 관리 ID를 사용하여 Azure AI 검색에 액세스하므로 Azure AI 검색에서 역할 기반 액세스 제어를 사용하도록 설정해야 합니다. Azure Portal에서 이 작업을 수행 하려면 Azure Portal의 키 탭에서 **모두**를 선택합니다.

Keys

API access control

API keys

Role-based access control

Both

Manage admin keys

REST API를 통해 역할 기반 액세스 제어를 사용하도록 설정하려면 `authOptions`를 `aadOrApiKey`로 설정합니다. 자세한 내용은 [Azure AI 검색 RBAC 문서](#)를 참조하세요.

JSON

```
"disableLocalAuth": false,  
"authOptions": {  
    "aadOrApiKey": {  
        "aadAuthFailureMode": "http401WithBearerChallenge"  
    }  
}
```

Azure OpenAI Studio를 사용하려면 Azure AI 검색에 대한 API 키 기반 인증을 사용하지 않도록 설정할 수 없습니다. Azure OpenAI Studio는 API 키를 사용하여 브라우저에서 Azure AI 검색 API를 호출하기 때문입니다.

💡 팁

최상의 보안을 위해 프로덕션 준비가 되었고 더 이상 테스트를 위해 Azure OpenAI Studio를 사용할 필요가 없으면 API 키를 사용하지 않도록 설정하는 것이 좋습니다. 자세한 내용은 [Azure AI 검색 RBAC 문서](#)를 참조하세요.

공용 네트워크 액세스 사용 안 함

Azure Portal에서 Azure AI 검색 리소스의 공용 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다.

Azure OpenAI Studio를 사용하는 것과 같이 클라이언트 컴퓨터에서 Azure AI 검색 리소스에 대한 액세스를 허용하려면 Azure AI 검색 리소스에 연결하는 [프라이빗 엔드포인트 연결](#)을 만들어야 합니다.

ⓘ 참고

Azure OpenAI 리소스에서 Azure AI 검색 리소스에 대한 액세스를 허용하려면 [애플리케이션 양식](#)을 제출해야 합니다. 신청서는 영업일 기준 10일 이내에 검토되며 결과는 이메일을 통해 연락드립니다. 자격이 있는 경우 Microsoft 관리되는 가상 네트워크에서 프라이빗 엔드포인트를 프로비전하고 검색 서비스에 프라이빗 엔드포인트 연결 요청을 보내며, 사용자는 요청을 승인해야 합니다.

The screenshot shows the Azure AI services | Cognitive search test Networking page. On the left, there's a sidebar with options like Search, Search management, Settings, Semantic search (Preview), Knowledge Center, Keys, Scale, Search traffic analytics, Identity, Networking (which is selected and highlighted in grey), and Properties. The main area has tabs for Public access, Private access (which is selected), and Shared private access. A note says: "Private endpoints allow access to this resource using a private IP address from a virtual network, effectively bringing the service into your network. Learn more." Below this is a section titled "Private endpoint connections" with a note: "Allow selected virtual networks to connect to your resource using private endpoints." It includes buttons for "+ Create a private endpoint", "Refresh", "Approve" (with a checkmark), "Reject", and "Remove". There's also a "Filter by name..." input field and a search icon. A table lists one connection: "test" (Private endpoint), "test" (Connection name), "searchService" (Sub-resource), "Pending" (Connection state), and "Azure OpenAI on your data" (Description). A magnifying glass icon is at the bottom right of the table.

프라이빗 엔드포인트 리소스는 연결된 리소스가 테넌트에 있는 동안 Microsoft 관리 테넌트에 프로비전됩니다. 네트워킹 페이지의 **프라이빗 액세스** 탭에서 **프라이빗 엔드포인트** 링크(파란색 글꼴)를 그냥 클릭하는 것으로는 프라이빗 엔드포인트 리소스에 액세스할 수 없습니다. 대신 행의 다른 곳을 클릭하면 위의 **승인** 단추를 클릭할 수 있습니다.

[수동 승인 작업 흐름](#)에 대해 자세히 알아봅니다.

공유 프라이빗 링크 만들기

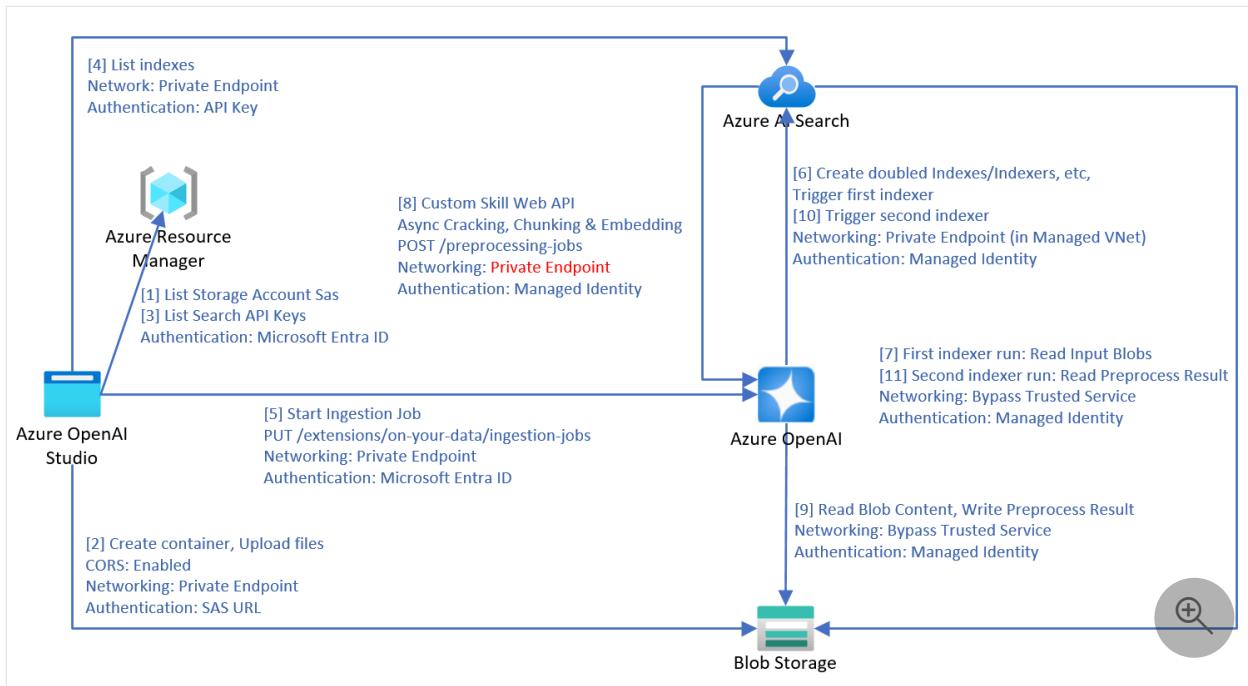
💡 팁

기본 또는 표준 가격 책정 계층을 사용하거나 모든 리소스를 안전하게 설정하는 것 이 처음인 경우 이 고급 항목을 건너뛰어야 합니다.

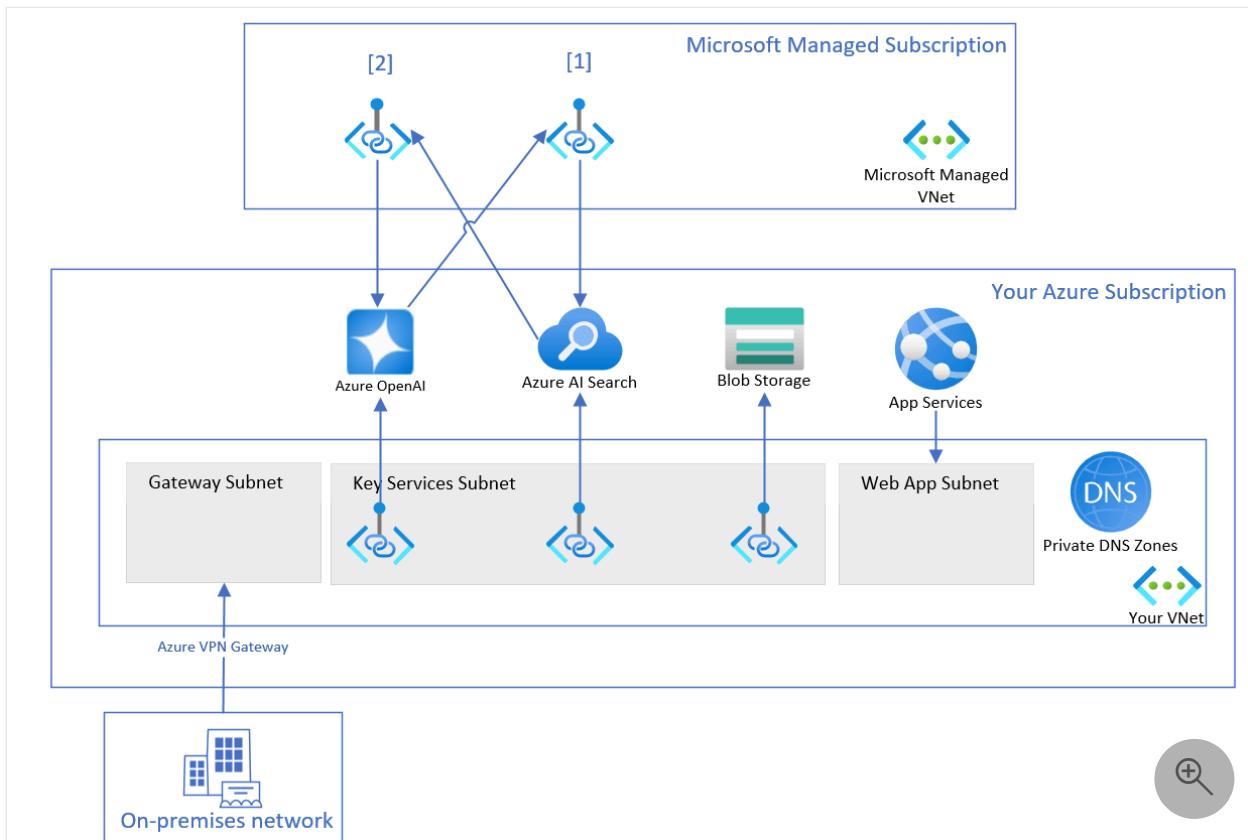
이 섹션은 [기술 집합이 있는 인덱서에 대한 프라이빗 엔드포인트 지원](#)이 필요하기 때문에 S2 가격 책정 계층 검색 리소스에만 적용됩니다.

Azure OpenAI 리소스에 연결하는 검색 리소스에서 공유 프라이빗 링크를 만들려면 [검색 설명서](#)를 참조하세요. 리소스 종류를 `Microsoft.CognitiveServices/accounts`로 그룹 ID 를 `openai_account`로 선택합니다.

공유 프라이빗 링크를 사용하면 데이터 수집 아키텍처 디아이어그램의 [8단계](#)가 신뢰할 수 있는 서비스 우회에서 **프라이빗 엔드포인트**로 변경됩니다.



만든 Azure AI 검색 공유 프라이빗 링크는 가상 네트워크가 아닌 Microsoft 관리형 가상 네트워크에도 있습니다. 앞서 만든 다른 관리형 프라이빗 엔드포인트와의 차이점은 Azure OpenAI에서 Azure Search까지의 관리형 프라이빗 엔드포인트 [1] 가 **양식 애플리케이션**을 통해 프로비전되는 반면, Azure Cognitive Search에서 Azure OpenAI로의 관리형 프라이빗 엔드포인트 [2] 는 Azure Portal 또는 Azure Cognitive Search의 REST API를 통해 프로비전된다는 점입니다.



스토리지 계정 구성

신뢰할 수 있는 서비스 사용

Azure OpenAI 및 Azure AI Search에서 스토리지 계정에 대한 액세스를 허용하려면 스토리지 계정에는 공용 네트워크 액세스가 없지만 Azure OpenAI 및 Azure AI Search를 관리 ID 기반의 신뢰할 수 있는 서비스로 우회하도록 스토리지 계정을 설정해야 합니다.

Azure Portal에서 스토리지 계정 네트워킹 탭으로 이동하여 "선택한 네트워크"를 선택한 다음 신뢰할 수 있는 서비스 목록의 Azure 서비스가 이 스토리지 계정에 액세스하도록 허용을 선택하고 저장을 클릭합니다.

① 참고

신뢰할 수 있는 서비스 기능은 위에서 설명한 명령줄에서만 사용할 수 있으며 Azure Portal에서는 수행할 수 없습니다.

공용 네트워크 액세스 사용 안 함

Azure Portal에서 스토리지 계정의 공용 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다.

Azure OpenAI Studio를 사용하는 것과 같이 클라이언트 컴퓨터에서 스토리지 계정에 대한 액세스를 허용하려면 Blob Storage에 연결하는 [프라이빗 엔드포인트 연결](#)을 만들어야 합니다.

역할 할당

지금까지는 이미 각 리소스 작업을 독립적으로 설정했습니다. 다음으로 서비스가 서로 권한 부여할 수 있도록 허용해야 합니다.

[] 테이블 확장

역할	담당자	리소스	설명
Search Index Data Reader	Azure OpenAI	Azure AI 검색	유추 서비스는 인덱스에서 데이터를 쿼리합니다.
Search Service Contributor	Azure OpenAI	Azure AI 검색	유추 서비스는 자동 필드 매핑을 위해 인덱스 스키마를 쿼리합니다. 데이터 수집 서비스는 인덱스, 데이터 원본, 기술 집합, 인덱서를 만들고 인덱서 상태를 쿼리합니다.
Storage Blob Data Contributor	Azure OpenAI	스토리지 계정	입력 컨테이너에서 읽고 전처리 결과를 출력 컨테이너에 씁니다.

역할	담당자	리소스	설명
Cognitive Services Contributor	Azure AI 검색	Azure OpenAI	사용자 지정 기술
Storage Blob Data Contributor	Azure AI 검색	스토리지 계정	BLOB을 읽고 지식 저장소를 씁니다.

위 표에서 **Assignee**는 해당 리소스의 시스템이 할당한 관리 ID를 의미합니다.

역할 할당을 추가하려면 관리자에게 이러한 리소스에 대한 **Owner** 역할이 있어야 합니다.

Azure Portal에서 이러한 역할을 설정하는 방법에 대한 자침은 [Azure RBAC 설명서](#)를 참조하세요. [GitHub에서 사용 가능한 스크립트](#)를 사용하여 프로그래밍 방식으로 역할 할당을 추가할 수 있습니다.

개발자가 이러한 리소스를 사용하여 애플리케이션을 빌드할 수 있도록 하려면 관리자는 다음 역할 할당을 사용하여 개발자의 ID를 리소스에 추가해야 합니다.

[+] 테이블 확장

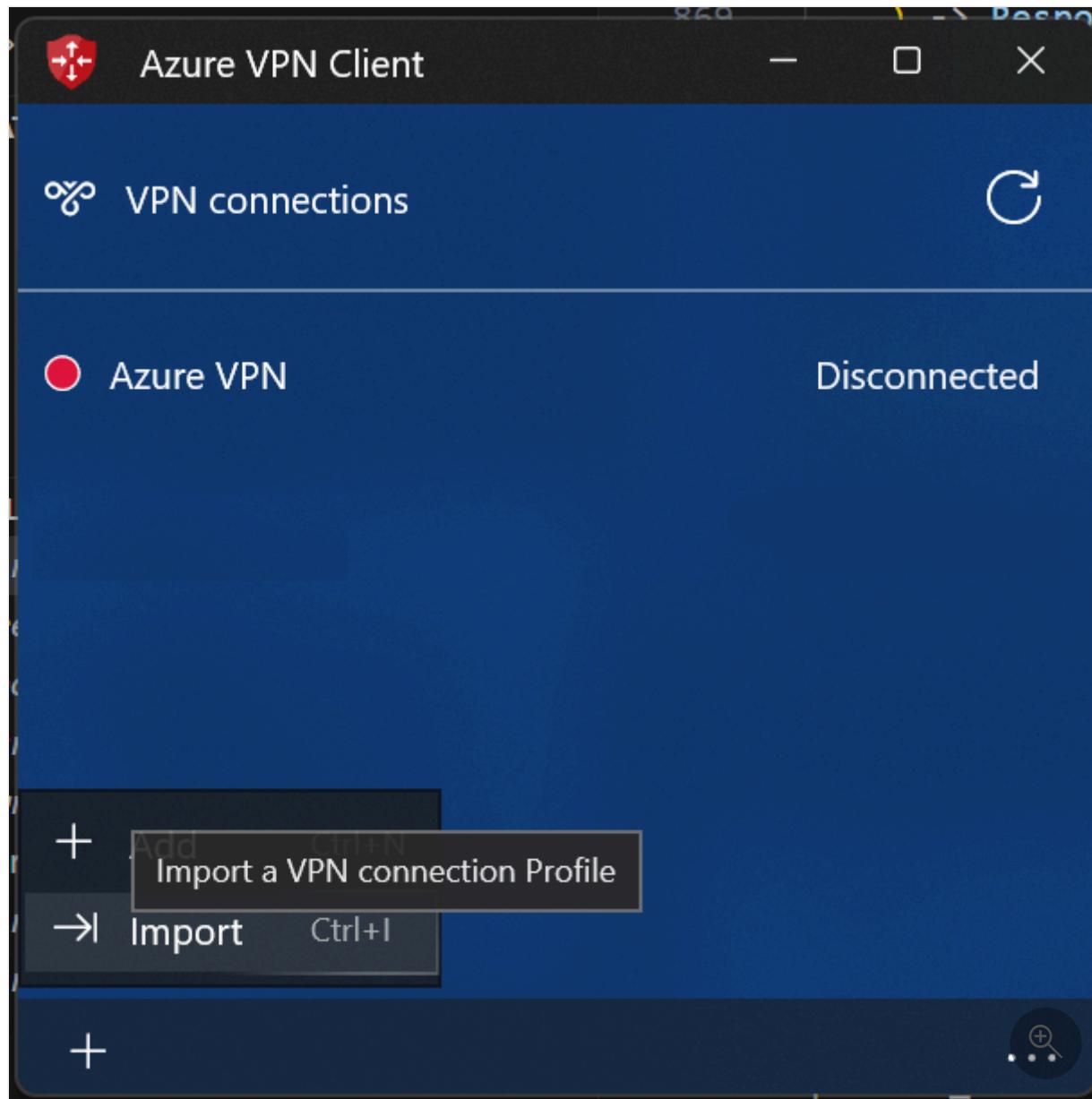
역할	리소스	설명
Cognitive Services Contributor	Azure OpenAI	Azure OpenAI Studio에서 공용 수집 API를 호출합니다. Contributor 역할만으로는 충분하지 않습니다. 왜냐하면 Contributor 역할만 있으면 Microsoft Entra ID 인증을 통해 데이터 평면 API를 호출할 수 없고 이 문서에 설명된 보안 설정에는 Microsoft Entra ID 인증이 필요하기 때문입니다.
Cognitive Services User	Azure OpenAI	Azure OpenAI Studio의 API 키를 나열합니다.
Contributor	Azure AI 검색	Azure OpenAI Studio의 인덱스를 나열하려면 API 키를 나열합니다.
Contributor	스토리지 계정	Azure OpenAI Studio에서 파일을 업로드하려면 계정 SAS를 나열합니다.
Contributor	개발자가 웹앱을 배포해야 하는 리소스 그룹 또는 Azure 구독	개발자의 Azure 구독에 웹앱을 배포합니다.

게이트웨이 및 클라이언트 구성

온-프레미스 클라이언트 컴퓨터에서 Azure OpenAI 서비스에 액세스하기 위한 방식 중 하나는 Azure VPN Gateway 및 Azure VPN Client를 구성하는 것입니다.

가상 네트워크용 가상 네트워크 게이트웨이를 만들려면 [이 지침](#)을 따릅니다.

지점 및 사이트 간 구성을 추가하고 Microsoft Entra ID 기반 인증을 사용하도록 설정하려면 [이 지침](#)을 따릅니다. Azure VPN Client 프로필 구성 패키지를 다운로드하고, 압축을 풀고, `AzureVPN/azurevpnconfig.xml` 파일을 Azure VPN Client로 가져옵니다.



리소스 호스트 이름이 가상 네트워크의 개인 IP를 가리키도록 로컬 컴퓨터 `hosts` 파일을 구성합니다. `hosts` 파일은 Windows의 경우 `C:\Windows\System32\drivers\etc`에 있고 Linux의 경우 `/etc/hosts`에 있습니다. 예시:

```
10.0.0.5 contoso.openai.azure.com
10.0.0.6 contoso.search.windows.net
```

Azure OpenAI Studio

온-프레미스 클라이언트 컴퓨터에서 수집 및 유추를 포함한 모든 Azure OpenAI Studio 기능을 사용할 수 있어야 합니다.

웹 앱

웹 앱은 Azure OpenAI 리소스와 통신합니다. Azure OpenAI 리소스에는 공용 네트워크가 사용하지 않도록 설정되어 있으므로 Azure OpenAI 리소스에 액세스하려면 가상 네트워크의 프라이빗 엔드포인트를 사용하도록 웹 앱을 설정해야 합니다.

웹 앱은 Azure OpenAI 호스트 이름을 Azure OpenAI용 프라이빗 엔드포인트의 개인 IP로 확인해야 합니다. 따라서 먼저 가상 네트워크에 대한 프라이빗 DNS 영역을 구성해야 합니다.

- 리소스 그룹에 [프라이빗 DNS 영역을 만듭니다.](#)
- [DNS 레코드를 추가합니다.](#) IP는 Azure OpenAI 리소스에 대한 프라이빗 엔드포인트의 개인 IP이며, Azure OpenAI에 대한 프라이빗 엔드포인트와 연결된 네트워크 인터페이스에서 IP 주소를 가져올 수 있습니다.
- [프라이빗 DNS 영역을 가상 네트워크에 연결하면](#) 이 가상 네트워크에 통합된 웹 앱이 이 프라이빗 DNS 영역을 사용할 수 있습니다.

Azure OpenAI Studio에서 웹 앱을 배포할 때 가상 네트워크와 동일한 위치를 선택하고 적절한 SKU를 선택하면 [가상 네트워크 통합 기능](#)을 지원할 수 있습니다.

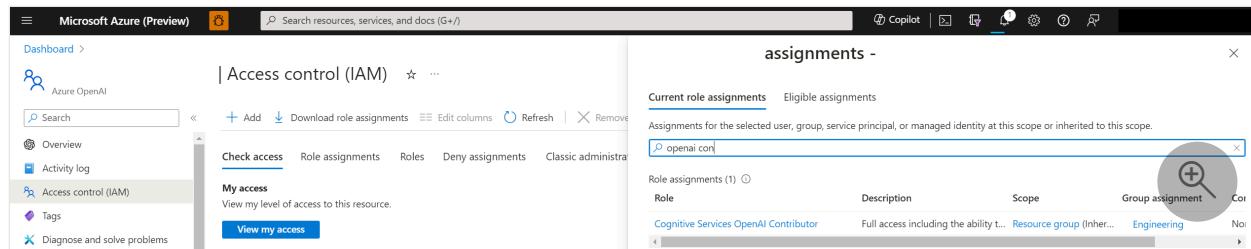
웹 앱이 배포된 후 Azure Portal 네트워킹 탭에서 웹 앱 아웃바운드 트래픽 가상 네트워크 통합을 구성하고 웹 앱용으로 예약한 세 번째 서브넷을 선택합니다.

The screenshot shows the Azure Portal interface for managing a web application named "webapp". The "Networking" tab is currently selected. At the top, there is a search bar with the text "net". Below the search bar, there are several navigation links: "Refresh", "Troubleshoot", and "Send us your feedback". The main content area displays the "Outbound traffic configuration" settings, which include "Virtual network integration" set to "vnet/webapp" and "Hybrid connections" set to "Not configured". On the left side, there is a sidebar with links for "Settings", "Environment variables", "Configuration", and "Networking". The "Networking" link is highlighted with a grey background.

API 사용

로그인 자격 증명에 Azure OpenAI 리소스에 대한 Cognitive Services OpenAI

Contributor 역할이 있는지 확인하고 먼저 az login을 실행합니다.



수집 API

수집 API에서 사용하는 요청 및 응답 개체에 대한 자세한 내용은 [수집 API 참조 문서](#)를 참조하세요.

추가 참고 사항:

- API 경로의 `JOB_NAME`은 Azure AI 검색에서 인덱스 이름으로 사용됩니다.
- api-key 대신 `Authorization` 헤더를 사용합니다.
- `storageEndpoint` 헤더를 명시적으로 설정합니다.
- `storageConnectionString` 헤더에 `ResourceId=` 형식을 사용하므로 Azure OpenAI 및 Azure AI 검색은 관리 ID를 사용하여 네트워크 제한을 무시하는 데 필요한 스토리지 계정을 인증합니다.
- `searchServiceAdminKey` 헤더를 설정하지 **마세요**. Azure OpenAI 리소스의 시스템 할당 ID는 Azure AI 검색을 인증하는 데 사용됩니다.
- `embeddingEndpoint` 또는 `embeddingKey`를 설정하지 **마세요**. 대신 `embeddingDeploymentName` 헤더를 사용하여 텍스트 벡터화를 사용하도록 설정합니다.

작업 제출 예

```
Bash

accessToken=$(az account get-access-token --resource
https://cognitiveservices.azure.com/ --query "accessToken" --output tsv)
curl -i -X PUT https://my-resource.openai.azure.com/openai/extensions/on-
your-data/ingestion-jobs/vpn1025a?api-version=2023-10-01-preview \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $accessToken" \
-H "storageEndpoint: https://mystorage.blob.core.windows.net/" \
-H "storageConnectionString: ResourceId=/subscriptions/1234567-abcd-1234-
5678-1234abcd/resourceGroups/my-
resource/providers/Microsoft.Storage/storageAccounts/mystorage" \
-H "storageContainer: my-container" \
-H "searchServiceEndpoint: https://mysearch.search.windows.net" \
-H "embeddingDeploymentName: ada" \
```

```
-d \
'
{
}
'
```

작업 상태 가져오기 예

Bash

```
accessToken=$(az account get-access-token --resource
https://cognitiveservices.azure.com/ --query "accessToken" --output tsv)
curl -i -X GET https://my-resource.openai.azure.com/openai/extensions/on-
your-data/ingestion-jobs/abc1234?api-version=2023-10-01-preview \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $accessToken"
```

유추 API

유추 API에서 사용하는 요청 및 응답 개체에 대한 자세한 내용은 [유추 API 참조 문서](#)를 참조하세요.

Azure OpenAI Service 리소스 생성 및 배포

아티클 • 2024. 01. 10.

이 문서에서는 Azure OpenAI Service 를 시작하는 방법을 설명하고 리소스를 만들고 모델을 배포하는 단계별 지침을 제공합니다. 여러 가지 방법으로 Azure에서 리소스를 만들 수 있습니다.

- [Azure Portal](#)
- REST API, Azure CLI, PowerShell 또는 클라이언트 라이브러리
- ARM(Azure Resource Manager) 템플릿

이 문서에서는 Azure portal 및 Azure CLI를 사용하여 리소스를 만들고 배포하는 예제를 검토합니다.

필수 구성 요소

- Azure 구독 [체험 계정 만들기](#)
- 원하는 Azure 구독에서 Azure OpenAI에 부여된 액세스 권한.
- Azure OpenAI 리소스를 만들고 모델을 배포하기 위한 권한에 액세스합니다.

① 참고

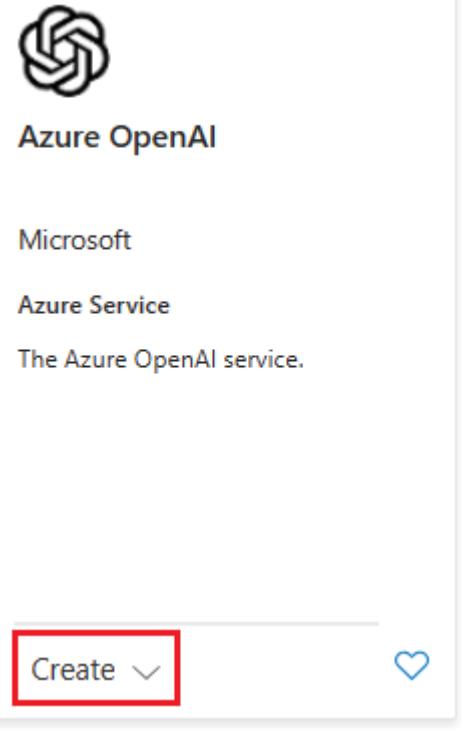
현재 Azure OpenAI Service 에 액세스하려면 신청서를 제출해야 합니다. 액세스를 신청하려면 [이 양식](#)을 작성하세요. 도움이 필요한 경우 이 리포지토리에서 문제를 열어 Microsoft에 문의하세요.

리소스 만들기

다음 단계에서는 Azure portal에서 Azure OpenAI 리소스를 만드는 방법을 보여줍니다.

리소스 식별

1. Azure portal에서 Azure 구독으로 로그인합니다.
2. 리소스 만들기를 선택하고 Azure OpenAI를 검색합니다. 서비스를 찾으면 만들기를 선택합니다.



3. Azure OpenAI 만들기 페이지에서 **기본 사항** 탭의 필드에 다음 정보를 제공합니다.

[+] 테이블 확장

필드	Description
구독	Azure OpenAI Service 온보딩 애플리케이션에 사용되는 Azure 구독입니다.
리소스 그룹	Azure OpenAI 리소스를 포함할 Azure 리소스 그룹입니다. 새 그룹을 만들거나 기존 그룹을 사용할 수 있습니다.
지역	인스턴스의 위치입니다. 위치에 따라 지역 시간이 발생할 수 있지만 리소스의 런타임 가용성에는 영향을 미치지 않습니다.
이름	<i>MyOpenAIResource</i> 와 같이 Azure OpenAI Service 리소스를 설명하는 이름입니다.
가격 책정 계층	리소스의 가격 책정 계층입니다. 현재 표준 계층만 Azure OpenAI Service에 사용할 수 있습니다. 가격 책정에 대한 자세한 내용은 Azure OpenAI 가격 책정 페이지를 참조하세요. ↗

Create Azure OpenAI

Basics

Network

Tags

Review + submit

Enable new business solutions with OpenAI's language generation capabilities powered by GPT-3 models. These models have been pretrained with trillions of words and can easily adapt to your scenario with a few short examples provided at inference. Apply them to numerous scenarios, from summarization to content and code generation.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

OpenAI Test Subscription

Resource group * ⓘ

test-resource-group

[Create new](#)

Instance details

Region * ⓘ

South Central US

Name * ⓘ

azure-openai-test-001

Pricing tier * ⓘ

Standard S0

[View full pricing details](#)

Content review policy

To detect and mitigate harmful use of the Azure OpenAI Service, Microsoft logs the content you send to the Completions and image generations APIs as well as the content it sends back. If content is flagged by the service's filters, it may be reviewed by a Microsoft full-time employee.

[Learn more about how Microsoft processes, uses, and stores your data](#)

[Apply for modified content filters and abuse monitoring](#)

[Review the Azure OpenAI code of conduct](#)

[Previous](#)

[Next](#)

4. 다음을 선택합니다.

네트워크 보안 구성

네트워크 탭에는 보안 유형에 대한 세 가지 옵션이 표시됩니다.

- 옵션 1: 인터넷을 포함한 모든 네트워크가 이 리소스에 액세스할 수 있습니다.
- 옵션 2: 선택한 네트워크에서 Azure AI 서비스 리소스에 대한 네트워크 보안을 구성합니다.
- 옵션 3: 사용 중지됨, 어떤 네트워크도 이 리소스에 액세스할 수 없습니다. 이 리소스에 액세스하는 유일한 방법이 될 프라이빗 엔드포인트 연결을 구성할 수 있습니다.

Create Azure OpenAI

Basics Network Tags Review + submit

Configure network security for your Azure AI services resource.

Type *

All networks, including the internet, can access this resource.

Selected networks, configure network security for your Azure AI services resource.

Disabled, no networks can access this resource. You could configure private endpoint connections that will be the exclusive way to access this resource.

선택한 옵션에 따라 추가 정보를 제공해야 할 수 있습니다.

옵션 1: 모든 네트워크 허용

첫 번째 옵션을 사용하면 인터넷을 포함한 모든 네트워크에서 리소스에 액세스할 수 있습니다. 이 옵션은 기본 설정입니다. 이 옵션에는 추가 설정이 필요하지 않습니다.

옵션 2: 특정 네트워크만 허용

두 번째 옵션을 사용하면 리소스에 액세스할 수 있는 특정 네트워크를 식별할 수 있습니다. 이 옵션을 선택하면 다음 필수 필드를 포함하도록 페이지가 업데이트됩니다.

[+] 테이블 확장

필드	설명
가상 네트워크	리소스에 대한 액세스가 허용되는 가상 네트워크를 지정합니다. Azure portal에서 기본 가상 네트워크 이름을 편집할 수 있습니다.
서브넷	리소스에 대한 액세스가 허용되는 서브넷을 지정합니다. Azure portal에서 기본 서브넷 이름을 편집할 수 있습니다.

Type *

All networks, including the internet, can access this resource.

Selected networks, configure network security for your Azure AI services resource.

Disabled, no networks can access this resource. You could configure private endpoint connections that will be the exclusive way to access this resource.

Virtual network *

(New) vnet01 (test-resource-group) ▼

[Edit virtual network](#)

Subnets *

(New) subnet-1 ▼

[Edit subnet](#) 172.18.0.0 - 172.18.0.63 (64 addresses)

Firewall

Add IP ranges to allow access from the internet or your on-premises networks.

[Learn more](#)

Address range

방화벽 섹션에서는 리소스에 대한 방화벽 설정을 구성하는 데 사용할 수 있는 선택적 주소 범위 필드를 제공합니다.

옵션 3: 네트워크 액세스 사용 안 함

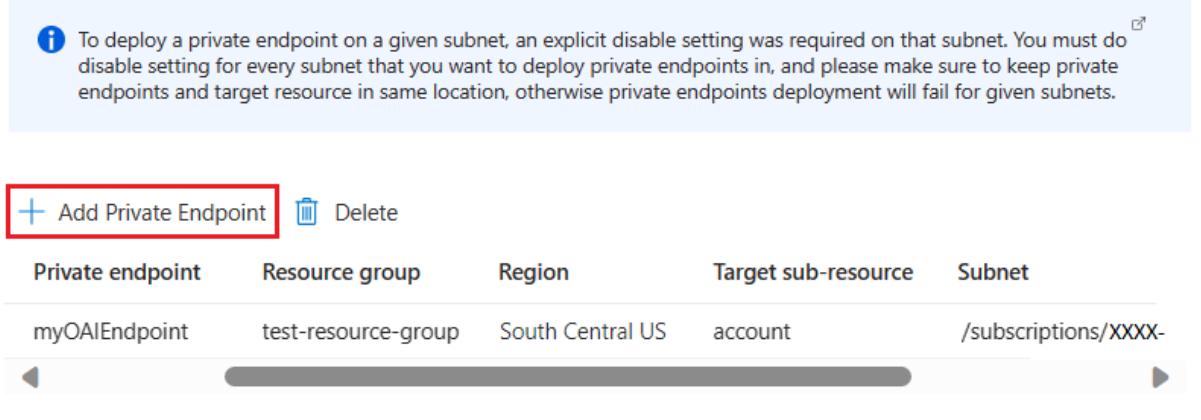
세 번째 옵션을 사용하면 리소스에 대한 네트워크 액세스를 사용하지 않도록 설정할 수 있습니다. 이 옵션을 선택하면 페이지가 업데이트되어 **프라이빗 엔드포인트** 테이블을 포함합니다.

- Type *
- All networks, including the internet, can access this resource.
 - Selected networks, configure network security for your Azure AI services resource.
 - Disabled, no networks can access this resource. You could configure private endpoint connections that will be the exclusive way to access this resource.

Private endpoint

Create a private endpoint to allow a private connection to this resource. Please make sure that the private endpoint has the same location as this resource. Additional private endpoint connections can be created within the Azure AI services account or private link center.

i To deploy a private endpoint on a given subnet, an explicit disable setting was required on that subnet. You must do disable setting for every subnet that you want to deploy private endpoints in, and please make sure to keep private endpoints and target resource in same location, otherwise private endpoints deployment will fail for given subnets.



 Add Private Endpoint	 Delete			
Private endpoint	Resource group	Region	Target sub-resource	Subnet
myOAIEndpoint	test-resource-group	South Central US	account	/subscriptions/XXXX-

선택적으로 리소스에 액세스하기 위한 프라이빗 엔드포인트를 추가할 수 있습니다. **프라이빗 엔드포인트 추가**를 선택하고 엔드포인트 구성을 완료합니다.

구성 확인 및 리소스 생성

1. 다음을 선택하고 원하는 대로 리소스에 대한 태그를 구성합니다.
2. 다음을 선택하여 프로세스의 마지막 단계인 검토 + 제출로 이동합니다.
3. 구성 설정을 확인하고 만들기를 선택합니다.

Azure portal은 새 리소스를 사용할 수 있을 때 알림을 표시합니다.

모델 배포

텍스트 또는 추론을 생성하려면 먼저 모델을 배포해야 합니다. Azure OpenAI Studio에서 사용 가능한 여러 모델 중 하나를 선택할 수 있습니다.

모델을 배포하려면 다음 단계를 수행합니다.

1. Azure OpenAI Studio에 [로그인합니다](#).
2. 작업할 구독과 Azure OpenAI 리소스를 선택하고 **리소스 사용**을 선택합니다.

3. 관리에서 배포를 선택합니다.

4. 새 배포 만들기를 선택하고 다음 필드를 구성합니다.

[:] 테이블 확장

필드	설명
모델 선택	모델 가용성은 지역에 따라 다릅니다. 지역별로 사용 가능한 모델 목록을 보려면 모델 요약 표 및 지역 가용성 을 참조하세요.
배포 이름	이름을 신중하게 선택합니다. 배포 이름은 코드에서 클라이언트 라이브러리 및 REST API를 사용하여 모델을 호출하는 데 사용됩니다.
고급 옵션 (선택 사항)	리소스에 필요한 대로 선택적 고급 설정을 지정할 수 있습니다. - 콘텐츠 필터 의 경우 배포에 콘텐츠 필터를 할당합니다. - 분당 토큰 비율 제한 의 경우 분당 토큰(TPM)을 조정하여 배포에 대한 효과적인 비율 제한을 설정합니다. 할당량 메뉴를 사용하여 언제든지 이 값을 수정할 수 있습니다.

5. 드롭다운 목록에서 모델을 선택합니다.

6. 모델을 식별할 배포 이름을 입력합니다.

ⓘ 중요

API를 통해 모델에 액세스하는 경우 API 호출의 기본 모델 이름이 아닌 배포 이름을 참조해야 합니다. 이는 OpenAI와 Azure OpenAI의 **주요 차이점** 중 하나입니다. OpenAI에는 모델 이름만 필요합니다. Azure OpenAI는 모델 매개 변수를 사용하는 경우에도 항상 배포 이름이 필요합니다. 문서에는 특정 API 엔드포인트에서 작동하는 모델을 나타내는데 도움이 되는 모델 이름과 동일한 배포 이름이 표시되는 예제가 있는 경우가 많습니다. 궁극적으로 배포 이름은 사용 사례에 가장 적합한 명명 규칙을 따를 수 있습니다.

7. 처음 배포하는 경우 고급 옵션을 기본값으로 둡니다.

8. 만들기를 실행합니다.

배포 테이블에는 새로 생성된 모델에 해당하는 새 항목이 표시됩니다.

배포가 완료되면 모델 배포 상태가 성공으로 변경됩니다.

다음 단계

- Azure OpenAI Service 빠른 시작을 사용하여 API를 호출하고 텍스트를 생성하세요.

- Azure OpenAI Service 모델에 대해 자세히 알아보세요.
- 가격 책정에 대한 자세한 내용은 Azure OpenAI 가격 책정 페이지를 참조하세요. ↗

Python을 사용하여 OpenAI와 Azure OpenAI 엔드포인트 간에 전환하는 방법

아티클 • 2024. 03. 10.

OpenAI 및 Azure OpenAI Service는 일반적인 [Python 클라이언트 라이브러리](#)에 의존하지만 엔드포인트 간에 교환하기 위해 코드를 약간 변경해야 합니다. 이 문서에서는 OpenAI 및 Azure OpenAI에서 작업할 때 발생하는 일반적인 변경 내용과 차이점을 안내합니다.

이 문서에서는 새 OpenAI Python 1.x API 라이브러리를 사용하는 예제만 보여 줍니다. [0.28.1](#)에서 [1.x\(으\)로의 마이그레이션](#)에 대한 자세한 내용은 [마이그레이션 가이드](#)를 참조하세요.

인증

환경 변수를 사용하는 것이 좋습니다. [Python 빠른 시작](#) 전에 이렇게 하지 않았다면 이 구성을 안내해드립니다.

API 키

 테이블 확장

OpenAI	Azure OpenAI
<pre>import os from openai import OpenAI client = OpenAI(api_key=os.getenv("OPENAI_API_KEY"))</pre>	<pre>import os from openai import AzureOpenAI client = AzureOpenAI(api_key=os.getenv("AZURE_OPENAI_API_KEY"), api_version="2023-12-01-preview", azure_endpoint=os.getenv("AZURE_OPENAI_ENDPOINT"))</pre>

Microsoft Entra ID 인증

OpenAI

```
import os
from openai import
OpenAI

client = OpenAI(
    api_key=os.getenv("OPEN
AI_API_KEY")
)
```

Azure OpenAI

```
from azure.identity import
DefaultAzureCredential,
get_bearer_token_provider
from openai import AzureOpenAI

token_provider = get_bearer_token_provider(
    DefaultAzureCredential(),
    "https://cognitiveservices.azure.com/.defaul
t"
)

api_version = "2023-12-01-preview"
endpoint = "https://my-
resource.openai.azure.com"

client = AzureOpenAI(
    api_version=api_version,
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
)
```

모델의 키워드 인수

OpenAI는 `model` 키워드 인수를 사용하여 사용할 모델을 지정합니다. Azure OpenAI에는 고유한 모델 **배포** 개념이 있습니다. Azure OpenAI `model`(을)를 사용하는 경우 모델을 배포할 때 선택한 기본 배포 이름을 참조해야 합니다.

① 중요

Azure OpenAI에서 API를 통해 모델에 액세스하는 경우 API 호출의 기본 모델 이름이 아닌 배포 이름을 참조해야 합니다. 이는 OpenAI와 Azure OpenAI 간의 **주요 차이점** 중 하나입니다. OpenAI에는 모델 이름만 필요하며, Azure OpenAI에는 모델 매개 변수를 사용하는 경우에도 항상 배포 이름이 필요합니다. 문서에는 특정 API 엔드포인트에서 작동하는 모델을 나타내는 데 도움이 되는 모델 이름과 동일한 배포 이름이 표시되는 예제가 있는 경우가 많습니다. 궁극적으로 배포 이름은 사용 사례에 가장 적합한 명명 규칙을 따를 수 있습니다.

OpenAI

Azure OpenAI

```
completion =  
client.completions.cr  
eate(  
    model="gpt-3.5-  
turbo-instruct",  
    prompt=<prompt>  
)  
  
chat_completion =  
client.chat.completo  
ns.create(  
    model="gpt-4",  
    messages="  
<messages>"  
)  
  
embedding =  
client.embeddings.cre  
ate(  
    model="text-  
embedding-ada-002",  
    input=<input>  
)
```

```
completion = client.completions.create(  
    model="gpt-35-turbo-instruct", # This  
    must match the custom deployment name you  
    chose for your model.  
    prompt=<prompt>  
)  
  
chat_completion =  
client.chat.completions.create(  
    model="gpt-35-turbo", # model =  
    "deployment_name".  
    messages=<messages>  
)  
  
embedding = client.embeddings.create(  
    model="text-embedding-ada-002", # model =  
    "deployment_name".  
    input=<input>  
)
```

다중 입력 지원을 포함하는 Azure OpenAI

OpenAI 및 Azure OpenAI는 현재 text-embedding-ada-002에 대해 최대 2048개의 입력 항목 배열을 지원합니다. 둘 다 이 모델에 대해 8191 미만으로 유지하려면 API 요청당 최대 입력 토큰 제한이 필요합니다.

 테이블 확장

OpenAI

Azure OpenAI

```
inputs = ["A",  
"B", "C"]  
  
embedding =  
client.embeddings  
.create(  
    input=inputs,  
    model="text-  
embedding-ada-  
002"
```

```
inputs = ["A", "B", "C"] #max array size=2048  
  
embedding = client.embeddings.create(  
    input=inputs,  
    model="text-embedding-ada-002" # This must  
    match the custom deployment name you chose for  
    your model.  
    # engine="text-embedding-ada-002"
```

다음 단계

- 방법 [가이드](#)를 통해 GPT-3.5-Turbo 및 GPT-4 모델로 작업하는 방법에 대해 자세히 알아봅니다.
- 더 많은 예제를 보려면 [Azure OpenAI 샘플 GitHub 리포지토리](#)를 체크 아웃합니다.

Azure OpenAI 서비스 할당량 관리

아티클 • 2024. 01. 10.

할당량은 구독 내에서 배포 전반에 걸쳐 비율 제한 할당을 적극적으로 관리할 수 있는 유연성을 제공합니다. 이 문서에서는 Azure OpenAI 할당량을 관리하는 프로세스를 안내합니다.

필수 조건

① 중요

할당량을 보고 모델을 배포하려면 **Cognitive Services 사용량 읽기 권한자** 역할이 필요합니다. 이 역할은 Azure 구독 전체의 할당량 사용량을 보는 데 필요한 최소한의 액세스 권한을 제공합니다. 이 역할과 Azure OpenAI에 액세스하는 데 필요한 다른 역할에 대해 자세히 알아보려면 [Azure RBAC\(Azure 역할 기반 액세스\) 가이드](#)를 참조하세요.

이 역할은 Azure Portal의 **구독>IAM(액세스 제어)>역할 할당 추가>** 검색에서 찾을 수 있습니다. **Cognitive Services 사용량 읽기 권한자**. 이 역할은 구독 수준에서 적용되어야 하며 리소스 수준에는 존재하지 않습니다.

이 역할을 사용하지 않으려면 구독 **읽기 권한자** 역할이 동등한 액세스 권한을 제공하지만 할당량 및 모델 배포를 보는 데 필요한 범위를 넘어서는 읽기 권한도 부여합니다.

할당량 소개

Azure OpenAI의 할당량 기능을 사용하면 "할당량"이라는 전역 제한까지 배포에 속도 제한을 할당할 수 있습니다. 할당량은 **TPM(분당 토큰)** 단위로 지역별, 모델별로 구독에 할당됩니다. Azure OpenAI 구독을 온보딩하면 사용할 수 있는 대부분의 모델에 대한 기본 할당량을 받게 됩니다. 그런 다음 배포가 만들어질 때 각 배포에 TPM을 할당하면 해당 모델에 사용할 수 있는 할당량이 그만큼 줄어듭니다. 할당량 한도에 도달할 때까지 계속해서 배포를 만들고 TPM을 할당할 수 있습니다. 그런 일이 발생하면 동일한 모델의 다른 배포에 할당된 TPM을 줄이거나(따라서 TPM을 사용할 수 있게 됨) 원하는 지역에서 모델 할당량 증가를 요청하고 승인받아 해당 모델의 새 배포를 만들 수 있습니다.

② 참고

미국 동부의 GPT-35-Turbo에 대해 240,000TPM 할당량을 사용하여 고객은 240K TPM의 단일 배포, 각각 120K TPM의 2개 배포 또는 하나 이상의 Azure OpenAI 리소스에 원하는 수의 배포를 만들 수 있습니다. 해당 지역의 TPM 합계는 총 240K 미만입니다.

배포가 만들어지면 할당된 TPM은 유추 요청에 적용되는 분당 토큰 속도 제한에 직접 매핑됩니다. **RPM(분당 요청)** 속도 제한도 적용되며 이 값은 다음 비율을 사용하여 TPM 할당에 비례하여 설정됩니다.

1000TPM당 6RPM.

구독 및 지역 내에서 TPM을 전역적으로 배포할 수 있는 유연성을 통해 Azure OpenAI Service는 다른 제한 사항을 완화할 수 있었습니다.

- 지역당 최대 리소스가 30개로 늘어났습니다.
- 리소스에 동일한 모델의 배포를 두 개만 만들 수 있는 제한이 제거되었습니다.

할당량 할당

모델 배포를 만들 때 해당 배포에 TPM(분당 토큰)을 할당하는 옵션이 있습니다. TPM은 1,000 단위로 수정할 수 있으며 위에서 설명한 대로 배포에 적용되는 TPM 및 RPM 속도 제한에 매핑됩니다.

Azure AI Studio 내에서 관리 아래에서 새 배포를 만들려면 **배포>새 배포 만들기**를 선택합니다.

TPM 설정 옵션은 **고급 옵션** 드롭다운에 있습니다.

Deploy model

X

Set up a deployment to make API calls against a provided base model or a custom model. Finished deployments are available for use. Your deployment status will move to succeeded when the deployment is complete and ready for use.

Select a model ⓘ

text-similarity-ada-001

Deployment name ⓘ

text-similarity-ada-001 *

Advanced options ▾

Content Filter ⓘ

Default

(i) 120K tokens per minute quota available for your deployment

Tokens per Minute Rate Limit (thousands) ⓘ

120K

Corresponding requests per minute (RPM) = 720

Create

Cancel



배포 후에는 Azure AI Studio의 관리 > 배포에서 배포 편집을 선택하여 TPM 할당을 조정할 수 있습니다. 관리 > 할당량 아래의 새로운 할당량 관리 환경 내에서 이 선택을 수정할 수도 있습니다.

ⓘ 중요

할당량 및 제한은 변경될 수 있습니다. 최신 정보를 보려면 [할당량 및 제한 문서](#)를 참조하세요.

모델별 설정

모델 클래스라고도 하는 다양한 모델 배포에는 이제 제어할 수 있는 고유한 최대 TPM 값이 있습니다. 이는 특정 지역에서 해당 형식의 모델 배포에 할당할 수 있는 최대 TPM 양

을 나타냅니다. 각 모델 형식은 고유한 모델 클래스를 나타내지만 최대 TPM 값은 현재 특정 모델 클래스에서만 다릅니다.

- GPT-4
- GPT-4-32K
- Text-Davinci-003

다른 모든 모델 클래스에는 공통된 최대 TPM 값이 있습니다.

① 참고

TPM(분당 토큰 할당량) 할당은 모델의 최대 입력 토큰 제한과 관련이 없습니다. 모델 입력 토큰 제한은 **모델 테이블**에 정의되어 있으며 TPM 변경 내용의 영향을 받지 않습니다.

할당량 보기 및 요청

특정 지역의 배포 전체에 대한 할당량 할당을 모두 보려면 Azure AI Studio에서 관리 > 할당량을 선택합니다.

The screenshot shows the Azure AI Studio interface with the 'Quotas' section selected. On the left sidebar, the 'Quotas' item is highlighted with a red box. The main content area displays a table of quota usage for different deployment types across regions. Each row shows the quota name, deployment type, usage limit, current usage, and a progress bar indicating the percentage used. A large circular button with a magnifying glass icon is visible in the bottom right corner of the table area.

Quota name	Deployment	Usage/Limit	Request quota
Tokens Per Minute (thousands) - Text-Similarity-Da	120 of 240	50%	[button]
Tokens Per Minute (thousands) - Text-Similarity-Ad	120 of 240	50%	[button]
Tokens Per Minute (thousands) - Text-Search-Davin	120 of 240	50%	[button]
Tokens Per Minute (thousands) - Text-Search-Davin	120 of 240	50%	[button]
Tokens Per Minute (thousands) - Text-Search-Ada-(120 of 240	50%	[button]
Tokens Per Minute (thousands) - Text-Search-Ada-[120 of 240	50%	[button]
Tokens Per Minute (thousands) - Text-Embedding-/-	361 of 361	100%	[button]

- **할당량 이름:** 각 모델 형식에 대해 지역당 하나의 할당량 값이 있습니다. 할당량에는 해당 모델의 모든 버전이 포함됩니다. 할당량 이름을 UI에서 확장하여 할당량을 사

용하는 배포를 표시할 수 있습니다.

- **배포**: 모델 배포를 모델 클래스로 나눕니다.
- **사용량/제한**: 할당량 이름의 경우 배포에 사용된 할당량과 이 구독 및 지역에 승인된 총 할당량이 표시됩니다. 사용된 할당량은 막대 그래프에도 표시됩니다.
- **할당량 요청**: 이 필드의 아이콘은 할당량 증가 요청을 제출할 수 있는 양식으로 이동합니다.

기존 배포 마이그레이션

새로운 할당량 시스템 및 TPM 기반 할당으로의 전환에 대한 일환으로 모든 기존 Azure OpenAI 모델 배포는 할당량을 사용하도록 자동으로 마이그레이션되었습니다. 이전 사용자 지정 비율 제한 증가로 인해 기존 TPM/RPM 할당이 기본값을 초과하는 경우 영향을 받는 배포에 동등한 TPM이 할당되었습니다.

속도 제한 이해

배포에 TPM을 할당하면 위에서 설명한 대로 배포에 대한 TPM(분당 토큰) 및 RPM(분당 요청) 속도 제한이 설정됩니다. TPM 속도 제한은 요청이 수신될 때 요청에 의해 처리될 것으로 예상되는 최대 토큰 수를 기반으로 합니다. 모든 처리가 완료된 후 계산되는 청구에 사용되는 토큰 개수와는 다릅니다.

각 요청이 수신되면 Azure OpenAI는 다음을 포함하는 예상 최대 처리 토큰 수를 계산합니다.

- 프롬프트 텍스트 및 개수
- max_tokens 매개 변수 설정
- best_of 매개 변수 설정

요청이 배포 엔드포인트로 들어오면 예상되는 최대 처리 토큰 수가 1분마다 다시 설정되는 모든 요청의 실행 중인 토큰 수에 추가됩니다. 해당 분 동안 언제든지 TPM 속도 제한 값에 도달하면 카운터가 다시 설정될 때까지 추가 요청에 429 응답 코드가 수신됩니다.

RPM 속도 제한은 시간 경과에 따라 수신된 요청 수를 기준으로 합니다. 속도 제한은 요청이 1분 동안 균등하게 분산될 것으로 예상합니다. 이 평균 흐름이 유지되지 않으면 1분 동안 측정했을 때 제한이 충족되지 않더라도 요청이 429 응답을 받을 수 있습니다. 이 동작을 구현하기 위해 Azure OpenAI 서비스는 짧은 시간(일반적으로 1초 또는 10초) 동안 수신 요청의 속도를 평가합니다. 해당 시간 동안 수신된 요청 수가 설정된 RPM 제한에서 예상되는 수를 초과하는 경우 새 요청은 다음 평가 기간까지 429 응답 코드를 받게 됩니다. 예를 들어, Azure OpenAI가 1초 간격으로 요청 속도를 모니터링하는 경우 1초마다 10개 이상의 요청이 수신되면(분당 600개 요청 = 초당 10개 요청) 600RPM 배포에 대해 속도 제한이 발생합니다.).

속도 제한 모범 사례

속도 제한과 관련된 문제를 최소화하려면 다음 기술을 사용하는 것이 좋습니다.

- max_tokens 및 best_of를 시나리오 요구 사항에 맞는 최솟값으로 설정합니다. 예를 들어, 응답이 작을 것으로 예상된다면 max-tokens 값을 크게 설정하지 마세요.
- 할당량 관리를 사용하여 트래픽이 많은 배포에서 TPM을 늘리고 요구 사항이 제한된 배포에서 TPM을 줄입니다.
- 애플리케이션에서 다시 시도 논리를 구현합니다.
- 워크로드가 급격히 변경되지 않도록 합니다. 워크로드를 점진적으로 늘립니다.
- 다양한 로드 증가 패턴을 테스트합니다.

배포 자동화

이 섹션에는 할당량을 사용하여 TPM 속도 제한을 설정하는 배포 만들기를 프로그래밍 방식으로 시작하는 데 도움이 되는 간략한 예 템플릿이 포함되어 있습니다. 할당량이 도입되면 리소스 관리 관련 작업에 API 버전 2023-05-01을 사용해야 합니다. 이 API 버전은 리소스 관리용으로만 사용되며 완료, 채팅 완료, 포함, 이미지 생성 등과 같은 호출 유추에 사용되는 API 버전에는 영향을 미치지 않습니다.

REST

배포

HTTP

PUT

```
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.CognitiveServices/accounts/{accountName}/deployments/{deploymentName}?api-version=2023-05-01
```

경로 매개 변수

[+] 테이블 확장

매개 변수	형식	필수 여부	설명
accountName	string	Required	Azure OpenAI 리소스의 이름입니다.
deploymentName	string	Required	기존 모델을 배포할 때 선택한 배포 이름 또는 새 모델 배포에 사용하려는 이름입니다.

매개 변수	형식	필수 여부	설명
resourceGroupName	string	Required	이 모델 배포에 연결된 리소스 그룹의 이름입니다.
subscriptionId	string	Required	연결된 구독의 구독 ID입니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-05-01 Swagger 사양 ↗

요청 본문

이는 사용할 수 있는 요청 본문 매개 변수의 하위 집합일 뿐입니다. 매개 변수의 전체 목록을 보려면 [REST API 참조 설명서](#)를 참조하세요.

[+] 테이블 확장

매개 변수	형식	설명
sku	SKU	SKU를 나타내는 리소스 모델 정의입니다.
capacity	정수	이는 이 배포에 할당하는 할당량 의 양을 나타냅니다. 값 1은 분당 토큰 (TPM) 1,000개와 같습니다. 값 10은 분당 토큰(TPM) 10,000개와 같습니다.

예제 요청

Bash

```
curl -X PUT https://management.azure.com/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/resource-group-temp/providers/Microsoft.CognitiveServices/accounts/docs-openai-test-001/deployments/gpt-35-turbo-test-deployment?api-version=2023-05-01 \
-H "Content-Type: application/json" \
-H 'Authorization: Bearer YOUR_AUTH_TOKEN' \
-d '{"sku": {"name": "Standard", "capacity": 10}, "properties": {"model": {"format": "OpenAI", "name": "gpt-35-turbo", "version": "0613"}}}'
```

① 참고

권한 부여 토큰을 생성하는 방법에는 여러 가지가 있습니다. 초기 테스트를 위한 가장 쉬운 방법은 [Azure Portal](#)에서 Cloud Shell을 시작하는 것입니다. 그

런 다음 az account get-access-token를 실행합니다. 이 토큰을 API 테스트를 위한 임시 권한 부여 토큰으로 사용할 수 있습니다.

자세한 내용은 [사용법](#) 및 [배포](#)에 대한 REST API 참조 설명서를 참조하세요.

사용

특정 지역에서 특정 구독에 대한 할당량 사용량을 쿼리하려면

HTML

GET
https://management.azure.com/subscriptions/{subscriptionId}/providers/Microsoft.CognitiveServices/locations/{location}/usages?api-version=2023-05-01

경로 매개 변수

[+] 테이블 확장

매개 변수	형식	필수 여부	설명
subscriptionId	string	Required	연결된 구독의 구독 ID입니다.
location	string	Required	사용량을 볼 수 있는 위치(예: eastus)
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-05-01 Swagger 사양 ↗

예제 요청

Bash

```
curl -X GET https://management.azure.com/subscriptions/00000000-0000-0000-0000-000000000000/providers/Microsoft.CognitiveServices/locations/eastus/usages?api-version=2023-05-01 \
-H "Content-Type: application/json" \
-H 'Authorization: Bearer YOUR_AUTH_TOKEN'
```

리소스 삭제

배포가 여전히 존재하는 경우 Azure Portal에서 Azure OpenAI 리소스를 삭제하려고 시도하면 연결된 배포가 삭제될 때까지 삭제가 차단됩니다. 배포를 먼저 삭제하면 할당량 할당이 적절하게 해제되어 새 배포에서 사용할 수 있습니다.

그러나 REST API 또는 기타 프로그래밍 방식을 사용하여 리소스를 삭제하면 먼저 배포를 삭제할 필요가 없습니다. 이런 일이 발생하면 리소스가 제거될 때까지 48시간 동안 관련 할당량 할당을 새 배포에 할당할 수 없는 상태로 유지됩니다. 제거된 리소스를 즉시 제거하여 할당량을 확보하려면 [제거된 리소스 제거 지침](#)을 따릅니다.

다음 단계

- Azure OpenAI의 할당량 기본값을 검토하려면 [할당량 및 제한 문서](#)를 참조 [하세요](#).

Azure OpenAI 동적 할당량(미리 보기)

아티클 • 2024. 02. 22.

동적 할당량은 추가 용량을 사용할 수 있을 때 표준(종량제) 배포를 통해 더 많은 할당량을 기회적으로 활용할 수 있도록 하는 Azure OpenAI 기능입니다. 동적 할당량이 꺼짐으로 설정되면 배포는 TPM(분당 토큰 수) 설정에 따라 설정된 최대 처리량을 처리할 수 있습니다. 사전 설정 TPM을 초과하면 요청이 HTTP 429 응답을 반환합니다. 동적 할당량이 사용하도록 설정되면 배포에서는 429 응답을 반환하기 전에 더 높은 처리량에 액세스할 수 있으므로 더 많은 호출을 더 일찍 수행할 수 있습니다. 추가 요청은 여전히 [일반 가격 책정 요율](#)로 청구됩니다.

동적 할당량은 사용 가능한 할당량을 일시적으로만 늘릴 수 있습니다. 구성된 값 이하로 줄어들지 않습니다.

동적 할당량을 사용하는 경우

동적 할당량은 대부분의 시나리오에서 유용하며, 특히 애플리케이션이 기회에 따라 추가 용량을 사용할 수 있거나 애플리케이션 자체가 Azure OpenAI API 호출 속도를 높이는 경우에 유용합니다.

일반적으로 동적 할당량을 피하고 싶은 상황은 할당량이 변동되거나 증가하는 경우 애플리케이션에서 불리한 환경을 제공하는 경우입니다.

동적 할당량의 경우 다음과 같은 시나리오를 고려합니다.

- 대량 처리,
- RAG(검색 증강 생성)를 위한 요약 또는 포함 만들기,
- 메트릭 및 평가 생성을 위한 오프라인 로그 분석,
- 우선 순위가 낮은 연구,
- 소량의 할당량이 할당된 앱입니다.

동적 할당량은 언제 적용되나요?

Azure OpenAI 백 엔드는 다양한 배포에서 추가 동적 할당량을 추가하거나 제거할지 여부, 시기 및 양을 결정합니다. 사전에 예측하거나 공지하지도 않으며, 예측할 수도 없습니다. Azure OpenAI는 HTTP 429로 응답하고 추가 API 호출을 허용하지 않음으로써 애플리케이션에 더 많은 할당량이 있음을 알립니다. 동적 할당량을 활용하려면 HTTP 429 응답이 드물기 때문에 애플리케이션 코드에서 더 많은 요청을 실행할 수 있어야 합니다.

동적 할당량은 비용을 어떻게 변경하나요?

- 기본 할당량을 초과하여 수행된 호출은 일반 호출과 동일한 비용이 발생합니다.
- 배포에서 동적 할당량을 설정하는 데 추가 비용은 없지만, 처리량 증가로 인해 배포에서 수신하는 트래픽 양에 따라 궁극적으로 비용이 증가할 수 있습니다.

① 참고

동적 할당량을 사용하면 "최대" 할당량 또는 처리량에 대한 호출 적용이 없습니다. Azure OpenAI는 기준 할당량을 초과하여 최대한 많은 요청을 처리합니다. 할당량이 덜 제한되어 있는 경우에도 지출 비율을 제어해야 하는 경우 애플리케이션 코드는 이에 따라 요청을 보류해야 합니다.

동적 할당량을 사용하는 방법

동적 할당량을 사용하려면 다음을 수행해야 합니다.

- Azure OpenAI 배포에서 동적 할당량 속성을 설정합니다.
- 애플리케이션이 동적 할당량을 활용할 수 있는지 확인합니다.

동적 할당량 사용

배포에 대한 동적 할당량을 활성화하려면 리소스 구성의 고급 속성으로 이동하여 이를 켜면 됩니다.

Deploy model

X

Set up a deployment to make API calls against a provided base model or a custom model. Finished deployments are available for use. Your deployment status will move to succeeded when the deployment is complete and ready for use.

Select a model ⓘ

gpt-35-turbo-16k

▼

Model version ⓘ

Auto-update to default

▼

*

Deployment name ⓘ

*

 Advanced options ▼

Content Filter ⓘ

Default

▼

(ⓘ) 62K tokens per minute quota available for your deployment

Tokens per Minute Rate Limit (thousands) ⓘ

62K

Corresponding requests per minute (RPM) = 6

Enable Dynamic Quota ⓘ

Enabled

Create

Cancel



또는 Azure CLI의 [az rest](#)를 사용하여 프로그래밍 방식으로 사용하도록 설정할 수 있습니다.

{subscriptionId}, {resourceGroupName}, {accountName} 및 {deploymentName} 을 리소스에 대한 관련 값으로 바꿉니다. 이 경우 accountName 은 Azure OpenAI 리소스 이름과 같습니다.

Azure CLI

```
az rest --method patch --url  
"https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/  
{resourceGroupName}/providers/Microsoft.CognitiveServices/accounts/{accountN  
ame}/deployments/{deploymentName}?2023-10-01-preview" --body '{"properties":  
{"dynamicThrottlingEnabled": true} }'
```

내 앱에 추가되는 처리량 동적 할당량을 어떻게 알 수 있나 요?

작동 방식을 모니터링하려면 Azure Monitor에서 애플리케이션의 처리량을 추적할 수 있습니다. 동적 할당량 미리 보기 중에는 할당량이 동적으로 증가 또는 감소했는지 여부를 나타내는 특정 메트릭이나 로그가 없습니다. 동적 할당량은 활용도가 높은 지역에서 실행되고 해당 지역의 사용량이 가장 많은 시간 동안 배포에 사용될 가능성이 적습니다.

다음 단계

- [할당량 작동](#) 방법에 대해 자세히 알아봅니다.
- [Azure OpenAI 모니터링](#)에 대해 자세히 알아봅니다.

Azure OpenAI Service 모니터링

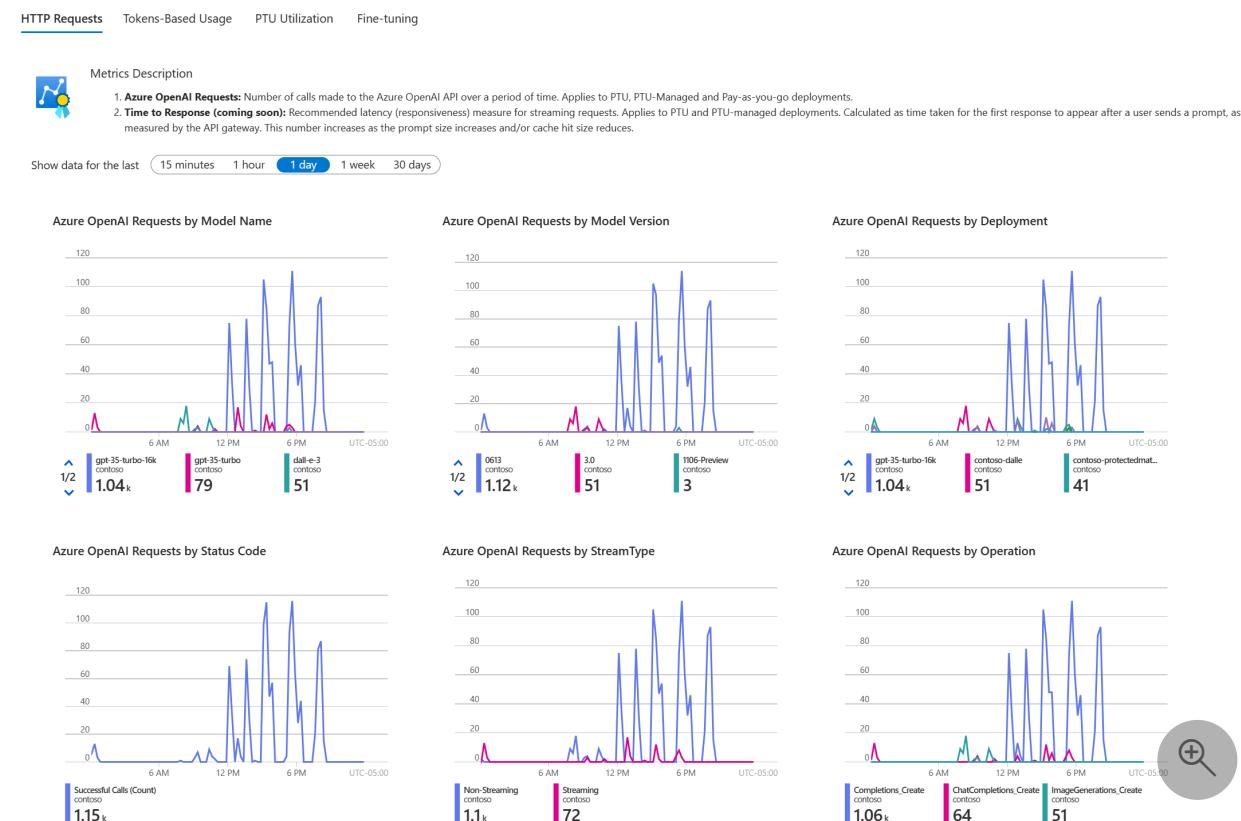
아티클 • 2024. 03. 29.

Azure 리소스를 사용하는 중요한 애플리케이션 및 비즈니스 프로세스가 있으면 이러한 리소스의 가용성, 성능 및 작업을 모니터링해야 합니다.

이 문서에서는 Azure OpenAI Service에서 생성된 데이터 모니터링에 대해 설명합니다. Azure OpenAI는 Azure AI 서비스의 일부이며, [Azure Monitor](#)를 사용합니다. Azure Monitor를 사용하는 모든 Azure 서비스에 일반적인 Azure Monitor 기능에 익숙하지 않은 경우 [Azure Monitor](#)를 사용하여 Azure 리소스 모니터링을 참조하세요.

대시보드

Azure OpenAI는 각 Azure OpenAI 리소스에 대한 기본 대시보드를 제공합니다. <https://portal.azure.com>에 대한 모니터링 대시보드에 액세스하고 Azure OpenAI 리소스 중 하나에 대한 개요 창을 선택합니다.



대시보드는 네 개의 범주인 HTTP 요청, 토큰 기반 사용량, PTU 사용률 및 미세 조정으로 그룹화됩니다.

Azure Monitor의 데이터 수집 및 라우팅

Azure OpenAI는 다른 Azure 리소스와 동일한 종류의 모니터링 데이터를 수집합니다. 활동 로그, 리소스 로그, 가상 머신 로그 및 플랫폼 메트릭에서 데이터를 생성하도록 Azure Monitor를 구성할 수 있습니다. 자세한 내용은 [Azure 리소스의 데이터 모니터링](#)을 참조하세요.

플랫폼 메트릭과 Azure Monitor 활동 로그는 자동으로 수집 및 저장됩니다. 이 데이터는 진단 설정을 사용하여 다른 위치로 라우팅될 수 있습니다. Azure Monitor 리소스 로그는 진단 설정을 만들고 하나 이상의 위치로 해당 로그를 라우팅할 때까지 수집 및 저장되지 않습니다.

진단 설정을 만들 때 수집할 로그 범주를 지정합니다. Azure Portal, Azure CLI 또는 PowerShell을 사용하여 진단 설정 만들기의 자세한 내용은 [Azure에서 플랫폼 로그 및 메트릭을 수집하는 진단 설정 만들기](#)를 참조하세요.

진단 설정을 사용하고 Azure Monitor 로그로 데이터를 보내는 데는 다른 비용이 발생한다는 점에 유의하세요. 자세한 내용은 [Azure Monitor 로그 비용 계산 및 옵션](#)을 참조하세요.

수집할 수 있는 메트릭 및 로그는 다음 섹션에서 설명합니다.

메트릭 분석

Azure Portal에서 Azure Monitor 도구를 사용하여 Azure OpenAI Service 리소스에 대한 메트릭을 분석할 수 있습니다. Azure OpenAI 리소스에 대한 [개요](#) 페이지에서 왼쪽 창에서 [모니터링](#) 아래의 [메트릭](#)을 선택합니다. 자세한 내용은 [Azure Monitor 메트릭 탐색기 시작](#)을 참조하세요.

Azure OpenAI에는 Azure AI 서비스의 하위 집합과 공통성이 있습니다. Azure Monitor에서 Azure OpenAI 및 비슷한 Azure AI 서비스에 대해 수집하는 모든 플랫폼 메트릭 목록은 [Microsoft.CognitiveServices/accounts에 대한 지원 메트릭](#)을 참조하세요.

Cognitive Services 메트릭

이러한 메트릭은 모든 Azure AI Services 리소스에 공통적인 레거시 메트릭입니다. 더 이상 Azure OpenAI에서 이러한 메트릭을 사용하지 않는 것이 좋습니다.

Azure OpenAI 메트릭

① 참고

프로비전된 관리되는 사용률 메트릭은 이제 더 이상 사용되지 않으며 더 이상 권장되지 않습니다. 이 메트릭은 [프로비전된 관리되는 사용률 V2](#) 메트릭으로 대체되었습니다.

습니다.

다음 표에는 Azure OpenAI에서 사용할 수 있는 메트릭의 현재 하위 집합이 요약되어 있습니다.

[+] 테이블 확장

메트릭	범주	집계	설명	차원
Azure OpenAI Requests	HTTP	Count	일정 기간 동안 Azure OpenAI API에 대해 수행된 총 호출 수입니다. PayGo, PTU 및 PTU 관리형 SKU에 적용됩니다.	ApiName, ModelDeploymentName, ModelName, ModelVersion, OperationName, Region, StatusCode, StreamType
Generated Completion Tokens	사용	Sum	Azure OpenAI 모델에서 생성된 토큰(출력) 수입니다. PayGo, PTU 및 PTU 관리형 SKU에 적용됩니다.	ApiName, ModelDeploymentName, ModelName, Region
Processed FineTuned Training Hours	사용	Sum	OpenAI FineTuned 모델에서 처리된 학습 시간 수	ApiName, ModelDeploymentName, ModelName, Region
Processed Inference Tokens	사용	Sum	OpenAI 모델에서 처리된 유추 토큰 수입니다. 프롬프트 토큰 (입력) + 생성된 토큰으로 계산됩니다. PayGo, PTU 및 PTU 관리형 SKU에 적용됩니다.	ApiName, ModelDeploymentName, ModelName, Region
Processed Prompt Tokens	사용	Sum	OpenAI 모델에서 처리된 총 프롬프트 토큰(입력) 수입니다. PayGo, PTU 및 PTU 관리형	ApiName, ModelDeploymentName, ModelName, Region

메트릭	범주	집계	설명	차원
			SKU에 적용됩니다.	
Provision-managed Utilization V2	사용	평균	프로비전된 관리형 사용률은 지정된 프로비전된 관리형 배포의 사용률입니다. (사용된 PTU/배포된 PTU)*100으로 계산됩니다. 사용률이 100% 이상인 경우 호출이 제한되고 429 오류 코드가 반환됩니다.	ModelDeploymentName, ModelName, ModelVersion, Region, StreamType

진단 설정 구성

모든 메트릭은 Azure Monitor의 진단 설정을 사용하여 내보낼 수 있습니다. Azure Monitor Log Analytics 쿼리를 사용하여 로그 및 메트릭 데이터를 분석하려면 Azure OpenAI 리소스 및 Log Analytics 작업 영역에 대한 진단 설정을 구성해야 합니다.

1. Azure OpenAI 리소스 페이지의 왼쪽 창에 있는 **모니터링** 아래에서 **진단 설정**을 선택합니다. 진단 설정 페이지에서 **진단 설정 추가**를 선택합니다.

The screenshot shows the Azure portal interface for managing diagnostic settings. On the left, there's a sidebar with navigation links: Monitoring, Alerts, Metrics, **Diagnostic settings** (which is highlighted with a red box), Logs, Automation, Help, Resource health, and Support + Troubleshooting. The main content area is titled 'my-openai-resource | Diagnostic settings'. It displays a table of diagnostic settings:

Name	Storage account	Event hub	Log Analytics workspace
my-openai-resource	-	-	my-log-workspace

Below the table, there's a button labeled '+ Add diagnostic setting' with a red box around it. At the bottom, there's a note: 'Click 'Add Diagnostic setting' above to configure the collection of the following data:' followed by a list: Audit Logs, Request and Response Logs, Trace Logs, and AllMetrics.

2. 진단 설정 페이지에서 다음 필드를 구성합니다.

- a. Log Analytics 작업 영역에 보내기를 선택합니다.
- b. Azure 계정 구독을 선택합니다.
- c. Log Analytics 작업 영역을 선택합니다.
- d. 로그에서 allLogs를 선택합니다.
- e. 메트릭에서 AllMetrics를 선택합니다.

The screenshot shows the 'Diagnostic setting' configuration page. At the top, there are buttons for Save (highlighted with a red box), Discard, Delete, and Feedback. Below that, a descriptive text explains what a diagnostic setting does, mentioning log categories and contents, with a 'Learn more' link and a 'JSON View' button. The main area is divided into 'Logs' and 'Metrics' sections. Under 'Logs', 'Category groups' include 'Audit' (unchecked) and 'allLogs' (checked). Under 'Metrics', 'Metrics' includes 'AllMetrics' (checked). On the right, 'Destination details' are configured: 'Send to Log Analytics workspace' is checked (highlighted with a red box), and dropdown menus show 'Subscription' and 'Log Analytics workspace' both set to '< [] >' (highlighted with red boxes). There are also options for 'Archive to a storage account', 'Stream to an event hub', and 'Send to partner solution'.

3. 진단 설정 이름을 입력하여 구성의 저장합니다.

4. 저장을 선택합니다.

진단 설정을 구성한 후 Log Analytics 작업 영역에서 Azure OpenAI 리소스에 대한 메트릭 및 로그 데이터를 사용할 수 있습니다.

로그 분석

Azure Monitor Logs의 데이터는 테이블마다 고유한 자체 속성 집합이 있는 테이블에 저장됩니다.

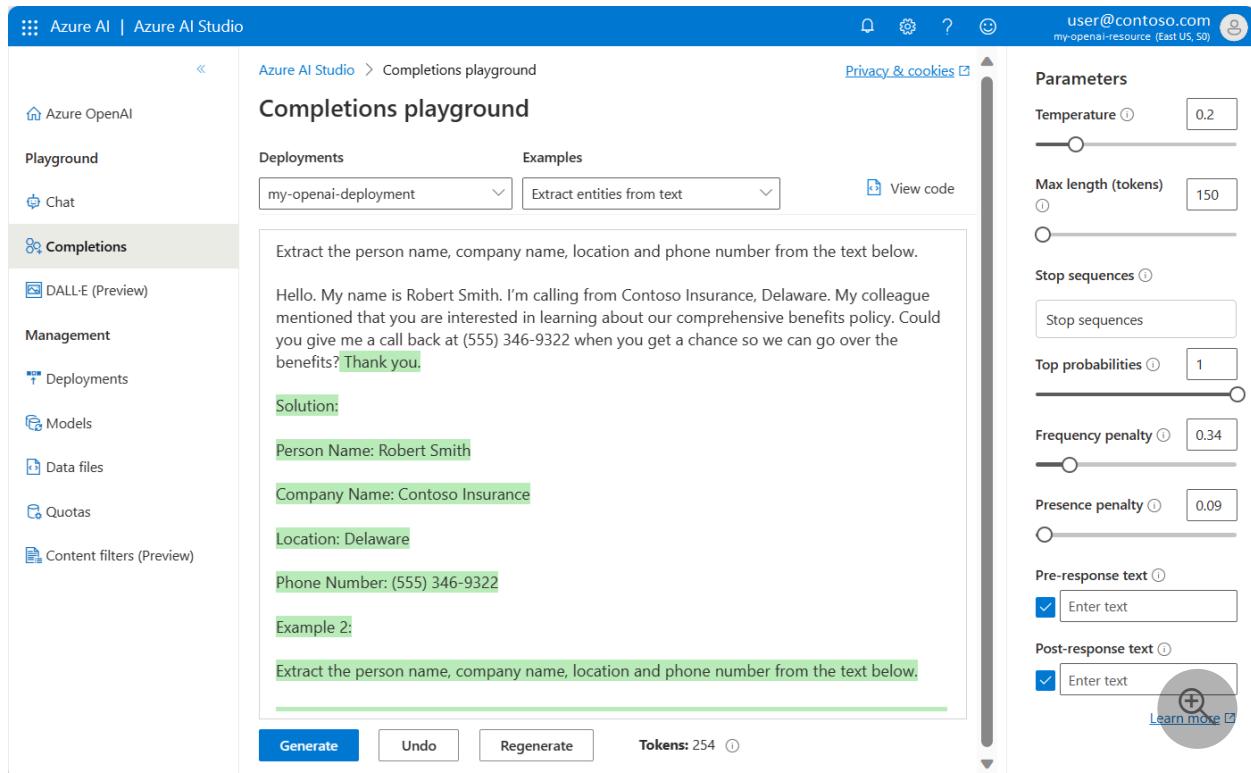
Azure Monitor의 모든 리소스 로그에는 동일한 필드와 그 뒤에 오는 서비스별 필드가 있습니다. 일반적인 스키마에 대한 내용은 [Azure 리소스 로그에 대한 공통 및 서비스별 스키마](#)를 참조하세요.

[활동 로그](#)는 구독 수준의 이벤트에 대한 인사이트를 제공하는 Azure의 플랫폼 로그 유형입니다. 이 로그를 독립적으로 보거나, Azure Monitor 로그로 라우팅할 수 있습니다. Azure Portal에서 Azure Monitor 로그의 활동 로그를 사용하여 Log Analytics로 복잡한 쿼리를 실행할 수 있습니다.

Azure OpenAI 및 비슷한 Azure AI 서비스에 사용할 수 있는 리소스 로그 유형의 목록은 [Microsoft.CognitiveServices](#) Azure 리소스 공급자 작업을 참조하세요.

Kusto 쿼리 사용

Azure OpenAI 모델을 배포한 후 [Azure AI Studio](#)의 [플레이그라운드](#) 환경을 사용하여 일부 완료 호출을 보낼 수 있습니다.



완료 플레이그라운드 또는 채팅 완료 플레이그라운드에 입력하는 모든 텍스트는 Azure OpenAI 리소스에 대한 메트릭 및 로그 데이터를 생성합니다. 리소스에 대한 Log Analytics 작업 영역에서 [Kusto](#) 쿼리 언어를 사용하여 모니터링 데이터를 쿼리할 수 있습니다.

ⓘ 중요

Azure OpenAI 리소스 페이지의 **쿼리 열기** 옵션을 선택하면 이 문서에 설명되지 않은 Azure Resource Graph로 이동합니다. 다음 쿼리는 Log Analytics에 대한 쿼리 환경을 사용합니다. [진단 설정 구성](#)의 단계에 따라 Log Analytics 작업 영역을 준비해야 합니다.

1. Azure OpenAI 리소스 페이지의 왼쪽 창에 있는 모니터링에서 로그를 선택합니다.

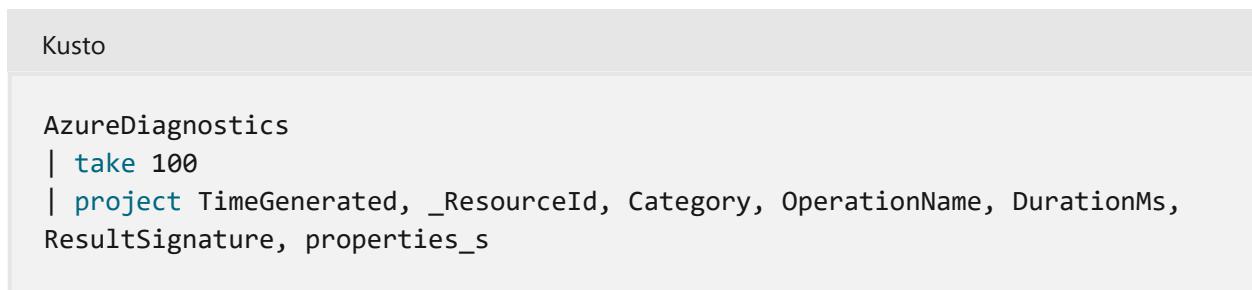
2. Azure OpenAI 리소스에 대해 진단을 구성한 Log Analytics 작업 영역을 선택합니다.

3. Log Analytics 작업 영역 페이지의 왼쪽 창에 있는 개요에서 로그를 선택합니다.

Azure Portal은 기본적으로 샘플 쿼리 및 제안이 포함된 쿼리 창을 표시합니다. 이 창을 닫을 수 있습니다.

다음 예제에서는 쿼리 창 맨 위에 있는 편집 영역에 Kusto 쿼리를 입력한 다음, 실행을 선택합니다. 쿼리 결과는 쿼리 텍스트 아래에 표시됩니다.

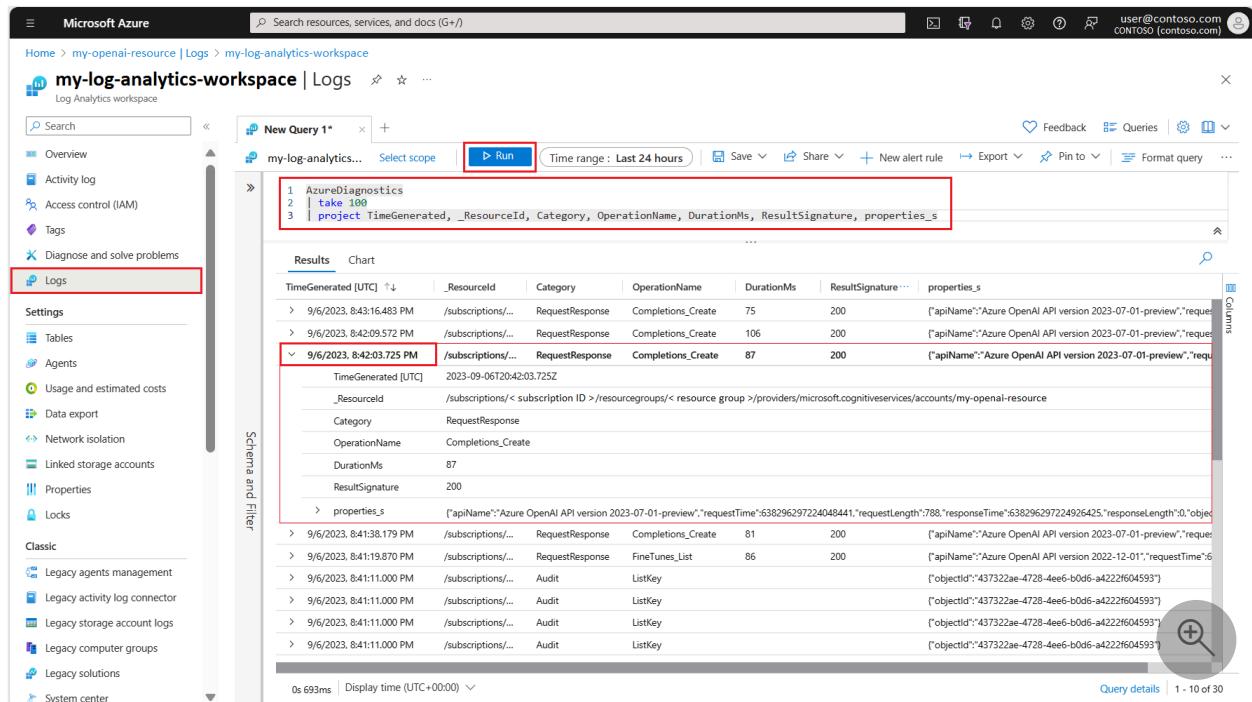
다음 Kusto 쿼리는 리소스에 대한 Azure Diagnostics(AzureDiagnostics) 데이터의 초기 분석에 유용합니다.



```
Kusto

AzureDiagnostics
| take 100
| project TimeGenerated, _ResourceId, Category, OperationName, DurationMs,
ResultSignature, properties_s
```

이 쿼리는 100개 항목의 샘플을 반환하고 로그에 사용 가능한 데이터 열의 하위 집합을 표시합니다. 쿼리 결과에서 테이블 이름 옆에 있는 화살표를 선택하여 사용 가능한 모든 열 및 관련 데이터 형식을 볼 수 있습니다.



The screenshot shows the Microsoft Azure Log Analytics workspace interface. On the left, there's a sidebar with navigation links like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, and a prominent Logs link which is highlighted with a red box. The main area displays a Kusto query editor with the following query:

```
1 AzureDiagnostics
2 | take 100
3 | project TimeGenerated, _ResourceId, Category, OperationName, DurationMs, ResultSignature, properties_s
```

Below the query editor, the results pane shows a table with several rows of data. The columns are: TimeGenerated [UTC], _ResourceId, Category, OperationName, DurationMs, ResultSignature, and properties_s. The first row has expand arrows next to TimeGenerated, _ResourceId, Category, OperationName, DurationMs, and ResultSignature. The properties_s column contains a large JSON object. A red box highlights the 'Run' button in the query editor toolbar.

사용 가능한 모든 데이터 열을 보려면 쿼리에서 범위 지정 매개 변수 줄 | project ... 를 제거할 수 있습니다.

Kusto

```
AzureDiagnostics  
| take 100
```

리소스에 대한 Azure Metrics(`AzureMetrics`) 데이터를 검사하려면 다음 쿼리를 실행합니다.

Kusto

```
AzureMetrics  
| take 100  
| project TimeGenerated, MetricName, Total, Count, Maximum, Minimum,  
Average, TimeGrain, UnitName
```

이 쿼리는 100개 항목의 샘플을 반환하고 Azure Metrics 데이터의 사용 가능한 열 하위 집합을 표시합니다.

Results Chart Columns

TimeGenerated [UTC] ↑↓	MetricName	Total	Count	Maximum	Minimum	Average	TimeGrain	UnitName
> 9/6/2023, 8:46:00.000 PM	Latency	105	1	105	105	105	PT1M	Milliseconds
> 9/6/2023, 8:46:00.000 PM	TotalCalls	1	1	1	1	1	PT1M	Count
▽ 9/6/2023, 8:46:00.000 ...	GeneratedTokens	150	1			150	PT1M	n/a
TimeGenerated [UTC] 2023-09-06T20:46:00Z								
MetricName GeneratedTokens								
Total 150								
Count 1								
Average 150								
TimeGrain PT1M								
UnitName n/a								
> 9/6/2023, 8:46:00.000 PM	DataIn	497	1	497	497	497	PT1M	Bytes
> 9/6/2023, 8:45:00.000 PM	ProcessedPromptTokens	310	4			77.5	PT1M	n/a
> 9/6/2023, 8:45:00.000 PM	GeneratedTokens	257	4			64.25	PT1M	n/a
> 9/6/2023, 8:45:00.000 PM	DataIn	1925	4	829	321	481.25	PT1M	Bytes
> 9/6/2023, 8:45:00.000 PM	TokenTransaction	567	4			141.75	PT1M	n/a

4s 53ms | Display time (UTC+00:00) ▾ Query details | 6 - 14 of 45

① 참고

리소스에 대한 Azure OpenAI 메뉴에서 **모니터링>로그**를 선택하면 쿼리 범위가 현재 리소스로 설정된 Log Analytics가 열립니다. 표시되는 로그 쿼리에는 해당 특정 리소스의 데이터만 포함됩니다. 다른 리소스의 데이터 또는 다른 Azure 리소스의 데이터를 포함하는 쿼리를 실행하려면 Azure Portal의 **Azure Monitor** 메뉴에서 **로그**를 선택합니다. 자세한 내용은 [Azure Monitor Log Analytics의 로그 쿼리 범위 및 시간 범위](#)를 참조하세요.

경고 설정

Azure Monitor 경고는 모니터링 데이터에서 중요한 조건이 발견될 때 사용자에게 사전에 알립니다. 이를 통해 사용자에게 알리기 전에 시스템 문제를 식별하고 해결할 수 있습니다. [메트릭](#), [로그](#) 및 [활동 로그](#)에서 경고를 설정할 수 있습니다. 서로 다른 형식의 경고에는 장점과 단점이 있습니다.

모든 조직의 경고 요구 사항은 다르며 시간이 지남에 따라 변경될 수 있습니다. 일반적으로 모든 경고는 실행 가능해야 하며 경고가 발생할 때 의도된 구체적인 응답이 있어야 합니다. 경고에 즉각적인 응답이 필요하지 않은 경우 경고가 아닌 보고서에서 조건을 캡처할 수 있습니다. 일부 사용 사례에서는 특정 오류 조건이 있을 때마다 경고가 필요할 수 있습니다. 다른 경우에는 지정된 기간 동안 특정 임계값을 초과하는 오류에 대한 경고가 필요할 수 있습니다.

특정 임계값 미만의 오류는 Azure Monitor 로그의 데이터를 정기적으로 분석하여 평가할 수 있는 경우가 많습니다. 시간이 지남에 따라 로그 데이터를 분석할 때 특정 조건이 예상 기간 동안 발생하지 않는 것을 발견할 수 있습니다. 경고를 사용하여 이 조건을 추적할 수 있습니다. 경우에 따라 로그에 이벤트가 없는 것이 오류만큼 중요한 신호일 수 있습니다.

Azure OpenAI를 사용하여 개발 중인 애플리케이션 유형에 따라 [Azure Monitor Application Insights](#)는 애플리케이션 계층에서 추가 모니터링 이점을 제공할 수 있습니다.

다음 단계

- [Azure Monitor를 사용하여 Azure 리소스 모니터링](#)
- [Azure Monitor 로그의 로그 검색 이해](#)

프로비전된 처리량 단위 온보딩

아티클 • 2024. 02. 21.

이 문서에서는 PTU(프로비전된 처리량 단위)에 온보딩하는 프로세스를 안내합니다. 초기 온보딩을 완료하면 PTU 시작 가이드를 참조하는 것이 좋습니다.

① 참고

프로비전된 처리량 단위(PTU)는 Azure OpenAI의 표준 할당량과 다르며 기본적으로 사용할 수 없습니다. 이 제품에 대한 자세한 내용은 Microsoft 계정 팀에 문의하세요.

크기 조정 및 예측: 프로비전된 관리 전용

워크로드에 필요한 프로비전된 처리량 또는 CPU의 적절한 양을 결정하는 것은 성능 및 비용을 최적화하는 데 필수적인 단계입니다. 이 섹션에서는 Azure OpenAI 용량 계획 도구를 사용하는 방법을 설명합니다. 이 도구는 워크로드의 요구 사항을 충족하는 데 필요한 PTU의 예상을 제공합니다.

프로비전된 처리량 및 비용 예측

워크로드에 대한 빠른 예상을 얻으려면 Azure OpenAI Studio에서 Capacity Planner를 엽니다. Capacity Planner는 프로비전된 관리>할당량 아래에 있습니다.>

프로비전된 옵션 및 용량 플래너는 할당량 창 내의 특정 지역에서만 사용할 수 있습니다. 이 옵션이 표시되지 않으면 할당량 지역을 스웨덴 중부로 설정하면 이 옵션을 사용할 수 있습니다. 워크로드에 따라 다음 매개 변수를 입력합니다.

☰ 테이블 확장

입력	설명
모델	사용하려는 OpenAI 모델입니다. 예: GPT-4
버전	사용하려는 모델의 버전(예: 0614)
프롬프트 토큰	각 호출에 대한 프롬프트의 토큰 수
생성 토큰	각 호출에서 모델에서 생성된 토큰 수
분당 최고 호출 수	분당 호출로 측정된 엔드포인트에 대한 최대 동시 로드

필요한 세부 정보를 입력한 후 계산을 선택하여 시나리오에 대해 제안된 PTU를 확인합니다.

Capacity calculator

Select a model * ⓘ

Model version * ⓘ

gpt-4

0613

Workload size

Prompt tokens * ⓘ

200

Generation tokens * ⓘ

400

Peak calls per min * ⓘ

1000

Estimate

Suggested value

PTU estimate ⓘ

5500

Close

The screenshot shows the Azure OpenAI Capacity Planner interface. It has sections for selecting a model (gpt-4, Model version 0613), specifying workload size (Prompt tokens: 200, Generation tokens: 400, Peak calls per min: 1000), and calculating the estimated Price-to-Usage (PTU) estimate (5500). A 'Suggested value' section is also present. At the bottom right is a 'Close' button and a circular icon with a plus sign.

① 참고

Capacity Planner는 간단한 입력 기준에 따른 추정치입니다. 용량을 결정하는 가장 정확한 방법은 사용 사례에 대한 표현 워크로드를 사용하여 배포를 벤치마킹하는 것입니다.

프로비전된 처리량 구매 모델 이해

사용량에 따라 요금이 청구되는 Azure 서비스와 달리 Azure OpenAI 프로비전된 처리량 기능은 재생 가능한 월별 약정으로 구매됩니다. 이 약정은 생성 시 및 매월 갱신할 때마다 구독에 청구됩니다. 프로비전된 처리량에 온보딩하는 경우 프로비전된 배포를 만들려는

각 Azure OpenAI 리소스에 대한 약정을 만들어야 합니다. 이러한 방식으로 구매한 CPU는 해당 리소스에 배포를 만들 때 사용할 수 있습니다.

약정을 통해 구매할 수 있는 총 CPU 수는 구독에 할당된 프로비전된 처리량 할당량의 양으로 제한됩니다. 다음 표에서는 프로비전된 처리량 할당량(PTU) 및 프로비전된 처리량 약정의 다른 특성을 비교합니다.

테이블 확장

항목	할당량	약정
목적	프로비전된 배포를 만들 수 있는 권한을 부여하고 사용할 수 있는 용량의 상한을 제공합니다.	프로비전된 처리량 용량을 위한 차량 구매
수명(lifetime)	할당이 부여된 후 5일 이내에 약정을 통해 구매하지 않으면 구독에서 할당량이 제거될 수 있습니다.	최소 기간은 1개월이며 고객이 선택할 수 있는 autorenewal 동작이 있습니다. 약정은 취소할 수 없으며 활성 상태인 동안 새 리소스로 이동할 수 없습니다.
범위	할당량은 구독 및 지역과 관련이 있으며 모든 Azure OpenAI 리소스에서 공유됩니다.	약정은 Azure OpenAI 리소스의 특성이며 해당 리소스 내의 배포로 범위가 지정됩니다. 구독에는 리소스가 있는 만큼의 활성 약정이 포함될 수 있습니다.
세분성	할당량은 모델 패밀리(예: GPT-4)에 한정되지만 제품군 내의 모델 버전 간에 공유할 수 있습니다.	약정은 모델 또는 버전별로 다릅니다. 예를 들어 리소스의 1000 PTU 약정은 GPT-4 및 GPT-35-Turbo의 배포를 포함할 수 있습니다.
용량 보장	할당량이 있다고 해서 배포를 만들 때 용량을 사용할 수 있다고 보장하지는 않습니다.	커밋된 CPU를 커버하는 용량 가용성은 약정이 활성 상태인 한 보장됩니다.
증가/감소	약정 갱신 날짜와 관계없이 언제든지 새 할당량을 요청하고 승인할 수 있습니다.	약정에서 적용되는 CPU 수는 언제든지 늘릴 수 있지만 갱신 시를 제외하고는 줄일 수 없습니다.

할당량과 약정은 함께 작동하여 구독 내에서 배포 만들기를 제어합니다. 프로비전된 배포를 만들려면 다음 두 가지 조건을 충족해야 합니다.

- 원하는 지역 및 구독 내에서 원하는 모델에 할당량을 사용할 수 있어야 합니다. 즉, 모델에 대한 구독/지역 전체 제한을 초과할 수 없습니다.
- 배포를 만드는 리소스에서 커밋된 PTU를 사용할 수 있어야 합니다. (배포에 할당하는 용량은 유료입니다).

약정 속성 및 층전 모델

약정에는 여러 속성이 포함됩니다.

☰ 테이블 확장

속성	설명	설정 시
Azure OpenAI 리소스	약정을 호스팅하는 리소스	약정 만들기
커밋된PTU	약정에서 적용되는PTU 수입니다.	처음에는 약정을 만들 때 설정되며 언제든지 늘릴 수 있지만 감소할 수는 없습니다.
용어	약정 기간입니다. 약정은 생성 날짜로부터 1개월 후에 만료됩니다. 갱신 정책은 다음에 수행되는 일을 정의합니다.	약정 만들기
만료 날짜	약정 만료 날짜입니다. 이 만료 시간은 자정 UTC입니다.	처음에는 생성 30일 후입니다. 그러나 약정이 갱신되면 만료 날짜가 변경됩니다.
갱신 정책	만료 시 수행할 작업에 대한 세 가지 옵션이 있습니다. - Autorenew: 새로운 약정 기간이 현재 CPU 수로 30일 더 시작됩니다. - 다른 설정이 있는 Autorenew: 이 설정은 갱신 시 커밋된 CPU 수를 줄일 수 있다는 점을 제외하고 Autorenew와 동일합니다. - 자동 갱신 안 함: 만료 시 약정이 종료되고 갱신되지 않습니다.	처음에는 약정을 만들 때 설정되며 언제든지 변경할 수 있습니다.

약정 요금

프로비전된 처리량 약정은 다음 시간에 Azure 구독에 대한 요금을 생성합니다.

- 약정을 만들 때. 요금은 현재 월별 PTU 속도 및 커밋된 CPU 수에 따라 계산됩니다. 청구서에 선불로 한 번의 요금이 청구됩니다.
- 약정 갱신 시. 갱신 정책이 자동 갱신으로 설정된 경우 새 기간에 커밋된 CPU에 따라 새 월별 요금이 생성됩니다. 이 요금은 청구서에 단일 선불 요금으로 표시됩니다.
- 새PTU가 기존 약정에 추가되는 경우 요금은 기존 약정 기간이 끝날 때까지 매시간 비례 배분된 약정에 추가된PTU 수를 기준으로 계산됩니다. 예를 들어, 300PTU가 해당 기간의 중간에 정확히 900PTU의 기존 약정에 추가되는 경우 추가 시 150PTU(약

정 만료 날짜에 비례하여 300PTU)에 대한 요금이 부과됩니다. 약정이 갱신되면 다음 달의 요금은 새 PTU 총 1,200PTU에 대한 요금이 됩니다.

리소스에 배포된 CPU 수가 리소스의 약정에 포함되는 한 약정 요금만 표시됩니다. 그러나 리소스에 배포된 CPU 수가 리소스의 커밋된 PTU보다 크면 초과 CPU는 시간당 초과분으로 청구됩니다. 일반적으로 이 초과분이 발생하는 유일한 방법은 리소스에 배포가 포함된 동안 약정이 만료되거나 갱신 시 감소되는 경우입니다. 예를 들어 300개의 PTU가 배포된 리소스에서 300 PTU 약정이 만료되도록 허용된 경우 배포된 PTU는 더 이상 약정에서 적용되지 않습니다. 만료 날짜에 도달하면 300개 초과 CPU에 따라 구독에 시간당 초과분 요금이 청구됩니다.

시간당 요금은 월별 약정 요금보다 높으며 요금은 며칠 이내에 월별 요금을 초과합니다. 시간당 초과분 요금을 종료하는 방법에는 두 가지가 있습니다.

- 커밋된 것보다 더 많은 PTU를 사용하지 않도록 배포를 삭제하거나 축소합니다.
- 배포된 CPU를 충당하기 위해 리소스에 대한 새 약정을 만듭니다.

약정 구매 및 관리

약정 계획

PTU(프로비전된 처리량 단위) 할당량이 구독에 할당되었다는 확인 메시지가 표시되면 대상 리소스에 대한 약정을 만들거나 기존 약정을 확장하여 배포에 할당량을 사용할 수 있도록 해야 합니다.

약정을 만들기 전에 프로비전된 배포를 사용하는 방법과 이를 호스트할 Azure OpenAI 리소스를 계획합니다. 약정 기간은 최소 1 개월이며 기간이 끝날 때까지 크기를 줄일 수 없습니다. 또한 만든 후에는 새 리소스로 이동할 수 없습니다. 마지막으로 커밋된 CPU의 합계는 할당량보다 클 수 없습니다. 리소스에서 커밋된 CPU는 더 이상 약정이 만료될 때까지 다른 리소스에 커밋할 수 없습니다. 프로비전된 배포에 사용할 리소스와 해당 리소스에 적용하려는 용량(최소 한 달)을 명확하게 계획하면 프로비전된 처리량 설정에 대한 최적의 환경을 보장하는 데 도움이 됩니다.

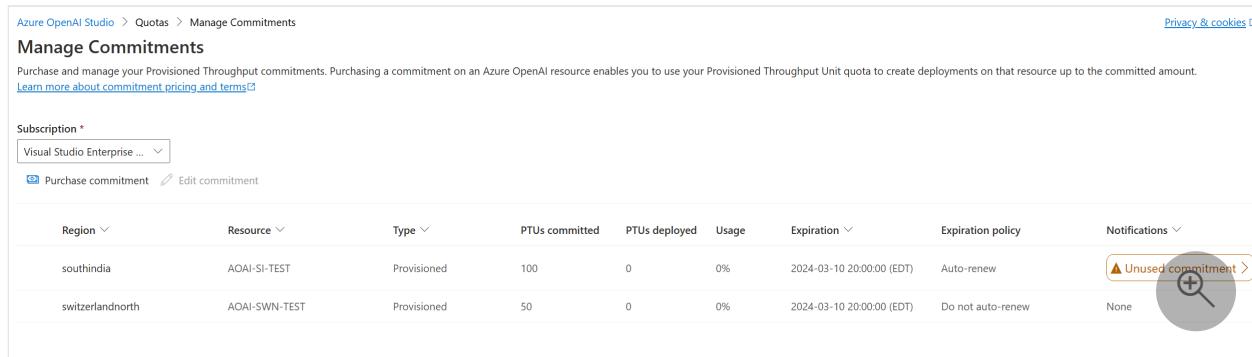
예시:

- 유효성 검사를 위해 임시 리소스에 대한 약정 및 배포를 만들지 마세요. 최소 한 달 동안 해당 리소스를 사용할 수 없습니다. 대신 최종적으로 프로덕션 리소스에서 PTU를 사용하는 계획인 경우 처음부터 해당 리소스에 대한 약정 및 테스트 배포를 만듭니다.
- 각 모델에서 배포를 만드는 데 필요한 최소 PTU 수를 염두에 두고 만들려는 배포의 수, 모델 및 크기에 따라 리소스에 커밋할 CPU 수를 계산합니다.

- 예제 1: GPT-4-32K를 배포하려면 최소 200PTU가 필요합니다. 리소스에 100PTU 만 약정하는 경우 GPT-4-32K를 배포하기에 커밋된 CPU가 충분하지 않습니다.
- 예제 2: 리소스에 여러 배포를 만들어야 하는 경우 각 배포에 필요한 PTU를 합산 합니다. GPT-4의 300PTU 및 GPT-4-32K의 500PTU에 대한 배포를 호스팅하는 프로덕션 리소스에는 두 배포를 모두 포함하려면 최소 800PTU의 약정이 필요합니다.
- 필요에 따라 CPU를 배포하거나 통합합니다. 예를 들어 배포를 지원하기 위해 필요에 따라 총 1000PTU 할당량을 리소스에 분산할 수 있습니다. 커밋된 CPU의 총 수가 할당량 1000보다 작거나 같으면 단일 리소스에 커밋하여 최대 1,000PTU를 추가하는 하나 이상의 배포를 지원하거나 여러 리소스(예: 개발 및 prod 리소스)에 분산될 수 있습니다.
- 계획에서 운영 요구 사항을 고려합니다. 예시:
 - 조직에서 필요한 리소스 명명 규칙
 - 여러 Azure OpenAI 리소스에서 지역당 모델을 여러 개 배포해야 하는 비즈니스 연속성 정책

프로비전된 처리량 약정 관리

프로비전된 처리량 약정은 Azure OpenAI Studio의 **약정 관리 보기**에서 만들어지고 관리됩니다. 할당량 창에서 약정 관리를 선택하여 이 보기로 이동할 수 있습니다.



The screenshot shows the 'Manage Commitments' page in Azure OpenAI Studio. It includes a header with 'Azure OpenAI Studio > Quotas > Manage Commitments' and a 'Privacy & cookies' link. Below the header is a 'Manage Commitments' section with a note about purchasing and managing provisioned throughput commitments. A 'Subscription' dropdown is set to 'Visual Studio Enterprise ...'. There are buttons for 'Purchase commitment' and 'Edit commitment'. The main table lists two commitments:

Region	Resource	Type	PTUs committed	PTUs deployed	Usage	Expiration	Expiration policy	Notifications
southindia	AOAI-SI-TEST	Provisioned	100	0	0%	2024-03-10 20:00:00 (EDT)	Auto-renew	Unused commitment >
switzerlandnorth	AOAI-SWN-TEST	Provisioned	50	0	0%	2024-03-10 20:00:00 (EDT)	Do not auto-renew	None

약정 관리 보기에서 다음과 같은 몇 가지 작업을 수행할 수 있습니다.

- 새 약정을 구매하거나 기존 약정을 편집합니다.
- 구독의 모든 약정을 모니터링합니다.
- 예기치 않은 청구를 일으킬 수 있는 약정을 식별하고 조치를 취합니다.

아래 섹션에서는 이러한 작업을 안내합니다.

프로비전된 처리량 약정 구매

약정 계획이 준비되면 다음 단계는 약정을 만드는 것입니다. 약정은 Azure OpenAI Studio를 통해 수동으로 생성되며 구독 수준에서 기여자 또는 Cognitive Services 기여자 역할을 갖도록 약정을 만드는 사용자에게 필요합니다.

새로 만들어야 하는 각 약정에 대해 다음 단계를 수행합니다.

1. 프로비전된 할당량>**프로비전된 약정 관리를 선택하여 프로비전된>처리량 구매 대화 상자를 시작합니다.**

The screenshot shows the Azure OpenAI Studio interface with the 'Quotas' page selected. On the left sidebar, the 'Quotas' option is highlighted with a red box. In the main content area, the 'Subscription' dropdown is set to 'Visual Studio Enterprise ...' and the 'Region' dropdown is set to 'SWEDENCENTRAL'. The 'Provisioned' tab under 'Standard' is selected and highlighted with a red box. Below it, the 'Manage Commitments' button is also highlighted with a red box. The table below lists provisioned managed throughput units.

Quota name	Deployment
Provisioned Managed Throughput Unit - GPT-35-Turbo-1106	
Provisioned Managed Throughput Unit - GPT-4	
Provisioned Managed Throughput Unit - GPT-4-32k	
Provisioned Managed Throughput Unit - GPT-4-Turbo	

2. 구매 약정을 선택합니다.

3. Azure OpenAI 리소스를 선택하고 약정을 구매합니다. 리소스는 기존 약정을 사용하여 리소스로 나뉘어 있으며, 편집할 수 있으며 현재 약정이 없는 리소스가 표시됩니다.

[+] 테이블 확장

설정	주의
리소스 선택	프로비전된 배포를 만들 리소스를 선택합니다. 약정을 구매한 후에는 현재 약정이 만료될 때까지 다른 리소스에서 CPU를 사용할 수 없습니다.
약정 유형 선택	프로비전됨을 선택합니다. 프로비전은 프로비저닝된 관리와 동일합니다.
커밋되지 않은 현재 프로비저닝된 할당량	이 리소스에 커밋할 수 있는 현재 사용할 수 있는 PTU 수입니다.

설정	주의
커밋할 금액(PTU)	커밋할 PTU 수를 선택합니다. 약정 기간 동안 이 수를 늘릴 수 있지만 줄일 수는 없습니다. 프로비전된 약정 유형에 대해 50씩 값을 입력합니다.
현재 기간의 약정 계층	약정 기간은 1개월로 설정됩니다.
갱신 설정	현재 PTU에서 자동 갱신 낮은 CPU에서 자동 갱신 자동 갱신 안 함

4. 구매를 선택합니다. 확인 대화 상자가 표시됩니다. 확인한 후에는 PTU가 커밋되고 이를 사용하여 프로비전된 배포를 만들 수 있습니다. |

Purchase provisioned commitment ×

Purchase a monthly commitment for your Provisioned Throughput Units (PTUs) to enable you to use your quota to create provisioned deployments.

Select a resource *

Select commitment type *

Amount to commit (PTU) *

Amounts entered will be rounded to the nearest increment of 100

Commitment tier for current period

1 month

Renewal settings *

Purchase

Cancel
+

ⓘ 중요

새로운 약정은 전체 기간에 대해 선불로 청구됩니다. 갱신 설정이 자동 갱신으로 설정된 경우 갱신 설정에 따라 각 갱신 날짜에 다시 요금이 청구됩니다.

기존 프로비전된 처리량 약정 편집

약정 관리 보기에서 기존 약정을 편집할 수도 있습니다. 기존 약정에 대해 수행할 수 있는 변경에는 두 가지 유형이 있습니다.

- 약정에 PTU를 추가할 수 있습니다.
- 갱신 설정을 변경할 수 있습니다.

약정을 편집하려면 편집할 현재를 선택한 다음 약정 편집을 선택합니다.

기존 약정에 프로비전된 처리량 단위 추가

기존 약정에 PTU를 추가하면 리소스 내에서 더 크거나 더 많은 배포를 만들 수 있습니다. 약정 기간 동안 언제든지 이 작업을 수행할 수 있습니다.

Edit provisioned commitment

X

Edit your commitment to increase the committed Provisioned Throughput Units (PTUs) or change the renewal policy.

Resource

AOAI-SI-TEST

Commitment type

Provisioned

Current PTUs committed

100

Current uncommitted quota

1000

Add PTUs to current commitment (increments of 50) *

100

New total of PTUs

200

Amounts entered will be rounded to the nearest increment of 50

Commitment tier for current period

1 month

Renewal settings *

Auto-renew at current PTUs



Purchase

Cancel



ⓘ 중요

약정에 PTU를 추가하면 현재 날짜부터 기존 약정 기간이 끝날 때까지 비례 배분된 금액으로 즉시 청구됩니다. PTU를 추가해도 약정 기간은 다시 설정되지 않습니다.

갱신 설정 변경

약정 갱신 설정은 약정 만료 날짜 이전에 언제든지 변경할 수 있습니다. 갱신 설정을 변경하려는 이유는 자동 갱신을 하지 않도록 약정을 설정하여 프로비전된 처리량 사용을 종료하거나 다음 기간에 커밋될 PTU 수를 줄여 프로비전된 처리량의 사용량을 줄이는 것입니다.

① 중요

리소스의 배포에 리소스 약정보다 더 많은 PTU가 필요하도록 약정이 만료되거나 크기가 감소하도록 허용하는 경우 초과 CPU에 대해 시간당 초과분 요금이 부과됩니다. 예를 들어 총 500PTU와 300PTU에 대한 약정을 포함하는 배포가 있는 리소스는 200PTU에 대한 시간당 초과분 요금을 생성합니다.

약정 모니터링 및 예기치 않은 청구 방지

약정 관리 창은 지정된 Azure 구독 내에서 약정 및 PTU 사용량이 있는 모든 리소스에 대한 구독 전체 개요를 제공합니다. 특히 중요도는 다음과 같습니다.

- 커밋, 배포 및 사용량** - 이러한 수치는 약정 크기 및 배포에서 사용 중인 양을 제공합니다. 커밋된 모든 CPU를 사용하여 투자를 최대화합니다.
- 만료 정책 및 날짜** - 만료 날짜 및 정책은 약정이 만료되는 시기와 만료 시점을 알려줍니다. 자동 갱신으로 설정된 약정은 갱신 날짜에 청구 이벤트를 생성합니다. 만료되는 약정의 경우 시간별 초과분 청구를 방지하기 위해 만료 날짜 이전에 이러한 리소스에서 배포를 삭제해야 합니다. 약정에 대한 현재 갱신 설정입니다.
- 알림** - 사용되지 않는 약정과 같은 중요한 조건 및 청구 초과가 발생할 수 있는 구성에 대한 경고입니다. 약정이 만료되고 배포가 여전히 존재하지만 시간당 청구로 전환된 경우와 같은 상황에서 청구 초과가 발생할 수 있습니다.

일반적인 약정 관리 시나리오

프로비전된 처리량 사용 중단

프로비전된 처리량의 사용을 종료하고 약정 만료 후 시간별 초과분 요금을 방지하려면 현재 약정이 만료된 후 요금을 중지하려면 다음 두 단계를 수행해야 합니다.

- 자동 갱신 안 함으로 모든 약정에 대해 갱신 정책을 설정합니다.
- 할당량을 사용하여 프로비전된 배포를 삭제합니다.

동일한 구독/지역의 새 리소스로 약정/배포 이동

Azure OpenAI Studio에서는 배포 또는 약정을 새 리소스로 직접 이동할 수 없습니다. 대신 대상 리소스에 새 배포를 만들고 트래픽을 이동해야 합니다. 이 작업을 수행하려면 새 리소스에 설정된 약정이 필요합니다. 약정 금액은 30일 동안 선불로 청구되므로 겹치는 동안 새 약정 및 "이중 청구"와 겹치는 것을 최소화하기 위해 원래 약정 만료와 함께 이 이동 시간이 필요합니다.

이 전환을 구현하기 위해 수행할 수 있는 두 가지 방법이 있습니다.

옵션 1: 겹치지 않는 전환

이 옵션을 사용하려면 약간의 가동 중지 시간이 필요하지만 추가 할당량이 필요하지 않으며 추가 비용이 발생하지 않습니다.

테이블 확장

단계	주의
만료할 기존 약정에 대한 한 개신 정책 설정	이렇게 하면 약정이 갱신되고 추가 요금이 발생하지 않습니다.
기존 약정이 만료되기 전에 배포를 삭제합니다.	가동 중지 시간은 이 시점에서 시작되며 새 배포가 생성되고 트래픽이 이동될 때까지 지속됩니다. 삭제 시간이 만료 날짜/시간에 최대한 가깝게 발생하도록 타이밍을 지정하여 기간을 최소화합니다.
기존 약정이 만료된 후 새 리소스에 대한 약정을 만듭니다.	만료 후 가능한 한 빨리 이 단계와 다음 단계를 실행하여 가동 중지 시간을 최소화합니다.
새 리소스에 배포를 만들고 트래픽을 해당 리소스로 이동합니다.	

옵션 2: 겹치는 전환

이 옵션은 기존 배포와 새 배포를 동시에 라이브로 설정하여 가동 중지 시간이 없습니다. 이렇게 하려면 새 배포를 만드는 데 할당량을 사용할 수 있어야 하며 겹치는 배포 기간 동안 추가 비용이 발생합니다.

테이블 확장

단계	주의
만료할 기존 약정에 대한 한 개신 정책 설정	이렇게 하면 약정이 갱신되고 추가 요금이 발생하지 않습니다.
기존 약정이 만료되기 전:	기존 약정이 만료되기 전에 모든 단계에 대해 충분한 시간을 남겨 두세요. 그렇지 않으면 초과분 요금이 생성됩니다(다음 섹션 참조).
1. 새 리소스에 대한 약정	

단계	주의
	<p>정을 만듭니다.</p> <p>2. 새 배포를 만듭니다.</p> <p>3. 트래픽 전환</p> <p>4. 기존 배포 삭제</p>

최종 단계가 예상보다 오래 걸리고 기존 약정이 만료된 후 완료되는 경우 초과분 요금을 최소화하는 세 가지 옵션이 있습니다.

- **자동 중지 시간:** 원래 배포를 삭제한 다음 이동을 완료합니다.
- **초과분 지불:** 원래 배포를 유지하고 트래픽을 이동하고 배포를 삭제할 때까지 매시 간 지불합니다.
- **원래 약정** 을 다시 설정하여 한 번 더 갱신합니다. 이렇게 하면 알려진 비용으로 이동을 완료할 수 있습니다.

초과분에 대해 지불하고 원래 약정을 다시 설정하면 원래 만료 날짜 이후의 요금이 발생합니다. 초과분 요금을 지불하는 것은 이동을 완료하는 데 하루 이틀만 필요한 경우 새로운 1개월 약정보다 저렴할 수 있습니다. 두 옵션의 비용을 비교하여 가장 저렴한 방법을 찾습니다.

배포를 새 지역 및 구독으로 이동

모든 경우에 새 위치에 사용 가능한 할당량이 필요하다는 점을 제외하고 지역 내에서 약정 및 배포를 이동하는 경우에도 동일한 방법이 적용됩니다.

기존 리소스 보기 및 편집

Azure OpenAI Studio에서 할당량>프로비전된>관리 약정을 선택하고 기존 약정을 사용하여 리소스를 보거나 변경합니다.

다음 단계

- [프로비전된 처리량 단위\(PTU\) 시작 가이드](#)
- [프로비전된 처리량 단위\(PTU\) 개념](#)

Azure OpenAI Service에서 프로비전된 배포 사용 시작

아티클 • 2024. 02. 22.

다음 가이드에서는 Azure OpenAI Service 리소스를 사용하여 프로비전된 배포를 설정하는 과정을 안내합니다.

필수 조건

- Azure 구독 - [체험 구독 만들기](#)
- 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한. 현재 이 서비스에 대한 액세스는 애플리케이션을 통해 이루어집니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI Service에 대한 액세스를 신청할 수 있습니다.
- 프로비전된 배포에 대한 할당량을 획득하고 약정을 구매했습니다.

① 참고

PTU(프로비전된 처리량 단위)는 Azure OpenAI의 표준 할당량과 다르며 기본적으로 사용할 수 없습니다. 이 서비스에 대해 자세히 알아보려면 Microsoft 계정 팀에 문의하세요.

프로비전된 배포 만들기

할당량에 대한 약정을 구매한 후 배포를 만들 수 있습니다. 프로비전된 배포를 만들려면 다음 단계를 따릅니다. 설명된 선택 사항은 스크린샷에 표시된 항목을 반영합니다.

Deploy model

X

Set up a deployment to make API calls against a provided base model or a custom model. Finished deployments are available for use. Your deployment status will move to succeeded when the deployment is complete and ready for use.

(i) Selected model version does not have a standard deployment type.

Select a model (i)

gpt-4

Model version (i)

0613 (Default)

Deployment name (i)

gpt-4

⚙️ Advanced options ▾

Content Filter (i)

Default

Deployment type

Provisioned-Managed

(i) 4300 Provisioned Throughput Units available for deployment

Provisioned throughput units (PTU) (i)



100



Create

Cancel



1. Azure OpenAI Studio [\[링크\]](#)에 로그인합니다.
2. 프로비전된 배포에 사용하도록 설정된 구독을 선택하고 할당량이 있는 지역에서 원하는 리소스를 선택합니다.
3. 왼쪽 탐색 메뉴의 관리에서 배포를 선택합니다.
4. 새 배포 만들기를 선택하고 다음 필드를 구성합니다. '고급 옵션' 드롭다운을 확장합니다.
5. 각 필드에 값을 작성합니다. 예를 들면 다음과 같습니다.

필드	Description	예시
모델 선택	배포하려는 특정 모델을 선택합니다.	GPT-4
모델 버전	배포할 모델 버전을 선택합니다.	0613
배포 이름	배포 이름은 코드에서 클라이언트 라이브러리 및 REST API를 사용하여 모델을 호출하는 데 사용됩니다.	gpt-4
콘텐츠 필터	배포에 적용할 필터링 정책을 지정합니다. 콘텐츠 필터링 방법 에 대해 자세히 알아봅니다.	기본값
배포 유형	이는 처리량과 성능에 영향을 미칩니다. 프로비전된 배포를 위해 프로비전-관리를 선택합니다.	프로비전-관리
프로비전된 처리량 단위	배포에 포함할 처리량을 선택합니다.	100

프로그래밍 방식으로 배포를 만들려면 다음 Azure CLI 명령을 사용하면 됩니다. 원하는 프로비전된 처리량 단위 수로 `sku-capacity`를 업데이트합니다.

cli

```
az cognitiveservices account deployment create \
--name <myResourceName> \
--resource-group <myResourceGroupName> \
--deployment-name MyModel \
--model-name GPT-4 \
--model-version 0613 \
--model-format OpenAI \
--sku-capacity 100 \
--sku-name ProvisionedManaged
```

REST, ARM 템플릿, Bicep 및 Terraform을 사용하여 배포를 만들 수도 있습니다. [할당량 관리](#) 방법 가이드에서 배포 자동화 섹션을 참조하고 `sku.name`을 "표준"이 아닌 "ProvisionedManaged"로 바꿉니다.

첫 번째 호출

프로비전된 배포에 대한 유추 코드는 표준 배포 유형과 동일합니다. 다음 코드 조각은 GPT-4 모델에 대한 채팅 완료 호출을 보여 줍니다. 이러한 모델을 프로그래밍 방식으로 처음 사용하는 경우 [빠른 시작 가이드](#)부터 시작하는 것이 좋습니다. 라이브러리 내에 다시 시도 논리가 포함되어 있으므로 버전 1.0 이상의 OpenAI 라이브러리를 사용하는 것이 좋습니다.

Python

```
#Note: The openai-python library support for Azure OpenAI is in preview.
import os
from openai import AzureOpenAI

client = AzureOpenAI(
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT"),
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2023-05-15"
)

response = client.chat.completions.create(
    model="gpt-4", # model = "deployment_name".
    messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Does Azure OpenAI support customer managed keys?"},
        {"role": "assistant", "content": "Yes, customer managed keys are supported by Azure OpenAI."},
        {"role": "user", "content": "Do other Azure AI services support this too?"}
    ]
)

print(response.choices[0].message.content)
```

ⓘ 중요

프로덕션의 경우 [Azure Key Vault](#)와 같은 자격 증명을 안전하게 저장하고 액세스하는 방법을 사용합니다. 자격 증명 보안에 대한 자세한 내용은 Azure AI 서비스 [보안](#) 문서를 참조하세요.

예상 처리량 이해

엔드포인트에서 달성할 수 있는 처리량은 배포된 PTU 수, 입력 크기, 출력 크기 및 호출 속도의 요소입니다. 동시 호출 수와 처리된 총 토큰 수는 이러한 값에 따라 달라질 수 있습니다. 배포 처리량을 결정하는 권장 방법은 다음과 같습니다.

1. 크기 예상을 위해 용량 계산기를 사용합니다. Azure OpenAI Studio의 할당량 페이지 및 프로비전됨 탭에서 용량 계산기를 찾을 수 있습니다.
2. 실제 트래픽 워크로드를 사용하여 부하를 벤치마킹합니다. 벤치마킹에 대한 자세한 내용은 [벤치마킹](#) 섹션을 참조하세요.

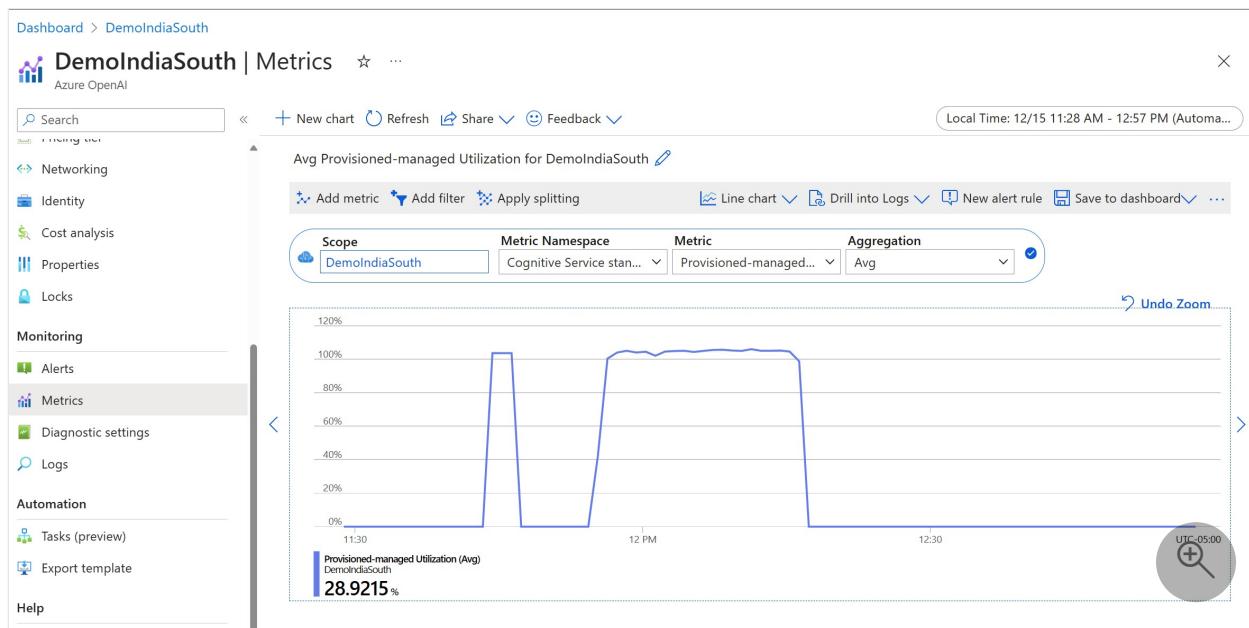
배포 사용률 측정

지정된 수의 PTU(프로비전된 처리량 단위)를 배포하면 해당 엔드포인트에서 설정된 양의 유추 처리량을 사용할 수 있습니다. 이 처리량의 활용은 모델, 모델 버전 호출 속도, 프롬프트 크기, 생성 크기를 기반으로 하는 복잡한 수식입니다. 이 계산을 간소화하기 위해 Azure Monitor에서 사용률 메트릭을 제공합니다. 사용률이 100% 이상으로 증가한 후 배포에서는 새 호출에 대해 429를 반환합니다. 프로비전된 사용률은 다음과 같이 정의됩니다.

PTU 배포 사용률 = (해당 기간에 소비된 PTU)/(해당 기간에 배포된 PTU)

리소스에 대한 Azure-Monitor 섹션에서 사용률 측정값을 찾을 수 있습니다.

<https://portal.azure.com>에 대한 모니터링 대시보드 로그인에 액세스하려면 Azure OpenAI 리소스로 이동하여 왼쪽 탐색 메뉴에서 메트릭 페이지를 선택합니다. 메트릭 페이지에서 '프로비전 관리 사용률' 측정값을 선택합니다. 리소스에 배포가 두 개 이상인 경우 '분할 적용' 단추를 클릭하여 각 배포별로 값을 분할해야 합니다.



배포 모니터링에 대한 자세한 내용은 [Azure OpenAI Service 모니터링](#) 페이지를 참조하세요.

높은 사용률 처리

프로비전된 배포는 특정 모델을 실행하기 위해 할당된 컴퓨팅 용량을 제공합니다. Azure Monitor의 '프로비전 관리 활용률' 메트릭은 배포 활용률을 1분 단위로 측정합니다. 프로비전-관리 배포는 수락된 호출이 일관된 호출별 최대 대기 시간으로 처리되도록 최적화되었습니다. 워크로드가 할당된 용량을 초과하면 서비스는 사용률이 100% 아래로 떨어질 때까지 429 HTTP 상태 코드를 반환합니다. 다시 시도 전 시간은 각각 초와 밀리초 단위로 시간을 제공하는 `retry-after` 및 `retry-after-ms` 응답 헤더에 제공됩니다. 이 방식은 호출별 대기 시간 대상을 유지하는 동시에 개발자에게 부하가 높은 상황(예: 다시 시도

또는 다른 환경/엔드포인트로 전환)을 처리하는 방법을 제어할 수 있는 권한을 부여합니다.

429 응답을 받으면 어떻게 해야 하나요?

429 응답은 호출 시 할당된 PTU가 완전히 소비되었음을 나타냅니다. 응답에는 다음 호출이 수락되기까지 기다려야 하는 시간을 알려 주는 `retry-after-ms` 및 `retry-after` 헤더가 포함되어 있습니다. 429 응답을 처리하기 위해 선택하는 방법은 애플리케이션 요구 사항에 따라 다릅니다. 다음은 몇 가지 고려 사항입니다.

- 호출당 대기 시간이 길어도 괜찮다면 클라이언트 쪽 다시 시도 논리를 구현하여 `retry-after-ms` 시간을 기다렸다가 다시 시도합니다. 이 방식을 사용하면 배포 시 처리량을 최대화할 수 있습니다. Microsoft에서 제공하는 클라이언트 SDK는 이미 적절한 기본값으로 이를 처리합니다. 사용 사례에 따라 추가 튜닝이 필요할 수도 있습니다.
- 트래픽을 다른 모델, 배포 또는 환경으로 리디렉션하는 것이 좋습니다. 이 방식은 429 신호를 받는 즉시 이 작업을 수행할 수 있으므로 대기 시간이 가장 짧은 솔루션입니다. 429 신호는 높은 사용률을 추진할 때 예기치 않은 오류 응답이 아니라 프로비전된 배포에 대한 큐 및 높은 로드를 관리하기 위한 디자인의 일부입니다.

클라이언트 라이브러리 내에서 다시 시도 논리 설정

Azure OpenAI SDK는 기본적으로 클라이언트에서 백그라운드에서 429 응답을 다시 시도합니다(최대 다시 시도까지). 라이브러리는 `retry-after` 시간을 존중합니다. 사용자 환경에 더 적합하도록 다시 시도 동작을 수정할 수도 있습니다. 다음은 Python 라이브러리의 예입니다.

`max_retries` 옵션을 사용하여 다시 시도 설정을 구성하거나 사용하지 않도록 설정할 수 있습니다.

Python

```
from openai import AzureOpenAI

# Configure the default for all requests:
client = AzureOpenAI(
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT"),
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2023-05-15",
    max_retries=5,# default is 2
)

# Or, configure per-request:
client.with_options(max_retries=5).chat.completions.create(
    model="gpt-4", # model = "deployment_name".
```

```
messages=[  
    {"role": "system", "content": "You are a helpful assistant."},  
    {"role": "user", "content": "Does Azure OpenAI support customer  
managed keys?"},  
    {"role": "assistant", "content": "Yes, customer managed keys are  
supported by Azure OpenAI."},  
    {"role": "user", "content": "Do other Azure AI services support this  
too?"}  
]  
)
```

벤치마크 실행

인스턴스의 정확한 성능과 처리량은 요청 종류와 정확한 워크로드에 따라 달라집니다. 워크로드의 처리량을 결정하는 가장 좋은 방법은 자체 데이터에 대한 벤치마크를 실행하는 것입니다.

이 작업을 지원하기 위해 벤치마킹 도구는 배포에서 벤치마크를 쉽게 실행할 수 있는 방법을 제공합니다. 이 도구는 사전 구성된 여러 가지 워크로드 형태와 함께 제공되며 주요 성능 메트릭을 출력합니다. GitHub 리포지토리: <https://aka.ms/aoai/benchmarking> 에서 도구 및 구성 설정에 대해 자세히 알아봅니다.

다음 워크플로를 권장합니다.

1. 용량 계산기를 사용하여 처리량 PTU를 예상합니다.
2. 이 트래픽 형태로 장기간(10분 이상) 벤치마크를 실행하여 안정적인 상태의 결과를 관찰합니다.
3. 벤치마크 도구 및 Azure Monitor에서 사용률, 처리된 토큰 및 호출 속도 값을 관찰합니다.
4. 클라이언트 구현을 사용하여 고유한 트래픽 형태와 워크로드로 벤치마크를 실행합니다. Azure Openai 클라이언트 라이브러리 또는 사용자 지정 논리를 사용하여 다시 시도 논리를 구현해야 합니다.

다음 단계

- 클라우드 애플리케이션 우수사례에 대한 자세한 내용은 [클라우드 애플리케이션 우수사례](#)를 참조하세요.
- 프로비전된 배포에 대한 자세한 내용은 [프로비전 처리량이란?](#)을 참조하세요.
- 각 SDK 내의 다시 시도 논리에 대한 자세한 내용은 다음을 확인합니다.
 - [Python 참조 설명서](#)
 - [.NET 참조 설명서](#)
 - [Java 참조 설명서](#)
 - [JavaScript 참조 설명서](#)

- GO 참조 설명서 ↴

Azure OpenAI 서비스 비용 관리 계획

아티클 • 2024. 04. 11.

이 문서에서는 Azure OpenAI Service의 비용을 플랜 및 관리하는 방법을 설명합니다. 서비스를 배포하기 전에 Azure 가격 계산기를 사용하여 Azure OpenAI의 비용을 예측합니다. 이후 Azure 리소스를 배포할 때 예상 비용을 검토합니다. Azure OpenAI 리소스 사용을 시작한 후 Cost Management 기능을 사용하여 예산을 설정하고 비용을 모니터링합니다.

예측 비용을 검토하고 지출 추세를 파악하여 작업할 수 있는 영역을 식별할 수도 있습니다. Azure OpenAI Service 비용은 Azure 청구서의 월별 비용의 일부일 뿐입니다. 이 문서에서는 Azure OpenAI에 대한 비용 계획 및 관리에 대해 설명하지만, 사용자에게는 타사 서비스를 비롯한 Azure 구독에 사용되는 모든 Azure 서비스 및 리소스에 대해 요금이 청구됩니다.

필수 조건

Cost Management에서의 비용 분석은 대부분의 Azure 계정 유형을 지원하지만 일부는 지원하지 않습니다. 지원되는 계정 유형의 전체 목록을 보려면 [Cost Management 데이터 이해](#)를 참조하세요. 비용 데이터를 보려면 최소한 Azure 계정에 대한 읽기 권한이 있어야 합니다. Azure Cost Management 데이터에 액세스하는 방법에 대한 정보는 [데이터에 대한 액세스 할당](#)을 참조하세요.

Azure OpenAI를 사용하기 전에 비용 예측

[Azure 가격 책정 계산기](#)를 사용하여 Azure OpenAI 사용 비용을 예측합니다.

Azure OpenAI 전체 청구 모델 이해

Azure OpenAI Service는 새 리소스를 배포할 때 비용이 발생하는 Azure 인프라에서 실행됩니다. 다른 인프라 비용이 발생할 수도 있습니다. 다음 섹션에서는 Azure OpenAI Service에 대한 요금이 청구된 방법을 설명합니다.

기본 시리즈 및 Codex 시리즈 모델

Azure OpenAI 기본 시리즈 및 Codex 시리즈 모델은 토큰 1,000개당 요금이 청구됩니다. 비용은 Ada, Babbage, Curie, Davinci 또는 Code-Cushman 중 선택한 모델 시리즈에 따라 달라집니다.

Azure OpenAI 모델은 텍스트를 토큰으로 세분화하여 이해하고 처리합니다. 참고로 각 토큰은 일반적인 영어 텍스트의 경우 대략 4자입니다.

토큰 비용은 입출력 모두에 대한 비용입니다. 예를 들어 Azure OpenAI 모델에 Python으로 변환하도록 요청하는 1,000개의 토큰 JavaScript 코드 샘플이 있는 경우입니다. 전송된 초기 입력 요청에 대해서는 약 1,000개의 토큰이 청구되고 총 2,000개의 토큰에 대한 응답으로 받은 출력에 대해서는 1,000개의 토큰이 추가로 청구됩니다.

실제로 이러한 유형의 완료 호출의 경우 토큰 입력/출력은 완벽하게 1:1이 아닐 수 있습니다. 한 프로그래밍 언어에서 다른 프로그래밍 언어로 변환하면 다양한 요인에 따라 출력이 길거나 짧아질 수 있습니다. 이러한 요소 중 하나는 `max_tokens` 매개 변수에 할당된 값입니다.

기본 시리즈 및 Codex 시리즈 미세 조정된 모델

Azure OpenAI 미세 조정된 모델은 다음 세 가지 요소에 따라 요금이 청구됩니다.

- 학습 시간
- 호스팅 시간
- 1,000개 토큰당 추론

호스팅 시간 비용은 미세 조정된 모델이 배포되면 적극적으로 사용되는지 여부에 관계없이 시간당 비용이 계속 발생하기 때문에 주의해야 합니다. 미세 조정된 모델 비용을 면밀히 모니터링합니다.

① 중요

사용자 지정된 모델을 배포한 후 언제든지 배포가 15일 이상 비활성 상태로 유지되면 배포가 삭제됩니다. 모델이 배포된 지 15일이 넘었고 연속 15일 동안 모델에 대한 완료 또는 채팅 완료 호출이 이루어지지 않은 경우 맞춤형 모델 배포는 비활성 상태입니다.

비활성 배포를 삭제해도 기본 사용자 지정 모델은 삭제되거나 영향을 받지 않으며 사용자 지정 모델은 언제든지 다시 배포될 수 있습니다.

배포된 각 사용자 지정(미세 조정된) 모델은 완료 또는 채팅 완료 호출이 모델에 대해 이루어지는지 여부에 관계없이 시간당 호스팅 비용이 발생합니다..

Azure OpenAI Service로 인해 발생할 수 있는 기타 비용

Azure Monitor 로그로의 데이터 전송 및 경고와 같은 기능을 사용하도록 설정하면 해당 서비스에 대한 추가 비용이 발생합니다. 이러한 비용은 다른 서비스 및 구독 수준에서 볼

수 있지만 Azure OpenAI 리소스로만 범위가 지정되는 경우에는 볼 수 없습니다.

Azure OpenAI Service와 함께 Azure 선불 사용

Azure 선불 크레딧을 사용하여 Azure OpenAI Service 요금을 지불할 수 있습니다. 그러나 Azure 선불 크레딧을 사용하여 Azure Marketplace에 있는 항목을 포함한 타사 제품 및 서비스에 대한 요금을 지불할 수는 없습니다.

Azure OpenAI Service의 HTTP 오류 응답 코드 및 청구 상태

서비스가 처리를 수행하는 경우 상태 코드가 성공하지 않은 경우(200이 아님)에도 요금이 청구될 수 있습니다. 예를 들어, 콘텐츠 필터나 입력 제한으로 인한 400 오류 또는 시간 제한으로 인한 408 오류가 있습니다.

서비스가 처리를 수행하지 않으면 요금이 청구되지 않습니다. 예를 들어, 인증으로 인한 401 오류 또는 속도 제한 초과로 인한 429 오류가 있습니다.

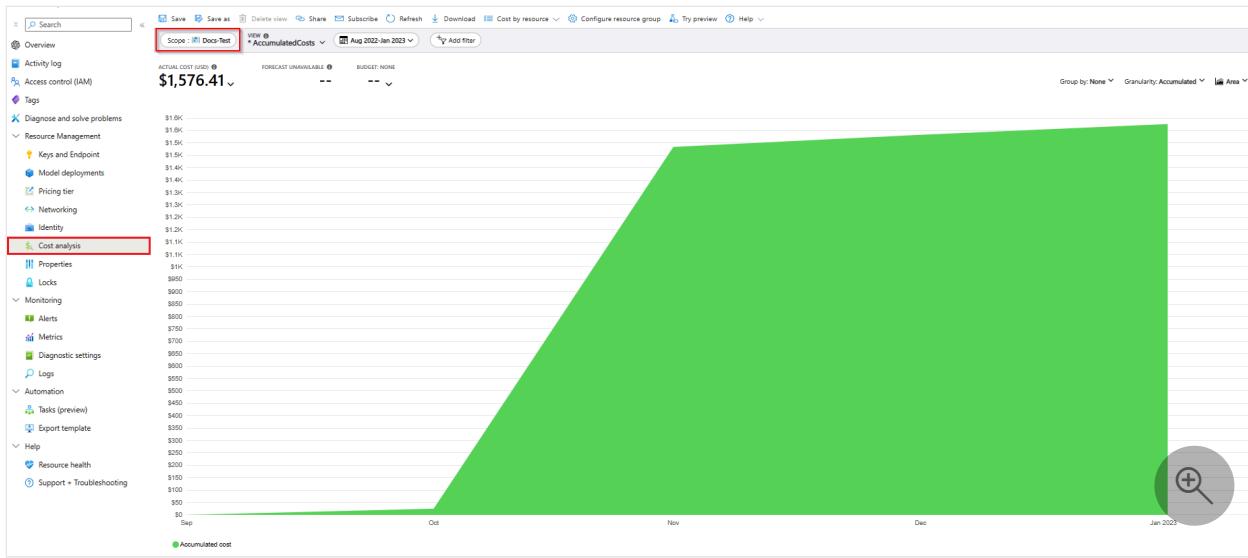
비용 모니터링

Azure OpenAI를 사용하여 Azure 리소스를 사용하는 경우 비용이 발생합니다. Azure 리소스 사용량 단위 비용은 시간 간격(초, 분, 시간 및 일) 또는 단위 사용량(바이트, 메가바이트 등)에 따라 달라집니다. Azure OpenAI 사용이 시작되는 즉시 비용이 발생하며 [비용 분석](#)에서 비용을 볼 수 있습니다.

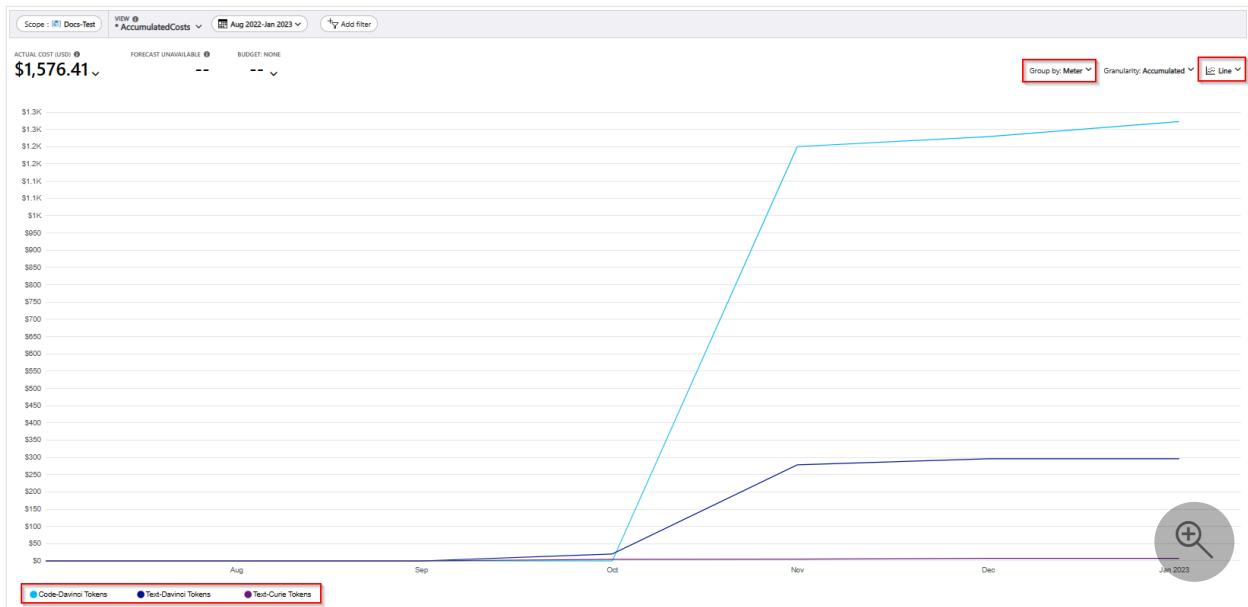
비용 분석을 사용하면 서로 다른 시간 간격에 대한 그래프 및 테이블의 Azure OpenAI 비용을 볼 수 있습니다. 몇 가지 예로 일, 현재 달과 이전 달 및 연도에 따라 확인할 수 있습니다. 예산 및 예상 비용에 대한 비용도 조회할 수 있습니다. 시간이 지남에 따라 더 긴 보기로 전환하면 지출 추세를 파악하는 데 도움이 됩니다. 과도한 지출이 발생한 위치를 확인할 수 있습니다. 예산을 만든 경우 초과된 부분도 쉽게 확인할 수 있습니다.

비용 분석에서 Azure OpenAI 비용을 보려면 다음을 수행합니다.

1. Azure Portal에 로그인합니다.
2. Azure OpenAI 리소스 중 하나를 선택합니다.
3. 리소스 관리에서 [비용 분석](#)을 선택합니다.
4. 기본적으로 비용 분석은 개별 Azure OpenAI 리소스로 범위가 지정됩니다.



해당 비용을 구성하는 요소의 분석을 이해하려면 **그룹화 기준을 미터로** 수정하고 이 경우 차트 종류를 **꺾은선형**으로 전환하는 것이 도움이 될 수 있습니다. 이제 이 특정 리소스의 경우 비용의 출처가 비용의 대부분을 나타내는 **토큰**이 포함된 세 개의 서로 다른 모델 시리즈임을 알 수 있습니다.



Azure OpenAI와 관련된 비용을 평가할 때 범위를 이해하는 것이 중요합니다. 리소스가 동일한 리소스 그룹에 속하는 경우 해당 수준에서 비용 분석의 범위를 지정하여 비용에 미치는 영향을 이해할 수 있습니다. 리소스가 여러 리소스 그룹에 분산되어 있으면 구독 수준까지 범위를 지정할 수 있습니다.

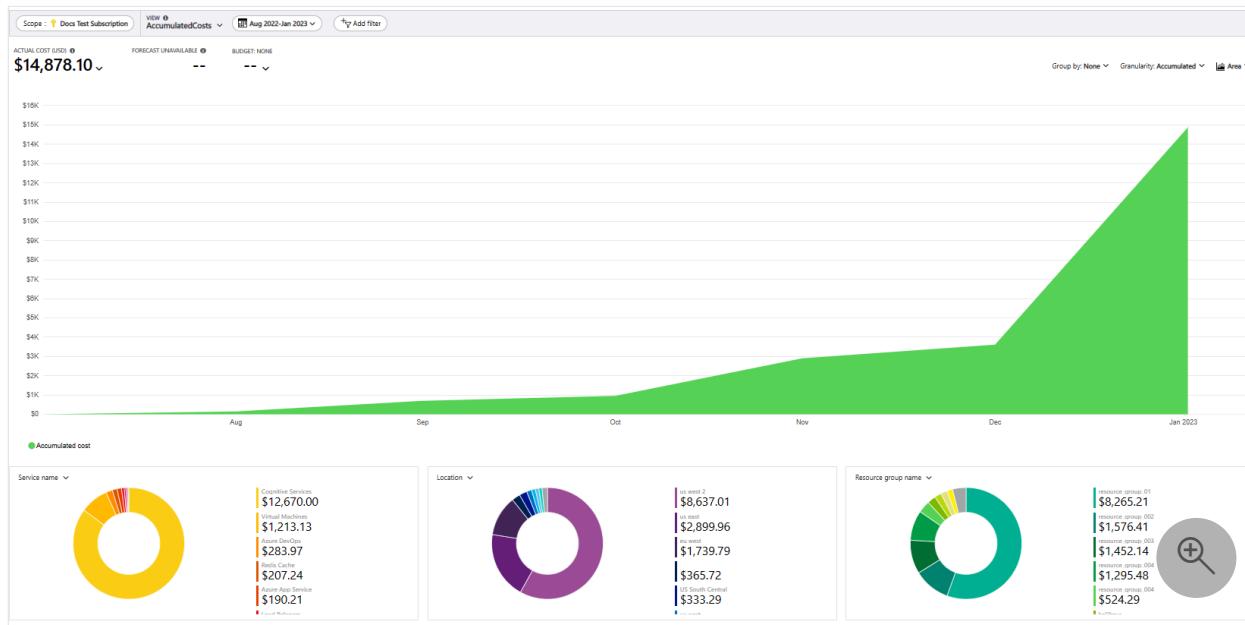
더 높은 수준에서 범위를 지정하는 경우 Azure OpenAI 사용량을 중점적으로 고려하는 더 많은 필터를 추가해야 하는 경우가 많습니다. 구독 수준에서 범위가 지정되면 Azure OpenAI 비용 관리의 컨텍스트에서 신경 쓰지 않을 수 있는 다른 많은 리소스가 표시됩니다. 구독 수준에서 범위를 지정할 때 **Cost Management** 서비스에서 전체 **비용 분석 도구**로 이동하는 것이 좋습니다.

다음은 **비용 분석 도구**를 사용하여 구독 또는 리소스 그룹에 대한 누적 비용을 확인하는 방법의 예입니다.

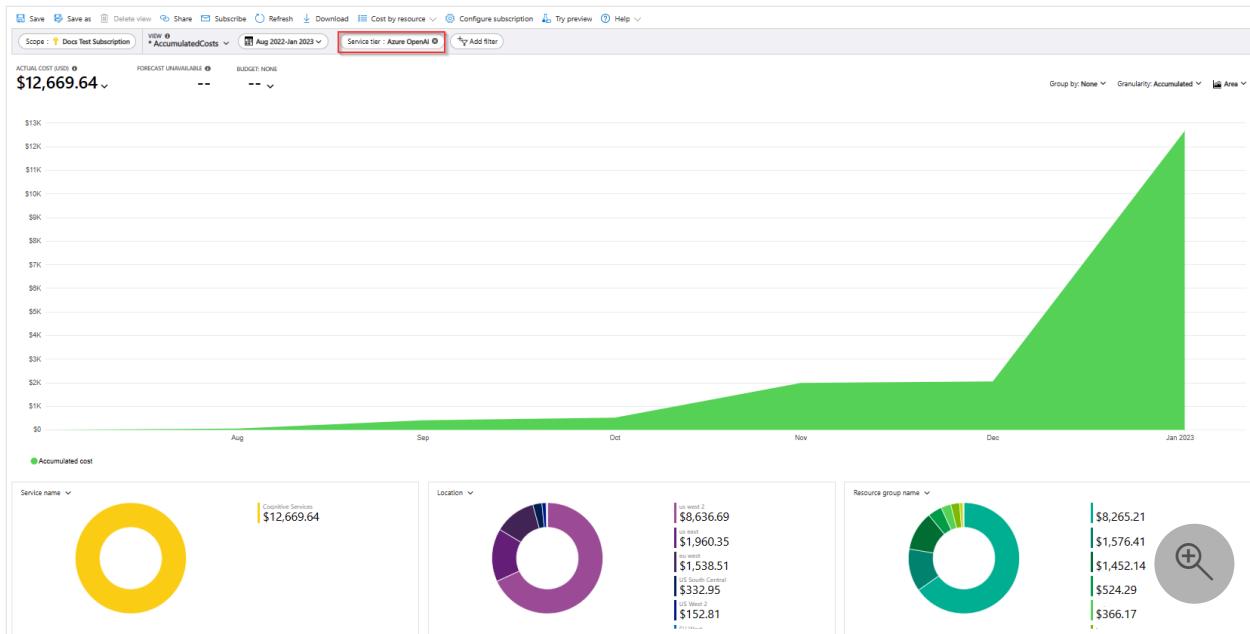
1. 위쪽 Azure 검색 창에서 *Cost Management*를 검색하여 예산 만들기와 같은 더 많은 옵션을 포함하는 전체 서비스 환경으로 이동합니다.
2. 필요한 경우 **범위**:가 분석하려는 리소스 그룹 또는 구독을 가리키지 않는 경우 **변경**을 선택합니다.
3. 왼쪽에서 **보고 + 분석>비용 분석**을 선택합니다.
4. **모든 보기** 탭에서 **누적 비용**을 선택합니다.

The screenshot shows the Azure Cost Analysis blade. On the left, a sidebar menu is open under 'Reporting + analytics', with 'Cost analysis' selected. In the center, the 'Accumulated costs' view is displayed. At the top, there's a search bar and a breadcrumb navigation: 'Scope: C+L Vendor Managed Services Subscriptions / BAH-OpenAI-RG1 (change)'. Below the search bar are tabs for 'New tab', 'Refresh', 'Try preview', 'Go to resource group', and '...'. A red box highlights the 'All views' tab. To the right of the tabs are 'Recent' and 'Settings' buttons. The main area has a heading 'Report on and analyze your cloud costs and review key insights to better understand and control spending patterns. From resources and the subscription hierarchy to the provider breaks your cost down to help you understand what you're spending and where.' Below this is a 'Recommended' section with four cards: 'Accumulated costs' (highlighted with a red box), 'Resources', 'Daily costs', and 'Services'. The 'Accumulated costs' card shows a table with three rows: Reservations, Resources, and Services, all grouped by 'This month'. At the bottom of the page is a search icon.

비용 분석 대시보드에는 **범위**에 대해 지정한 항목에 따라 분석되는 누적 비용을 보여 줍니다.



서비스별로 필터를 추가하려고 하면 목록에서 Azure OpenAI를 찾을 수 없습니다. 이 상황은 서비스 수준 필터가 **Cognitive Services**이 있기 때문에 발생합니다. 다른 유형의 Azure AI 서비스 리소스 없이 구독 전체에서 모든 Azure OpenAI 리소스를 보려면 대신 **서비스 계층: Azure OpenAI**로 범위를 지정해야 합니다.



예산 만들기

예산을 만들면 비용을 관리하고 관련자에게 비정상 지출 및 과다 지출 위험을 알리는 경고를 만들 수 있습니다. 경고는 예산 및 비용 임계값에 따른 지출을 기준으로 합니다. Azure 구독 및 리소스 그룹에 대한 예산 및 경고를 만듭니다. 이러한 항목은 전체 비용 모니터링 전략의 일부로 유용합니다.

모니터링을 더 세분화하려는 경우 Azure의 특정 리소스 또는 서비스에 대한 필터를 사용하는 예산을 만들 수도 있습니다. 필터는 추가 비용이 드는 새 리소스를 실수로 만들지 않도록 도움을 줍니다. 예산을 만들 때 사용할 수 있는 필터 옵션에 대한 자세한 내용은 [그룹 및 필터 옵션](#)을 참조하세요.

ⓘ 중요

OpenAI에는 예산 초과를 방지하는 하드 한도 옵션이 있지만 Azure OpenAI는 현재 이 기능을 제공하지 않습니다. 더 많은 고급 작업을 수행하려면 예산 알림의 일부로 작업 그룹에서 자동화를 시작할 수 있지만 이를 위해서는 사용자 지정 개발이 추가로 필요합니다.

비용 데이터 내보내기

[비용 데이터를 스토리지 계정으로 내보낼 수도 있습니다](#). 이 기능은 다른 사용자가 비용을 위해 추가 데이터 분석을 수행해야 하는 경우에 유용합니다. 예를 들어 재무 팀은 Excel 또는 Power BI를 사용하여 데이터를 분석할 수 있습니다. 매일, 매주 또는 매월 일정으로 비용을 내보내고 사용자 지정 날짜 범위를 설정할 수 있습니다. 비용 데이터 세트를 검색하려면 비용 데이터를 내보내는 것이 좋습니다.

다음 단계

- Azure Cost Management를 통해 클라우드 투자를 최적화하는 방법에 대해 알아봅니다.
- 비용 분석을 통한 비용 관리에 대해 자세히 알아봅니다.
- 예기치 않은 비용 방지 방법에 대해 알아봅니다.
- Cost Management 단계별 학습 과정을 수강합니다.

성능 및 대기 시간

아티클 • 2024. 03. 01.

이 문서에서는 Azure OpenAI에서 대기 시간 및 처리량이 작동하는 방식과 성능을 향상시키기 위해 환경을 최적화하는 방법에 대한 배경 정보를 제공합니다.

처리량 및 대기 시간 이해

애플리케이션 크기를 조정하는 경우 두 가지 주요 개념인 (1) 시스템 수준 처리량과 (2) 호출당 응답 시간(대기 시간이라고도 함)을 고려해야 합니다.

시스템 수준 처리량

여기서는 배포의 전체 용량, 즉 분당 요청 수와 처리할 수 있는 총 토큰 수가 표시됩니다.

표준 배포의 경우 배포에 할당된 할당량이 달성 가능한 처리량 크기에 어느 정도 영향을 미칩니다. 그러나 할당량은 배포 호출에 대한 허용 논리에만 영향을 주며 처리량에 직접 작용하지는 않습니다. 호출당 대기 시간 변화로 인해 할당량만큼 높은 처리량을 달성하지 못할 수 있습니다. [할당량 관리에 대해 자세히 알아봅니다.](#)

프로비전된 배포에서 설정된 양의 모델 처리 용량이 엔드포인트에 할당됩니다. 엔드포인트에서 달성을 할 수 있는 처리량은 입력 크기, 출력 크기, 호출 속도 및 캐시 일치 속도의 요소입니다. 동시 호출 수와 처리된 총 토큰 수는 이러한 값에 따라 달라질 수 있습니다. 다음 단계에서는 프로비전된 배포에서 지정된 워크로드를 가져올 수 있는 처리량을 평가하는 방법을 안내합니다.

- 크기 예상을 위해 용량 계산기를 사용합니다.
- 실제 트래픽 워크로드를 사용하여 부하를 벤치마킹합니다. Azure Monitor에서 사용률 및 토큰 처리 메트릭을 측정합니다. 연장된 기간 동안 실행합니다. [Azure OpenAI 벤치마킹 리포지토리](#)에는 벤치마크를 실행하기 위한 코드가 포함되어 있습니다. 마지막으로, 가장 정확한 방법은 사용자 고유의 데이터 및 워크로드 특성을 사용하여 테스트를 실행하는 것입니다.

GPT-4 0613 모델에 대한 몇 가지 예는 다음과 같습니다.

[\[+\] 테이블 확장](#)

프롬프트 크기(토큰)	생성 크기(토큰)	분당 호출 수	필요한 PTU
800	150	30	100

프롬프트 크기(토큰)	생성 크기(토큰)	분당 호출 수	필요한 PTU
1000	50	300	700
5000	100	50	600

PTU의 수는 워크로드 배포가 일정하게 유지되는 경우 호출 속도와 대략적으로 비례해서 (거의 선형일 수 있음) 스케일링됩니다.

대기 시간: 호출당 응답 시간

이 컨텍스트에서 대기 시간을 대략적으로 정의하면 모델에서 응답을 다시 가져오는 데 걸리는 시간으로 나타낼 수 있습니다. 완료 및 채팅 완료 요청의 경우 대기 시간은 주로 모델 유형, 프롬프트의 토큰 수 및 생성된 토큰 수에 따라 달라집니다. 일반적으로 각 프롬프트 토큰은 생성된 각 증분 토큰에 비해 약간의 시간을 추가합니다.

이러한 모델에서는 호출당 예상 대기 시간을 예측하는 것이 어려울 수 있습니다. 완료 요청의 대기 시간은 네 가지 주요 요인인 (1) 모델, (2) 프롬프트의 토큰 수, (3) 생성된 토큰 수 및 (4) 배포 및 시스템의 전체 부하에 따라 달라질 수 있습니다. 1과 3이 총 시간이 주로 기여합니다. 다음 섹션에서는 큰 언어 모델 유추 호출의 구조에 대해 좀 더 자세히 알아봅니다.

성능 향상

애플리케이션의 호출당 대기 시간을 개선하기 위해 제어할 수 있는 몇 가지 요소가 있습니다.

모델 선택

대기 시간은 사용 중인 모델에 따라 달라집니다. 동일한 요청의 경우 서로 다른 모델에서 채팅 완료 호출에 대한 대기 시간이 다를 것으로 예상합니다. 사용 사례에서 응답 시간은 가장 빠르고 대기 시간은 가장 낮은 모델이 필요한 경우 [GPT-3.5 Turbo 모델 시리즈](#)의 최신 모델을 사용하는 것이 좋습니다.

생성 크기 및 최대 토큰

Azure OpenAI 엔드포인트에 완료 요청을 보내면 입력 텍스트가 배포된 모델로 전송되는 토큰으로 변환됩니다. 모델은 입력 토큰을 받은 다음, 응답 생성을 시작합니다. 반복적인 순차 프로세스이며 한 번에 하나의 토큰이 있습니다. 이것을 `n tokens = n iterations`의 for 루프처럼 생각할 수 있습니다. 대부분의 모델에서 응답 생성은 프로세스에서 가장 느린 단계입니다.

요청 시 요청된 생성 크기(max_tokens 매개 변수)가 생성 크기의 초기 예측값으로 사용됩니다. 전체 크기를 생성하는 컴퓨팅 시간은 요청이 처리될 때 모델에 의해 예약됩니다. 생성이 완료되면 나머지 할당량이 해제됩니다. 토큰 수를 줄이는 방법은 다음과 같습니다.

- 각 호출의 `max_token` 매개 변수를 가능한 한 작게 설정합니다.
- 추가 콘텐츠 생성을 방지하기 위한 중지 시퀀스를 포함합니다.
- 더 적은 응답 생성: `best_of` 및 `n` 매개 변수는 여러 출력을 생성하기 때문에 대기 시간을 크게 늘릴 수 있습니다. 가장 빠른 응답의 경우 이러한 값을 지정하거나 1로 설정하지 마세요.

요약하면 요청당 생성된 토큰 수를 줄이면 각 요청의 대기 시간이 줄어듭니다.

스트리밍

요청에서 `stream: true`를 설정하면 서비스는 전체 토큰 시퀀스가 생성될 때까지 기다리지 않고 토큰이 사용 가능해진 즉시 토큰을 반환합니다. 모든 토큰을 가져오는 시간은 달라지지 않지만 첫 번째 응답 시간이 줄어듭니다. 이 방법은 최종 사용자가 생성되는 응답을 읽을 수 있으므로 더 나은 사용자 환경을 제공합니다.

스트리밍은 처리하는 데 시간이 오래 걸리는 큰 호출에서도 유용합니다. 많은 클라이언트 및 중간 계층에는 개별 호출에 대한 시간 제한이 있습니다. 클라이언트 쪽 시간 초과로 인해 장기 생성 호출이 취소될 수 있습니다. 데이터를 다시 스트리밍하면 충분 데이터가 수신되는지 확인할 수 있습니다.

스트리밍을 사용하는 경우의 예

채팅봇 및 대화형 인터페이스.

스트리밍은 인식된 대기 시간에 영향을 줍니다. 스트리밍을 사용하도록 설정하면 토큰이 사용 가능해지는 즉시 청크로 다시 받습니다. 최종 사용자의 경우 요청을 완료하기 위한 전체 시간은 동일하게 유지되더라도 모델이 더 빠르게 응답하는 것처럼 느껴지는 경우가 많습니다.

스트리밍이 덜 중요한 경우의 예:

감정 분석, 언어 번역, 콘텐츠 생성.

실시간 응답이 아니라 완료된 결과에만 관심이 있는 대량 작업을 수행하는 많은 사용 사례가 있습니다. 스트리밍을 사용하지 않도록 설정하면 모델이 전체 응답을 완료할 때까지 토큰을 받지 않습니다.

콘텐츠 필터링

Azure OpenAI에는 핵심 모델과 함께 작동하는 [콘텐츠 필터링 시스템](#)이 포함되어 있습니다. 이 시스템은 유해한 콘텐츠의 출력을 탐지하고 방지하기 위한 분류 모델의 양상들을 통해 프롬프트와 완료를 모두 실행하여 작동합니다.

콘텐츠 필터링 시스템은 입력 프롬프트와 출력 완료 모두에서 잠재적으로 유해한 콘텐츠의 특정 범주를 탐지하고 조치를 취합니다.

콘텐츠 필터링을 추가하면 안전성뿐만 아니라 대기 시간도 증가합니다. 이러한 성능 절충이 필요한 애플리케이션이 많이 있지만 성능 향상을 위해 콘텐츠 필터를 사용하지 않도록 설정해야 하는 저위험 사용 사례가 있습니다.

기본 [콘텐츠 필터링 정책](#)에 대한 수정을 요청하는 방법에 대해 자세히 알아봅니다.

워크로드 분리

동일한 엔드포인트에서 다른 워크로드를 혼합하면 대기 시간에 부정적인 영향을 줄 수 있습니다. 이는 (1) 유추 중에 함께 일괄 처리되고 짧은 호출이 더 긴 완료를 기다릴 수 있고 (2) 호출을 혼합하면 둘 다 동일한 공간을 위해 경쟁하므로 캐시 적중률이 줄어들 수 있기 때문입니다. 가능하면 각 워크로드에 대해 별도의 배포를 사용하는 것이 좋습니다.

프롬프트 크기

프롬프트 크기는 생성 크기보다 대기 시간에 미치는 영향이 적지만, 특히 크기가 커지는 경우 전체 시간에 영향을 줍니다.

일괄 처리

동일한 엔드포인트에 여러 요청을 보내는 경우 요청을 단일 호출로 일괄 처리할 수 있습니다. 이렇게 하면 수행해야 하는 요청 수가 줄어들고 시나리오에 따라 전반적인 응답 시간이 향상될 수 있습니다. 이 방법을 테스트하여 도움이 되는지 확인하는 것이 좋습니다.

처리량을 측정하는 방법

다음 두 가지 측정값을 사용하여 배포의 전체 처리량을 측정하는 것이 좋습니다.

- **분당 호출 수:** 분당 API 유추 호출 수입니다. Azure OpenAI 요청 메트릭을 사용하고 ModelDeploymentName으로 분할하여 Azure-monitor에서 측정할 수 있습니다.
- **분당 총 토큰 수:** 배포에서 분당 처리되는 총 토큰 수입니다. 여기에는 프롬프트 및 생성된 토큰이 포함됩니다. 배포 성능을 보다 깊이 있게 이해하기 위해 둘 다를 측정하는 방식으로 추가로 분할하는 경우가 많습니다. 처리된 유추 토큰 메트릭을 사용하여 Azure-Monitor에서 측정할 수 있습니다.

Azure OpenAI 서비스 모니터링에 대해 자세히 알아볼 수 있습니다.

호출당 대기 시간을 측정하는 방법

각 호출에 걸리는 시간은 모델을 읽고, 출력을 생성하고, 콘텐츠 필터를 적용하는 데 걸리는 시간에 따라 달라집니다. 스트리밍을 사용하는지 여부에 따라 시간을 측정하는 방법이 달라집니다. 각 사례에 대해 다른 측정값 집합을 제안합니다.

Azure OpenAI 서비스 모니터링에 대해 자세히 알아볼 수 있습니다.

비스트리밍:

- 엔드투엔드 요청 시간: API 게이트웨이에서 측정한 대로 비스트리밍 요청에 대한 전체 응답을 생성하는 데 걸린 총 시간입니다. 프롬프트 및 생성 크기가 증가함에 따라 이 수치가 증가합니다.

스트리밍:

- 응답 시간: 요청 스트리밍에 권장되는 대기 시간(응답성) 측정값입니다. PTU 및 PTU 관리형 배포에 적용됩니다. API 게이트웨이에서 측정한 대로 사용자가 프롬프트를 보낸 후 첫 번째 응답이 표시되는 데 걸린 시간으로 계산됩니다. 프롬프트 크기가 증가하거나 적중 크기가 감소하면 이 수치가 증가합니다.
- 첫 번째 토큰에서 마지막 토큰까지의 평균 토큰 생성 속도 시간을 API 게이트웨이에서 측정한, 생성된 토큰 수로 나눈 값입니다. 이렇게 하면 응답 생성 속도가 측정되며, 이 값은 시스템 부하가 증가함에 따라 증가합니다. 요청 스트리밍에 권장되는 대기 시간 측정값입니다.

요약

- 모델 대기 시간:** 모델 대기 시간이 중요한 경우 GPT-3.5 Turbo 모델 시리즈의 최신 모델을 사용해보는 것이 좋습니다.
- 최대 토큰 감소:** OpenAI는 생성된 총 토큰 수가 비슷한 경우에도 최대 토큰 매개 변수에 대해 더 높은 값이 설정된 요청의 대기 시간이 더 긴 것으로 나타났습니다.
- 생성된 총 토큰 감소:** 생성된 토큰 수가 적을수록 전체 응답이 더 빨라집니다. 이는 `n tokens = n iterations`에서 for 루프를 사용하는 것과 같습니다. 생성된 토큰 수를 줄이면 전반적인 응답 시간이 그에 따라 향상됩니다.
- 스트리밍:** 스트리밍을 사용하도록 설정하면 마지막 토큰이 준비될 때까지 기다리지 않고도 생성되는 모델 응답을 볼 수 있도록 하여 특정 상황에서 사용자가 더 나은 결

과를 예상할 수 있도록 하는 데 유용할 수 있습니다.

- 콘텐츠 필터링은 안전성을 향상시키지만 대기 시간에도 영향을 줍니다. 워크로드가 [수정된 콘텐츠 필터링 정책](#)의 이점을 활용할 수 있는지 평가합니다.

Azure OpenAI Service에 대한 역할 기반 액세스 제어

아티클 • 2024. 04. 03.

Azure OpenAI Service는 Azure 리소스에 대한 개별 액세스를 관리하기 위한 권한 부여 시스템인 Azure RBAC(Azure 역할 기반 액세스 제어)를 지원합니다. Azure RBAC를 사용하여 지정된 프로젝트에 대한 요구 사항에 따라 서로 다른 수준의 권한을 다른 팀 구성원에게 할당합니다. 자세한 내용은 [Azure RBAC 설명서](#)를 참조하세요.

Azure OpenAI 리소스에 역할 할당 추가

Azure RBAC는 Azure OpenAI 리소스에 할당할 수 있습니다. Azure 리소스에 대한 액세스 권한을 부여하려면 역할 할당을 추가합니다.

1. [Azure portal](#)에서 Azure OpenAI를 검색합니다.
2. Azure OpenAI를 선택하고 특정 리소스로 이동합니다.

① 참고

또한 전체 리소스 그룹, 구독 또는 관리 그룹에 대해 Azure RBAC를 설정할 수 있습니다. 원하는 범위 수준을 선택한 다음, 원하는 항목으로 이동하여 이 작업을 수행합니다. 예를 들어 **리소스 그룹**을 선택한 다음, 특정 리소스 그룹으로 이동합니다.

3. 왼쪽 탐색 창에서 **액세스 제어(IAM)**을 선택합니다.
4. **추가**를 선택한 다음, **역할 할당 추가**를 선택합니다.
5. 다음 화면의 **역할** 탭에서 추가할 역할을 선택합니다.
6. 구성원 탭에서 사용자, 그룹, 서비스 주체 또는 관리 ID를 선택합니다.
7. **검토 + 할당** 탭에서 **검토 + 할당**을 선택하여 역할을 할당합니다.

몇 분 이내에 선택한 범위에서 선택한 역할이 대상에 할당됩니다. 이 단계에 대한 도움말은 [Azure Portal을 사용하여 Azure 역할 할당](#)을 참조하세요.

Azure OpenAI 역할

- Cognitive Services OpenAI 사용자

- Cognitive Services OpenAI 기여자
- Cognitive Services 기여자
- Cognitive Services 사용량 읽기 권한자

① 참고

구독 수준 소유자 및 기여자 역할은 상속되며 리소스 그룹 수준에서 적용되는 사용자 지정 Azure OpenAI 역할보다 우선 순위가 높습니다.

이 섹션에서는 다양한 계정 및 계정 조합이 Azure OpenAI 리소스에 대해 수행할 수 있는 일반적인 작업에 대해 설명합니다. 사용 가능한 **작업** 및 **DataActions**의 전체 목록을 보려면 Azure OpenAI 리소스에서 개별 역할이 부여됩니다. **액세스 제어(IAM)**>**역할**>로 이동합니다. 관심 있는 역할의 **세부 정보** 열에서 **보기**를 선택합니다. 기본적으로 **작업** 방사형 버튼이 선택되어 있습니다. 역할에 할당된 기능의 전체 범위를 이해하려면 **Actions**와 **DataActions**를 모두 조사해야 합니다.

Cognitive Services OpenAI 사용자

사용자에게 Azure OpenAI 리소스에 대해서만 이 역할에 대한 역할 기반 액세스 권한이 부여된 경우 다음과 같은 일반적인 작업을 수행할 수 있습니다.

- [Azure portal](#)에서 리소스 보기
- 키 및 엔드포인트에서 리소스 엔드포인트를 봅니다.
- Azure OpenAI Studio에서 리소스 및 관련 모델 배포를 볼 수 있는 기능입니다.
- Azure OpenAI Studio에서 배포에 사용할 수 있는 모델을 볼 수 있는 기능입니다.
- 채팅, 완료 및 DALL-E(미리 보기) 플레이그라운드 환경을 사용하여 이 Azure OpenAI 리소스에 이미 배포된 모델로 텍스트와 이미지를 생성합니다.
- Microsoft Entra ID를 사용하여 유추 API 호출을 만듭니다.

이 역할만 할당된 사용자는 다음을 수행할 수 없습니다.

- 새로운 Azure OpenAI 리소스 만들기
- 키 및 엔드포인트에서 키 보기/복사/다시 생성
- 새 모델 배포 생성 또는 기존 모델 배포 수정
- 사용자 지정 미세 조정 모델 생성/배포
- 미세 조정을 위한 데이터 세트 업로드
- 액세스 할당량
- 사용자 지정 콘텐츠 필터 만들기
- 데이터 기능을 사용하기 위한 데이터 소스를 추가하세요.

Cognitive Services OpenAI 기여자

이 역할에는 Cognitive Services OpenAI 사용자의 모든 권한이 있으며 다음과 같은 추가 작업을 수행할 수도 있습니다.

- 사용자 지정 미세 조정 모델 만들기
- 미세 조정을 위한 데이터 세트 업로드
- 새 모델 배포 만들기 또는 기존 모델 배포 편집 [2023년 가을에 추가됨]

이 역할만 할당된 사용자는 다음을 수행할 수 없습니다.

- 새로운 Azure OpenAI 리소스 만들기
- 키 및 엔드포인트에서 키 보기/복사/다시 생성
- 액세스 할당량
- 사용자 지정 콘텐츠 필터 만들기
- 데이터 기능을 사용하기 위한 데이터 소스를 추가하세요.

Cognitive Services 기여자

이 역할은 일반적으로 추가 역할과 함께 사용자의 리소스 그룹 수준에서 액세스 권한을 부여합니다. 그 자체로 이 역할을 통해 사용자는 다음 작업을 수행할 수 있습니다.

- 할당된 리소스 그룹 내에 새 Azure OpenAI 리소스를 만듭니다.
- [Azure portal](#)에서 할당된 리소스 그룹의 리소스를 봅니다.
- 키 및 엔드포인트에서 리소스 엔드포인트를 봅니다.
- 키 및 엔드포인트에서 키 보기/복사/다시 생성
- Azure OpenAI Studio에서 배포에 사용할 수 있는 모델을 볼 수 있는 기능
- Chat, Completions 및 DALL-E(미리 보기) 플레이그라운드 환경을 사용하여 이 Azure OpenAI 리소스에 이미 배포된 모델로 텍스트와 이미지를 생성하세요.
- 사용자 지정 콘텐츠 필터 만들기
- 데이터 기능을 사용하기 위한 데이터 원본을 추가하세요.
- 새 모델 배포 생성 또는 기존 모델 배포 수정(API를 통해)
- 사용자 지정 미세 조정 모델 만들기 [2023년 가을에 추가됨]
- 미세 조정을 위한 데이터 세트 업로드 [2023년 가을에 추가됨]
- 새 모델 배포 만들기 또는 기존 모델 배포 편집(Azure OpenAI Studio 사용) [2023년 가을에 추가됨]

이 역할만 할당된 사용자는 다음을 수행할 수 없습니다.

- 액세스 할당량
- Microsoft Entra ID를 사용하여 유추 API 호출을 만듭니다.

Cognitive Services 사용량 읽기 권한자

할당량을 보려면 **Cognitive Services 사용량 읽기 권한자** 역할이 필요합니다. 이 역할은 Azure 구독 전체의 할당량 사용량을 보는 데 필요한 최소한의 액세스 권한을 제공합니다.

이 역할은 Azure portal의 **구독 > *액세스 제어(IAM) > 역할 할당 추가 > Cognitive Services 사용량 읽기 권한자** 검색에서 찾을 수 있습니다. 역할은 구독 수준에서 적용해야 하며 리소스 수준에는 없습니다.

이 역할을 사용하지 않으려면 구독 **읽기 권한자** 역할이 동등한 액세스 권한을 제공하지만 할당량을 보는 데 필요한 범위를 넘어서는 읽기 액세스 권한도 부여합니다. Azure OpenAI Studio를 통한 모델 배포도 이 역할의 존재에 부분적으로 종속됩니다.

이 역할은 그 자체로 작은 값을 제공하며 일반적으로 이전에 설명한 역할 중 하나 이상과 함께 할당됩니다.

Cognitive Services 사용량 읽기 권한자 + Cognitive Services OpenAI 사용자

Cognitive Services OpenAI User의 모든 기능과 다음을 수행할 수 있는 기능:

- Azure OpenAI Studio에서 할당량 할당 보기

Cognitive Services 사용량 읽기 권한자 + Cognitive Services OpenAI 기여자

Cognitive Services OpenAI 기여자의 모든 기능과 다음을 수행할 수 있는 기능

- Azure OpenAI Studio에서 할당량 할당 보기

Cognitive Services 사용량 읽기 권한자 + Cognitive Services 기여자

Cognitive Services 기여자의 모든 기능과 다음을 수행할 수 있는 기능

- Azure OpenAI Studio에서 보기 및 할당량 할당 편집
- 새 모델 배포 생성 또는 기존 모델 배포 수정(Azure OpenAI Studio를 통해)

요약

사용 권한	Cognitive Services OpenAI 사용자	Cognitive Services OpenAI 기여자	Cognitive Services 기여자	Cognitive Services 사용량 읽기 권한자
Azure portal에서 리소스 보기	✓	✓	✓	—
키 및 엔드포인트에서 리소스 엔드포인트를 봅니다.	✓	✓	✓	—
Azure OpenAI Studio에서 리소스 및 관련 모델 배포 보기	✓	✓	✓	—
Azure OpenAI Studio에서 배포에 사용할 수 있는 모델 보기	✓	✓	✓	—
이 Azure OpenAI 리소스에 이미 배포된 모델에 대해 채팅, 완료 및 DALL-E(미리 보기) 플레이그라운드 환경을 사용합니다.	✓	✓	✓	—
모델 배포 만들기 또는 편집	✗	✓	✓	—
사용자 지정 미세 조정 모델 만들기 또는 배포	✗	✓	✓	—
미세 조정을 위한 데이터 세트 업로드	✗	✓	✓	—
새로운 Azure OpenAI 리소스 만들기	✗	✗	✓	—
키 및 엔드포인트에서 키 보기/복사/다시 생성	✗	✗	✓	—
사용자 지정 콘텐츠 필터 만들기	✗	✗	✓	—
"사용자 데이터" 기능을 위한 데이터 원본 추가	✗	✗	✓	—
액세스 할당량	✗	✗	✗	✓
Microsoft Entra ID를 사용하여 유추 API 호출	✓	✓	✗	—

일반적인 문제

Azure OpenAI Studio에서 Azure Cognitive Search 옵션을 볼 수 없음

문제:

기존 Azure Cognitive Search 리소스를 선택하면 검색 인덱스가 로드되지 않고 로드 훈이 계속 회전합니다. Azure OpenAI Studio에서 도우미 설정 아래의 **플레이그라운드 채팅> 데이터 추가(미리 보기)**로 이동합니다. **데이터 원본 추가**를 선택하면 Azure Cognitive Search 또는 Blob Storage를 통해 데이터 원본을 추가할 수 있는 모달이 열립니다. Azure Cognitive Search 옵션과 기존 Azure Cognitive Search 리소스를 선택하면 선택할 수 있는 Azure Cognitive Search 인덱스가 로드됩니다.

근본 원인

Azure Cognitive Search 서비스를 나열하기 위한 일반 API 호출을 수행하려면 다음 호출이 수행됩니다.

```
https://management.azure.com/subscriptions/{subscriptionId}/providers/Microsoft.Search/searchServices?api-version=2021-04-01-Preview
```

{subscriptionId}를 실제 구독 ID로 바꾸세요.

이 API 호출을 위해서는 **구독 수준 범위** 역할이 필요합니다. 읽기 전용 액세스에는 **읽기 권한자** 역할을 사용하고 읽기-쓰기 액세스에는 **기여자** 역할을 사용할 수 있습니다. Azure Cognitive Search 서비스에만 액세스해야 하는 경우에는 **Azure Cognitive Search 서비스 기여자** 또는 **Azure Cognitive Search 서비스 읽기 권한자** 역할을 사용할 수 있습니다.

솔루션 옵션

- 구독 관리자 또는 소유자에게 문의: Azure 구독을 관리하는 사람에게 연락하여 적절한 액세스를 요청합니다. 요구 사항 및 필요한 특정 역할(예: 읽기 권한자, 기여자, Azure Cognitive Search Service 기여자 또는 Azure Cognitive Search Service 읽기 권한자)을 설명합니다.
- 구독 수준 또는 리소스 그룹 수준 액세스 요청: 특정 리소스에 액세스해야 하는 경우 구독 소유자에게 적절한 수준(구독 또는 리소스 그룹)에서 액세스 권한을 부여하도록 요청합니다. 이렇게 하면 관련 없는 리소스에 액세스하지 않고도 필요한 작업을 수행할 수 있습니다.
- Azure Cognitive Search API 키 사용: Azure Cognitive Search Search와만 상호 작용해야 하는 경우 구독 소유자로부터 관리 키 또는 쿼리 키를 요청할 수 있습니다. 이러한 키를 사용하면 Azure RBAC 역할 없이 검색 서비스에 직접 API를 호출할 수 있습니다. API 키를 사용하면 Azure RBAC 액세스 제어를 우회하므로 신중하게 사용하고 보안 모범 사례를 따르세요.

데이터에 대한 Azure OpenAI Studio에서 파일을 업로드할 수 없음

증상: Azure OpenAI Studio를 사용하여 데이터 저장 기능을 위한 스토리지에 액세스할 수 없습니다.

근본 원인:

Azure OpenAI Studio에서 Blob Storage에 액세스하려는 사용자의 구독 수준 액세스가 부족합니다. 사용자에게 Azure Management API 엔드포인트를 호출하는 데 필요한 권한이 없을 수도 있습니다.

```
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Storage/storageAccounts/{accountName}/listAccountSas?api-version=2022-09-01
```

보안상의 이유로 Azure 구독 소유자가 Blob 스토리지에 대한 공용 액세스를 사용하지 않도록 설정합니다.

API 호출에 필요한 권한:

`**Microsoft.Storage/storageAccounts/listAccountSas/action:**` 이 권한을 통해 사용자는 지정된 스토리지 계정에 대한 SAS(공유 액세스 서명) 토큰을 나열할 수 있습니다.

사용자에게 권한이 없을 수 있는 이유는 다음과 같습니다.

- 사용자에게 API 호출에 필요한 권한을 포함하지 않는 제한된 역할이 Azure 구독에 할당됩니다.
- 보안 문제 또는 조직 정책으로 인해 구독 소유자 또는 관리자가 사용자의 역할을 제한했습니다.
- 사용자의 역할이 최근에 변경되었으며 새 역할에 필요한 권한이 부여되지 않았습니다.

솔루션 옵션

- 액세스 권한 확인 및 업데이트:** 사용자에게 API 호출에 필요한 권한 (`Microsoft.Storage/storageAccounts/listAccountSas/action`)을 포함하여 적절한 구독 수준 액세스 권한이 있는지 확인합니다. 필요한 경우 구독 소유자 또는 관리자에게 필요한 액세스 권한을 부여하도록 요청합니다.
- 소유자 또는 관리자의 지원 요청:** 위의 솔루션이 불가능한 경우 구독 소유자 또는 관리자에게 사용자 대신 데이터 파일을 업로드하도록 요청하는 것이 좋습니다. 이 접근 방식을 사용하면 사용자가 구독 수준 액세스 또는 Blob 스토리지에 대한 공개 액세스를 요구하지 않고도 Azure OpenAI Studio로 데이터를 가져올 수 있습니다.

다음 단계

- Azure RBAC(Azure 역할 기반 액세스 제어)에 대해 자세히 알아보세요.
- 또한 Azure portal을 사용하여 Azure 역할 할당도 확인해 보세요.

Azure OpenAI Service를 사용한 BCDR(비즈니스 연속성 및 재해 복구) 고려 사항

아티클 • 2023. 09. 22.

Azure OpenAI는 여러 지역에서 사용할 수 있습니다. Azure OpenAI 리소스를 만들 때 지역을 지정합니다. 그때부터 리소스와 모든 작업이 해당 Azure 서버 지역과 연결된 상태를 유지합니다.

전체 지역에 적용되는 네트워크 문제가 발생하는 것은 드물지만 불가능한 것도 아닙니다. 서비스를 항상 사용할 수 있어야 하는 경우, 다른 지역으로 장애 조치(failover)하거나 둘 이상의 지역 간에 워크로드를 분할하도록 설계해야 합니다. 두 방식은 모두 서로 다른 지역에 있는 둘 이상의 Azure OpenAI 리소스가 필요합니다. 이 문서에서는 Azure OpenAI 애플리케이션에 BCDR(비즈니스 연속성 및 재해 복구)을 구현하는 방법에 대한 일반적인 권장 사항을 제공합니다.

BCDR에 사용자 지정 코드 필요

오늘날 고객은 추론을 위해 배포 중에 제공된 엔드포인트를 호출합니다. 추론 작업은 상태 비저장이므로 지역을 사용할 수 없는 경우 데이터가 손실되지 않습니다.

지역이 작동하지 않는 경우 고객은 서비스 연속성을 보장하기 위한 조치를 취해야 합니다.

기본 모델 및 사용자 지정 모델에 대한 BCDR

기본 모델을 사용하는 경우 오류를 모니터링하도록 클라이언트 코드를 구성해야 하며, 오류가 지속되면 Azure OpenAI 구독이 있는 다른 지역을 선택하여 리디렉션할 준비를 해야 합니다.

다음 단계에 따라 오류를 모니터링하도록 클라이언트를 구성합니다.

- 모델 페이지를 사용하여 적합한 데이터 센터 및 지역을 선택합니다.
- 목록에서 기본 및 하나 이상의 보조/백업 영역을 선택합니다.
- 선택한 각 지역에 대해 Azure OpenAI 리소스를 만듭니다.
- 기본 지역 및 모든 백업 지역의 경우 코드가 다음을 알아야 합니다.
 - 리소스의 기본 URI
 - 지역 액세스 키 또는 Azure Active Directory 액세스

5. 연결 오류(일반적으로 연결 시간 초과 및 서비스 사용 불가 오류)를 모니터링하도록 코드를 구성합니다.

- 네트워크에 일시적인 오류가 발생하므로 단일 연결 문제가 발생하는 경우 다시 시도하는 것이 좋습니다.
- 연결 문제가 계속 발생하는 경우 만든 지역의 백업 리소스로 트래픽을 리디렉션합니다.

주 지역에서 모델을 미세 조정한 경우 동일한 학습 데이터를 사용하여 보조 지역의 기본 모델을 다시 학습시켜야 합니다. 그런 다음, 위 단계를 수행합니다.

자습서: Azure OpenAI Service 포함 및 문서 검색 살펴보기

아티클 • 2024. 03. 10.

이 자습서에서는 Azure OpenAI 포함 API를 사용하여 문서 검색을 수행하는 과정을 안내합니다. 여기에서 기술 자료를 쿼리하여 가장 관련성이 높은 문서를 찾습니다.

이 자습서에서는 다음을 하는 방법을 알아볼 수 있습니다.

- ✓ Azure OpenAI를 설치합니다.
- ✓ 샘플 데이터 세트를 다운로드하고 분석을 위해 준비합니다.
- ✓ 리소스 엔드포인트 및 API 키에 대한 환경 변수를 만듭니다.
- ✓ **text-embedding-ada-002(버전 2)** 모델 사용
- ✓ **코사인 유사성**을 사용하여 검색 결과의 순위를 지정합니다.

필수 조건

- Azure 구독 - [체험 구독 만들기](#)
- 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한. 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 업니다.
- **text-embedding-ada-002(버전 2)** 모델이 배포된 Azure OpenAI 리소스. 이 모델은 현재 특정 지역에서만 사용할 수 있습니다. 리소스가 없는 경우 만들기 프로세스는 [리소스 배포 가이드](#)에 설명되어 있습니다.
- [Python 3.8 이상 버전](#)
- 다음 Python 라이브러리: openai, num2words, matplotlib, plotly, scipy, scikit-learn, Pandas, tiktoken.
- [Jupyter 노트북](#)

설정

Python 라이브러리

아직 설치하지 않은 경우 다음 라이브러리를 설치해야 합니다.

OpenAI Python 1.x

콘솔

```
pip install openai num2words matplotlib plotly scipy scikit-learn pandas tiktoken
```

BillSum 데이터 세트 다운로드

BillSum은 미국 의회 및 캘리포니아 주 법안의 데이터 세트입니다. 설명을 위해 미국 청구서만 살펴보겠습니다. 코퍼스는 의회의 103-115차(1993-2018) 세션의 법안으로 구성됩니다. 데이터는 18,949개의 학습 청구서와 3,269개의 테스트 청구서로 분할되었습니다. BillSum 코퍼스는 5,000자에서 20,000자 길이의 중간 길이 입법에 중점을 둡니다. 프로젝트에 대한 자세한 정보와 이 데이터 세트가 파생된 원본 학술 논문은 [BillSum 프로젝트의 GitHub 리포지토리](#)에서 확인할 수 있습니다.

이 자습서에서는 [GitHub 샘플 데이터](#)에서 다운로드할 수 있는 `bill_sum_data.csv` 파일을 사용합니다.

로컬 컴퓨터에서 다음 명령을 실행하여 샘플 데이터를 다운로드할 수도 있습니다.

Windows 명령 프롬프트

```
curl "https://raw.githubusercontent.com/Azure-Samples/Azure-OpenAI-Docs-Samples/main/Samples/Tutorials/Embeddings/data/bill_sum_data.csv" --output bill_sum_data.csv
```

키 및 엔드포인트 검색

Azure OpenAI에 대해 성공적으로 호출하려면 **엔드포인트**와 **키**가 필요합니다.

[\[+\] 테이블 확장](#)

변수 이름

ENDPOINT 이 값은 Azure Portal에서 리소스를 검사할 때 **키 및 엔드포인트** 섹션에서 찾을 수 있습니다. 또는 Azure OpenAI Studio>플레이그라운드>코드 보기에서 값을 찾을 수 있습니다. 예제 엔드포인트는 <https://docs-test-001.openai.azure.com/>입니다.

API-KEY 이 값은 Azure Portal에서 리소스를 검사할 때 **키 및 엔드포인트** 섹션에서 찾을 수 있습니다. **KEY1** 또는 **KEY2**를 사용할 수 있습니다.

Azure Portal에서 해당 리소스로 이동합니다. **엔드포인트 및 키는 리소스 관리** 섹션에서 찾을 수 있습니다. 엔드포인트 및 액세스 키를 복사합니다. API 호출을 인증하는 데 모두

필요합니다. KEY1 또는 KEY2를 사용할 수 있습니다. 항상 두 개의 키를 사용하면 서비스 중단 없이 키를 안전하게 회전하고 다시 생성할 수 있습니다.

The screenshot shows the Azure Cognitive Service Keys and Endpoint page. On the left, there's a sidebar with various management options like Overview, Activity log, Access control (IAM), Tags, and Diagnose and solve problems. The 'Resource Management' section is expanded, showing 'Keys and Endpoint' which is highlighted with a red box. The main content area shows two key fields (KEY 1 and KEY 2) each containing several dots, a location field set to 'eastus', and an endpoint URL field containing 'https://docs-test-001.openai.azure.com/'. A blue button labeled 'Show Keys' is visible above the key fields. A note at the top right says: 'These keys are used to access your Cognitive Service API. Do not share your keys. Store them securely—for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.'

환경 변수

The screenshot shows a Jupyter Notebook interface. The top bar has a '명령줄' tab selected. Below it, there are two code cells. The first cell is titled 'CMD' and contains the command: 'setx AZURE_OPENAI_API_KEY "REPLACE_WITH_YOUR_KEY_VALUE_HERE"'. The second cell is also titled 'CMD' and contains the command: 'setx AZURE_OPENAI_ENDPOINT "REPLACE_WITH_YOUR_ENDPOINT_HERE"'. Both cells have a light gray background.

환경 변수를 설정한 후에는 환경 변수에 액세스할 수 있도록 Jupyter Notebook 또는 사용 중인 IDE를 닫고 다시 열어야 할 수 있습니다. Jupyter Notebook을 사용하는 것이 강력히 권장되지만, 어떤 이유로든 코드 블록의 끝에서 수행되는 것처럼 직접 호출 `dataframe_name` 하는 대신 사용하여 `print(dataframe_name)` pandas 데이터 프레임을 반환하는 코드를 수정할 필요가 없습니다.

기본 설정하는 Python IDE에서 다음 코드를 실행합니다.

라이브러리 가져오기

Python

```
import os
import re
import requests
import sys
from num2words import num2words
import os
import pandas as pd
import numpy as np
import tiktoken
from openai import AzureOpenAI
```

이제 csv 파일을 읽고 Pandas DataFrame을 만들어야 합니다. 초기 DataFrame이 만들어진 후 `df`를 실행하여 테이블의 콘텐츠를 볼 수 있습니다.

Python

```
df=pd.read_csv(os.path.join(os.getcwd(),'bill_sum_data.csv')) # This assumes
that you have placed the bill_sum_data.csv in the same directory you are
running Jupyter Notebooks
df
```

출력:

Unnamed: 0	bill_id	text	summary	title	text_len	sum_len
0	0	110_hr37	SECTION 1. SHORT TITLE\n\nThis Act ma...	National Science Education Tax Incentive for B...	To amend the Internal Revenue Code of 1986 to ...	8494 321
1	1	112_hr2873	SECTION 1. SHORT TITLE\n\nThis Act ma...	Small Business Expansion and Hiring Act of 201...	To amend the Internal Revenue Code of 1986 to ...	6522 1424
2	2	109_s2408	SECTION 1. RELEASE OF DOCUMENTS CAPTURED IN IR...	Requires the Director of National Intelligence...	A bill to require the Director of National Inte...	6154 463
3	3	108_s1899	SECTION 1. SHORT TITLE\n\nThis Act ma...	National Cancer Act of 2003 - Amends the Publi...	A bill to improve data collection and dissemin...	19853 1400
4	4	107_s1531	SECTION 1. SHORT TITLE\n\nThis Act ma...	Military Call-up Relief Act - Amends the Inter...	A bill to amend the Internal Revenue Code of 1...	6273 278
5	5	107_hr4541	SECTION 1. RELIQUIDATION OF CERTAIN ENTRIES PR...	Requires the Customs Service to reliquidate ce...	To provide for reliquidation of entries premat...	11691 114
6	6	111_s1495	SECTION 1. SHORT TITLE\n\nThis Act ma...	Service Dogs for Veterans Act of 2009 - Direct...	A bill to require the Secretary of Veterans Af...	5328 379
7	7	111_s3885	SECTION 1. SHORT TITLE\n\nThis Act ma...	Race to the Top Act of 2010 - Directs the Secr...	A bill to provide incentives for States and lo...	16668 1525
8	8	113_hr1796	SECTION 1. SHORT TITLE\n\nThis Act ma...	Troop Talent Act of 2013 - Directs the Secreta...	Troop Talent Act of 2013	15352 2151
9	9	103_hr1987	SECTION 1. SHORT TITLE\n\nThis Act ma...	Taxpayer's Right to View Act of 1993 - Amends ...	Taxpayer's Right to View Act of 1993	5633 894
10	10	103_hr1677	SECTION 1. SHORT TITLE\n\nThis Act ma...	Full-Service Schools Act - Establishes the Fed...	Full-Service Schools Act	12472 1107
11	11	111_s3149	SECTION 1. SHORT TITLE\n\nThis Act ma...	Wall Street Compensation Reform Act of 2010 - ...	A bill to amend the Internal Revenue Code of 1...	18226 1297
12	12	110_hr1007	SECTION 1. FINDINGS.\n\nThe Congress f...	Amends the Marine Mammal Protection Act of 197...	To amend the Marine Mammal Protection Act of 1...	5261 276
13	13	113_hr3137	SECTION 1. SHORT TITLE\n\nThis Act ma...	Freedom and Mobility in Consumer Banking Act -...	Freedom and Mobility in Consumer Banking Act	17690 2044
14	14	115_hr1634	SECTION 1. SHORT TITLE\n\nThis Act ma...	Education and Training for Health Act of 2017 ...	Education and Training for Health Act of 2017	9037 772
15	15	103_hr1815	SECTION 1. SHORT TITLE\n\nThis Act ma...	Recreational Hunting Safety and Preservation A...	Recreational Hunting Safety and Preservation A...	13024 475
16	16	113_s1773	SECTION 1. SHORT TITLE\n\nThis Act ma...	Andrew Prior Act or Andrew's Law - Amends the ...	Andrew's Law	5149 613
17	17	106_hr5585	SECTION 1. SHORT TITLE\n\nThis Act ma...	Directs the President, in coordination with de...	Energy Independence Act of 2000	8007 810
18	18	114_hr2499	SECTION 1. SHORT TITLE\n\nThis Act ma...	This measure has not been amended since it was...	Veterans Entrepreneurship Act of 2015	7539 1421
19	19	111_hr3141	SECTION 1. SHORT TITLE\n\nThis Act ma...	Strengthening the Health Care Safety Net Act o...	To amend title XIX of the Social Security Act ...	18429 514

초기 테이블에는 필요한 것보다 더 많은 열이 있습니다. `text`, `summary` 및 `title`에 대한 열만 포함하는 `df_bills`라는 더 작은 새 DataFrame을 만듭니다.

Python

```
df_bills = df[['text', 'summary', 'title']]  
df_bills
```

출력:

	text	summary	title
0	SECTION 1. SHORT TITLE\n\n This Act may be...	National Science Education Tax Incentive for B...	To amend the Internal Revenue Code of 1986 to ...
1	SECTION 1. SHORT TITLE\n\n This Act may be...	Small Business Expansion and Hiring Act of 201...	To amend the Internal Revenue Code of 1986 to ...
2	SECTION 1. RELEASE OF DOCUMENTS CAPTURED IN IR...	Requires the Director of National Intelligence...	A bill to require the Director of National Int...
3	SECTION 1. SHORT TITLE\n\n This Act may be...	National Cancer Act of 2003 - Amends the Publi...	A bill to improve data collection and dissemin...
4	SECTION 1. SHORT TITLE\n\n This Act may be...	Military Call-up Relief Act - Amends the Inter...	A bill to amend the Internal Revenue Code of 1...
5	SECTION 1. RELIQUIDATION OF CERTAIN ENTRIES PR...	Requires the Customs Service to reliquidate ce...	To provide for reliquidation of entries premat...
6	SECTION 1. SHORT TITLE\n\n This Act may be...	Service Dogs for Veterans Act of 2009 - Direct...	A bill to require the Secretary of Veterans Af...
7	SECTION 1. SHORT TITLE\n\n This Act may be...	Race to the Top Act of 2010 - Directs the Secr...	A bill to provide incentives for States and lo...
8	SECTION 1. SHORT TITLE\n\n This Act may be...	Troop Talent Act of 2013 - Directs the Secreta...	Troop Talent Act of 2013
9	SECTION 1. SHORT TITLE\n\n This Act may be...	Taxpayer's Right to View Act of 1993 - Amends ...	Taxpayer's Right to View Act of 1993
10	SECTION 1. SHORT TITLE\n\n This Act may be...	Full-Service Schools Act - Establishes the Fed...	Full-Service Schools Act
11	SECTION 1. SHORT TITLE\n\n This Act may be...	Wall Street Compensation Reform Act of 2010 - ...	A bill to amend the Internal Revenue Code of 1...
12	SECTION 1. FINDINGS.\n\n The Congress finds...	Amends the Marine Mammal Protection Act of 197...	To amend the Marine Mammal Protection Act of 1...
13	SECTION 1. SHORT TITLE\n\n This Act may be...	Freedom and Mobility in Consumer Banking Act -...	Freedom and Mobility in Consumer Banking Act
14	SECTION 1. SHORT TITLE\n\n This Act may be...	Education and Training for Health Act of 2017 ...	Education and Training for Health Act of 2017
15	SECTION 1. SHORT TITLE\n\n This Act may be...	Recreational Hunting Safety and Preservation A...	Recreational Hunting Safety and Preservation A...
16	SECTION 1. SHORT TITLE\n\n This Act may be...	Andrew Prior Act or Andrew's Law - Amends the ...	Andrew's Law
17	SECTION 1. SHORT TITLE\n\n This Act may be...	Directs the President, in coordination with de...	Energy Independence Act of 2000
18	SECTION 1. SHORT TITLE\n\n This Act may be c...	This measure has not been amended since it was...	Veterans Entrepreneurship Act of 2015
19	SECTION 1. SHORT TITLE\n\n This Act may be...	Strengthening the Health Care Safety Net Act o...	To amend title XIX of the Social Security Act ...

다음으로 불필요한 공백을 제거하고 문장 부호를 정리하여 토큰화를 위한 데이터를 준비하여 간단한 데이터 정리를 수행합니다.

Python

```
pd.options.mode.chained_assignment = None #https://pandas.pydata.org/pandas-  
docs/stable/user_guide/indexing.html#evaluation-order-matters  
  
# s is input text  
def normalize_text(s, sep_token = " \n "):  
    s = re.sub(r'\s+', ' ', s).strip()  
    s = re.sub(r". ,","",s)  
    # remove all instances of multiple spaces  
    s = s.replace(..,..)  
    s = s.replace(.. ..,..)  
    s = s.replace("\n", "")  
    s = s.strip()  
  
    return s  
  
df_bills['text']= df_bills["text"].apply(lambda x : normalize_text(x))
```

이제 토큰 제한(8192 토큰)에 비해 너무 긴 청구서를 제거해야 합니다.

Python

```
tokenizer = tiktoken.get_encoding("cl100k_base")
df_bills['n_tokens'] = df_bills["text"].apply(lambda x:
len(tokenizer.encode(x)))
df_bills = df_bills[df_bills.n_tokens<8192]
len(df_bills)
```

출력

20

① 참고

이 경우 모든 청구서는 포함 모델 입력 토큰 한도에 속하지만 위의 기술을 사용하여 포함 실패를 유발할 수 있는 항목을 제거할 수 있습니다. 포함 제한을 초과하는 콘텐츠에 직면하면 콘텐츠를 더 작은 조각으로 청크한 다음 한 번에 하나씩 포함할 수 있습니다.

다시 한 번 `df_bills`를 검토합니다.

Python

```
df_bills
```

출력:

	text	summary	title	n_tokens
0	SECTION 1. SHORT TITLE. This Act may be cited ...	National Science Education Tax Incentive for B...	To amend the Internal Revenue Code of 1986 to ...	1466
1	SECTION 1. SHORT TITLE. This Act may be cited ...	Small Business Expansion and Hiring Act of 201...	To amend the Internal Revenue Code of 1986 to ...	1183
2	SECTION 1. RELEASE OF DOCUMENTS CAPTURED IN IR...	Requires the Director of National Intelligence...	A bill to require the Director of National Int...	937
3	SECTION 1. SHORT TITLE. This Act may be cited ...	National Cancer Act of 2003 - Amends the Publi...	A bill to improve data collection and dissemin...	3670
4	SECTION 1. SHORT TITLE. This Act may be cited ...	Military Call-up Relief Act - Amends the Inter...	A bill to amend the Internal Revenue Code of 1...	1038
5	SECTION 1. RELIQUIDATION OF CERTAIN ENTRIES PR...	Requires the Customs Service to reliquidate ce...	To provide for reliquidation of entries premat...	2026
6	SECTION 1. SHORT TITLE. This Act may be cited ...	Service Dogs for Veterans Act of 2009 - Direct...	A bill to require the Secretary of Veterans Af...	880
7	SECTION 1. SHORT TITLE. This Act may be cited ...	Race to the Top Act of 2010 - Directs the Secr...	A bill to provide incentives for States and lo...	2815
8	SECTION 1. SHORT TITLE. This Act may be cited ...	Troop Talent Act of 2013 - Directs the Secreta...	Troop Talent Act of 2013	2479
9	SECTION 1. SHORT TITLE. This Act may be cited ...	Taxpayer's Right to View Act of 1993 - Amends ...	Taxpayer's Right to View Act of 1993	947
10	SECTION 1. SHORT TITLE. This Act may be cited ...	Full-Service Schools Act - Establishes the Fed...	Full-Service Schools Act	2164
11	SECTION 1. SHORT TITLE. This Act may be cited ...	Wall Street Compensation Reform Act of 2010 - ...	A bill to amend the Internal Revenue Code of 1...	2331
12	SECTION 1. FINDINGS. The Congress finds the fo...	Amends the Marine Mammal Protection Act of 197...	To amend the Marine Mammal Protection Act of 1...	1192
13	SECTION 1. SHORT TITLE. This Act may be cited ...	Freedom and Mobility in Consumer Banking Act -...	Freedom and Mobility in Consumer Banking Act	2402
14	SECTION 1. SHORT TITLE. This Act may be cited ...	Education and Training for Health Act of 2017 ...	Education and Training for Health Act of 2017	1648
15	SECTION 1. SHORT TITLE. This Act may be cited ...	Recreational Hunting Safety and Preservation A...	Recreational Hunting Safety and Preservation A...	2209
16	SECTION 1. SHORT TITLE. This Act may be cited ...	Andrew Prior Act or Andrew's Law - Amends the ...	Andrew's Law	608
17	SECTION 1. SHORT TITLE. This Act may be cited ...	Directs the President, in coordination with de...	Energy Independence Act of 2000	1352
18	SECTION 1. SHORT TITLE. This Act may be cited ...	This measure has not been amended since it was...	Veterans Entrepreneurship Act of 2015	1393
19	SECTION 1. SHORT TITLE. This Act may be cited ...	Strengthening the Health Care Safety Net Act o...	To amend title XIX of the Social Security Act ...	2678

n_tokens 열과 텍스트가 궁극적으로 토큰화되는 방식을 조금 더 이해하려면 다음 코드를 실행하는 것이 도움이 될 수 있습니다.

Python

```
sample_encode = tokenizer.encode(df_bills.text[0])
decode = tokenizer.decode_tokens_bytes(sample_encode)
decode
```

문서의 경우 의도적으로 출력을 자르지만 환경에서 이 명령을 실행하면 청크로 토큰화된 인덱스 0의 전체 텍스트가 반환됩니다. 어떤 경우에는 전체 단어가 단일 토큰으로 표시되는 반면 다른 경우에는 단어의 일부가 여러 토큰으로 분할되는 것을 볼 수 있습니다.

출력

```
[b'SECTION',
 b' ',
 b'1',
 b'.',
 b' SHORT',
 b' TITLE',
 b'.',
 b' This',
 b' Act',
 b' may',
 b' be',
 b' cited',
 b' as',
 b' the',
 b' `',
 b'National',
```

```
b' Science',
b' Education',
b' Tax',
b' In',
b'cent',
b'ive',
b' for',
b' Businesses',
b' Act',
b' of',
b' ',
b'200',
b'7',
b'''."',
b' SEC',
b'.',
b' ',
b'2',
b'.',
b' C',
b'RED',
b'ITS',
b' FOR',
b' CERT',
b'AIN',
b' CONTRIBUT',
b'IONS',
b' BEN',
b'EF',
b'IT',
b'ING',
b' SC',
```

그런 다음 `decode` 변수의 길이를 확인하면 `n_tokens` 열의 첫 번째 숫자와 일치함을 알 수 있습니다.

Python

```
len(decode)
```

출력

```
1466
```

이제 토큰화가 작동하는 방식에 대해 더 많이 이해했으므로 포함으로 넘어갈 수 있습니다. 문서를 실제로 토큰화하지 않았다는 점에 유의해야 합니다. `n_tokens` 열은 단순히 토큰화 및 포함을 위해 모델에 전달하는 데이터가 입력 토큰 제한인 8,192를 초과하지 않도록 하는 방법입니다. 포함 모델에 문서를 전달하면 문서를 위의 예와 유사한 토큰(반드시 동일하지는 않음)으로 나눈 다음 토큰을 벡터 검색을 통해 액세스할 수 있는 일련의 부동

소수점 숫자로 변환합니다. 이러한 임베딩은 로컬로 저장하거나 Azure 데이터베이스에 저장하여 벡터 검색을 지원할 수 있습니다. 결과적으로 각 청구서에는 DataFrame의 오른쪽에 있는 새 `ada_v2` 열에 해당하는 자체 포함 벡터가 포함됩니다.

아래 예제에서는 포함하려는 모든 항목당 한 번씩 포함 모델을 호출합니다. 큰 포함 프로젝트로 작업할 때 한 번에 하나의 입력이 아닌 포함할 입력의 배열을 모델에 전달할 수도 있습니다. 모델에 입력의 배열을 전달하면 포함 엔드포인트에 대한 호출당 최대 입력 항목 수는 2048입니다.

OpenAI Python 1.x

Python

```
client = AzureOpenAI(  
    api_key = os.getenv("AZURE_OPENAI_API_KEY"),  
    api_version = "2023-05-15",  
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")  
)  
  
def generate_embeddings(text, model="text-embedding-ada-002"): # model =  
    "deployment_name"  
    return client.embeddings.create(input = [text],  
model=model).data[0].embedding  
  
df_bills['ada_v2'] = df_bills["text"].apply(lambda x :  
generate_embeddings (x, model = 'text-embedding-ada-002')) # model  
should be set to the deployment name you chose when you deployed the  
text-embedding-ada-002 (Version 2) model
```

Python

```
df_bills
```

출력:

	text	summary	title	n_tokens	ada_v2
0	SECTION 1. SHORT TITLE. This Act may be cited ...	National Science Education Tax Incentive for E...	To amend the Internal Revenue Code of 1986 to ...	1466	[0.01333628874272108, -0.02151912823319435, 0...
1	SECTION 1. SHORT TITLE. This Act may be cited ...	Small Business Expansion and Hiring Act of 201...	To amend the Internal Revenue Code of 1986 to ...	1183	[0.005016345530748367, -0.00569863710552454, 0...
2	SECTION 1. RELEASE OF DOCUMENTS CAPTURED IN IR...	Requires the Director of National Intelligence...	A bill to require the Director of National Inte...	937	[0.012699966318905354, -0.0189779107093811, 0...
3	SECTION 1. SHORT TITLE. This Act may be cited ...	National Cancer Act of 2003 - Amends the Publi...	A bill to improve data collection and dissemin...	3670	[0.004736857954412699, -0.026448562741279602, 0...
4	SECTION 1. SHORT TITLE. This Act may be cited ...	Military Call-up Relief Act - Amends the Publi...	A bill to amend the Internal Revenue Code of 1...	1038	[0.010082815773785114, -0.0007545037078671157, 0...
5	SECTION 1. RELIQUIDATION OF CERTAIN ENTRIES PR...	Requires the Customs Service to reliquidate ce...	To provide for reliquidation of entries premat...	2026	[0.012738252058625221, 0.004982588812708855, 0...
6	SECTION 1. SHORT TITLE. This Act may be cited ...	Service Dogs for Veterans Act of 2009 - Direct...	A bill to require the Secretary of Veterans Af...	880	[0.005205095745623112, -0.016558492556214333, 0...
7	SECTION 1. SHORT TITLE. This Act may be cited ...	Race to the Top Act of 2010 - Directs the Secre...	A bill to provide incentives for States and lo...	2815	[0.0245393868853575706, -0.016805868595838547, 0...
8	SECTION 1. SHORT TITLE. This Act may be cited ...	Troop Talent Act of 2013 - Directs the Secreta...	Troop Talent Act of 2013	2479	[+0.005527574568986893, -0.014311426319181919, 0...
9	SECTION 1. SHORT TITLE. This Act may be cited ...	Taxpayer's Right to View Act of 1993 - Amends ...	Taxpayer's Right to View Act of 1993	947	[0.004519130103290081, -0.023599395528435707, 0...
10	SECTION 1. SHORT TITLE. This Act may be cited ...	Full-Service Schools Act - Establishes the Fed...	Full-Service Schools Act	2164	[0.0075974976643919945, -0.006962535437196493, 0...
11	SECTION 1. SHORT TITLE. This Act may be cited ...	Wall Street Compensation Reform Act of 2010 - ...	A bill to amend the Internal Revenue Code of 1...	2331	[0.014871294610202312, -0.001929433667100966, 0...
12	SECTION 1. FINDINGS. The Congress finds the fo...	Amends the Marine Mammal Protection Act of 197...	To amend the Marine Mammal Protection Act of 1...	1192	[0.04441450908780098, 0.02687789686024189, 0...
13	SECTION 1. SHORT TITLE. This Act may be cited ...	Freedom and Mobility in Consumer Banking Act -...	Freedom and Mobility in Consumer Banking Act	2402	[0.021314678713679314, -0.008310768753290176, 0...
14	SECTION 1. SHORT TITLE. This Act may be cited ...	Education and Training for Health Act of 2017 ...	Education and Training for Health Act of 2017	1648	[+0.009376125410199165, -0.0360078439116478, 0...
15	SECTION 1. SHORT TITLE. This Act may be cited ...	Recreational Hunting Safety and Preservation A...	Recreational Hunting Safety and Preservation A...	2209	[0.024976342916488647, -0.005445675924420357, 0...
16	SECTION 1. SHORT TITLE. This Act may be cited ...	Andrew Prior Act or Andrew's Law - Amends the ...	Andrew's Law	608	[0.029043208807706833, -0.01100732292799557, 0...
17	SECTION 1. SHORT TITLE. This Act may be cited ...	Directs the President, in coordination with de...	Energy Independence Act of 2000	1352	[+0.0034495051950216293, -0.02827893753500133...
18	SECTION 1. SHORT TITLE. This Act may be cited ...	This measure has not been amended since it was...	Veterans Entrepreneurship Act of 2015	1393	[+0.0026434329338371754, -0.00496460217982306...
19	SECTION 1. SHORT TITLE. This Act may be cited ...	Strengthening the Health Care Safety Net Act o...	To amend title XIX of the Social Security Act ...	2678	[0.009399736300110817, -0.02588636800646782, 0...

아래의 검색 코드 블록을 실행할 때 동일한 *text-embedding-ada-002*(버전 2) 모델과 함께 "케이블 회사 세금 수익에 대한 정보를 얻을 수 있나요?" 검색 쿼리를 포함합니다. 다음으로 [코사인 유사성](#)으로 순위가 매겨진 쿼리에서 새로 포함된 텍스트에 삽입된 가장 가까운 청구서를 찾습니다.

```
OpenAI Python 1.x

Python

def cosine_similarity(a, b):
    return np.dot(a, b) / (np.linalg.norm(a) * np.linalg.norm(b))

def get_embedding(text, model="text-embedding-ada-002"): # model =
    "deployment_name"
    return client.embeddings.create(input = [text],
model=model).data[0].embedding

def search_docs(df, user_query, top_n=4, to_print=True):
    embedding = get_embedding(
        user_query,
        model="text-embedding-ada-002" # model should be set to the
        deployment name you chose when you deployed the text-embedding-ada-002
        (Version 2) model
    )
    df["similarities"] = df.ada_v2.apply(lambda x: cosine_similarity(x,
embedding))

    res = (
        df.sort_values("similarities", ascending=False)
        .head(top_n)
    )
    if to_print:
        display(res)
    return res
```

```
res = search_docs(df_bills, "Can I get information on cable company tax revenue?", top_n=4)
```

출력:

text	summary	title	n_tokens	ada_v2	similarities
9 SECTION 1. SHORT TITLE. This Act may be cited ... Taxpayer's Right to View Act of 1993 - Amends ...	Taxpayer's Right to View Act of 1993	947	[0.004519130103290081, -0.023599395528435707, ...]	0.767584	
11 SECTION 1. SHORT TITLE. This Act may be cited ... Wall Street Compensation Reform Act of 2010 - ... A bill to amend the Internal Revenue Code of 1...	A bill to amend the Internal Revenue Code of 1...	2331	[0.014871294610202312, -0.001929433667100966, ...]	0.714282	
1 SECTION 1. SHORT TITLE. This Act may be cited ... Small Business Expansion and Hiring Act of 201... To amend the Internal Revenue Code of 1986 to ...	To amend the Internal Revenue Code of 1986 to ...	1183	[0.005016345530748367, -0.00569863710552454, 0...	0.702599	
4 SECTION 1. SHORT TITLE. This Act may be cited ... Military Call-up Relief Act - Amends the Inter...	A bill to amend the Internal Revenue Code of 1...	1038	[0.010082815773785114, -0.0007545037078671157, ...]	0.699490	

마지막으로 전체 기술 자료에 대한 사용자 쿼리를 기반으로 문서 검색의 최상위 결과를 표시합니다. 이는 "1993년 납세자의 조회권법"의 최상위 결과를 반환합니다. 이 문서는 쿼리와 문서 간의 코사인 유사성 점수가 0.76입니다.

Python

```
res["summary"][9]
```

출력

"Taxpayer's Right to View Act of 1993 - Amends the Communications Act of 1934 to prohibit a cable operator from assessing separate charges for any video programming of a sporting, theatrical, or other entertainment event if that event is performed at a facility constructed, renovated, or maintained with tax revenues or by an organization that receives public financial support. Authorizes the Federal Communications Commission and local franchising authorities to make determinations concerning the applicability of such prohibition. Sets forth conditions under which a facility is considered to have been constructed, maintained, or renovated with tax revenues. Considers events performed by nonprofit or public organizations that receive tax subsidies to be subject to this Act if the event is sponsored by, or includes the participation of a team that is part of, a tax exempt organization."

이 방식을 사용하면 기술 자료의 문서 전체에서 포함을 검색 메커니즘으로 사용할 수 있습니다. 그런 다음 사용자는 상위 검색 결과를 가져와 다운스트림 작업에 사용할 수 있으며 이로 인해 초기 쿼리가 표시됩니다.

리소스 정리

이 자습서를 완료하기 위해서 OpenAI 리소스만 만들었고 OpenAI 리소스를 정리하고 제거하려는 경우 배포된 모델을 삭제한 다음 테스트 리소스 전용인 경우 리소스 또는 연결된 리소스 그룹을 삭제해야 합니다. 리소스 그룹을 삭제하면 해당 리소스 그룹에 연결된 다른 모든 리소스가 함께 삭제됩니다.

- 포털
- Azure CLI

다음 단계

Azure OpenAI의 모델에 대해 자세히 알아봅니다.

Azure OpenAI Service 모델

- 선택한 Azure 서비스를 사용하여 포함을 저장하고 벡터(유사성) 검색을 수행합니다.
 - [Azure AI 검색](#)
 - [Azure Cosmos DB for MongoDB vCore](#)
 - [Azure SQL Database](#)
 - [Azure Cosmos DB for NoSQL](#)
 - [Azure Cosmos DB for PostgreSQL](#)
 - [Azure Cache for Redis](#)

Azure OpenAI GPT-3.5 Turbo 미세 조정 자습서

아티클 • 2024. 03. 22.

이 자습서에서는 `gpt-35-turbo-0613` 모델을 미세 조정하는 과정을 안내합니다.

이 자습서에서는 다음을 하는 방법을 알아볼 수 있습니다.

- ✓ 샘플 미세 조정 데이터 세트를 만듭니다.
- ✓ 리소스 엔드포인트 및 API 키에 대한 환경 변수를 만듭니다.
- ✓ 미세 조정을 위해 샘플 학습 및 유효성 검사 데이터 세트를 준비합니다.
- ✓ 미세 조정을 위해 학습 파일 및 유효성 검사 파일을 업로드합니다.
- ✓ `gpt-35-turbo-0613`에 대한 미세 조정 작업을 만듭니다.
- ✓ 사용자 지정 미세 조정된 모델을 배포합니다.

필수 조건

- Azure 구독 – [체험 구독을 만듭니다](#).
- 원하는 Azure 구독에서 Azure OpenAI에 부여된 액세스 권한 현재 이 서비스에 대한 액세스 권한은 애플리케이션에 의해서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다.
- Python 3.8 이상 버전
- 다음 Python 라이브러리: `json`, `requests`, `os`, `tiktoken`, `time`, `openai`.
- OpenAI Python 라이브러리는 버전 `0.28.1` 이상이어야 합니다.
- [Jupyter 노트북](#)
- `gpt-35-turbo-0613` 미세 조정을 사용할 수 있는 [지역의 Azure OpenAI 리소스입니다](#). 리소스가 없는 경우 리소스 만들기 프로세스는 리소스 [배포 가이드](#)에 설명되어 있습니다.
- 액세스를 미세 조정하려면 [Cognitive Services OpenAI 기여자가 필요합니다](#).
- Azure OpenAI Studio에서 할당량을 보고 모델을 배포할 수 있는 액세스 권한이 아직 없는 경우 [추가 권한](#)이 필요합니다.

ⓘ 중요

이 자습서를 시작하기 전에 미세 조정할 수 있도록 [가격 정보](#)를 검토하여 관련 비용에 익숙해지는 것이 좋습니다. 테스트에서 이 자습서 수행 결과 미세 조정 유추와 관련된 비용 및 미세 조정된 모델을 배포하는 데 드는 시간당 호스팅 비용 외에도 1

시간의 학습 요금이 청구되었습니다. 자습서를 완료한 후에는 미세 조정된 모델 배포를 삭제해야 합니다. 그렇지 않으면 시간당 호스팅 비용이 계속 발생합니다.

설정

Python 라이브러리

OpenAI Python 1.x

Windows 명령 프롬프트

```
pip install openai requests tiktoken
```

키 및 엔드포인트 검색

Azure OpenAI에 대해 성공적으로 호출하려면 **엔드포인트**와 **키**가 필요합니다.

[+] 테이블 확장

변수 이름
값

ENDPOINT 이 값은 Azure Portal에서 리소스를 검사할 때 **키 및 엔드포인트** 섹션에서 찾을 수 있습니다. 또는 Azure OpenAI Studio>플레이그라운드>코드 보기에서 값을 찾을 수 있습니다. 예제 엔드포인트는 <https://docs-test-001.openai.azure.com/>입니다.

API-KEY 이 값은 Azure Portal에서 리소스를 검사할 때 **키 및 엔드포인트** 섹션에서 찾을 수 있습니다. **KEY1** 또는 **KEY2**를 사용할 수 있습니다.

Azure Portal에서 해당 리소스로 이동합니다. **엔드포인트 및 키는 리소스 관리** 섹션에서 찾을 수 있습니다. 엔드포인트 및 액세스 키를 복사합니다. API 호출을 인증하는 데 모두 필요합니다. **KEY1** 또는 **KEY2**를 사용할 수 있습니다. 항상 두 개의 키를 사용하면 서비스 중단 없이 키를 안전하게 회전하고 다시 생성할 수 있습니다.

Home >

docs-test-001 | Keys and Endpoint

Cognitive Service | Directory: Microsoft

Search (Ctrl+ /) Regenerate Key1 Regenerate Key2

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Resource Management

Keys and Endpoint (highlighted)

Deployments Pricing tier Networking Identity Cost analysis Properties Locks

These keys are used to access your Cognitive Service API. Do not share your keys. Store them securely—for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

Show Keys

KEY 1 (redacted)

KEY 2 (redacted)

Location/Region eastus

Endpoint https:// docs-test-001.openai.azure.com/ (redacted)

환경 변수

명령줄

CMD

```
setx AZURE_OPENAI_API_KEY "REPLACE_WITH_YOUR_KEY_VALUE_HERE"
```

CMD

```
setx AZURE_OPENAI_ENDPOINT "REPLACE_WITH_YOUR_ENDPOINT_HERE"
```

샘플 데이터 세트 만들기

미세 조정 gpt-35-turbo-0613을 수행하려면 특별히 형식이 지정된 JSON 학습 파일이 필요합니다. OpenAI는 설명서에 다음 예제를 제공합니다.

JSON

```
{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What's the capital of France?"}, {"role": "assistant", "content": "Paris, as if everyone doesn't know that already."}]}  
{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "Who wrote 'Romeo and Juliet'?"}, {"role": "assistant", "content": "Oh, just some guy named"}]
```

```
William Shakespeare. Ever heard of him?"}]}  
{"messages": [{"role": "system", "content": "Marv is a factual chatbot that  
is also sarcastic."}, {"role": "user", "content": "How far is the Moon from  
Earth?"}, {"role": "assistant", "content": "Around 384,400 kilometers. Give  
or take a few, like that really matters."}]}
```

이 예제에서는 다음과 같이 변경하여 약간 수정할 것입니다.

JSON

```
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "What's the capital  
of France?"}, {"role": "assistant", "content": "Paris, as if everyone  
doesn't know that already."}]}  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "Who wrote 'Romeo and  
Juliet'?"}, {"role": "assistant", "content": "Oh, just some guy named  
William Shakespeare. Ever heard of him?"}]}  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "How far is the Moon  
from Earth?"}, {"role": "assistant", "content": "Around 384,400 kilometers.  
Give or take a few, like that really matters."}]}
```

이 세 가지 예제는 일반적인 형식을 제공하는 데 유용하지만 사용자 지정 미세 조정된 모델을 조정하여 비슷한 방식으로 응답하려면 더 많은 예제가 필요합니다. 일반적으로 50 개 이상의 고품질 예제로 시작합니다. 그러나 성공하려면 1,000개의 고품질 학습 예제가 필요할 수 있는 사용 사례를 사용할 수 있습니다.

일반적으로 데이터 세트 크기가 두 배로 증가할 때마다 모델 품질이 선형적으로 증가할 수 있습니다. 그러나 저품질 예제는 성능에 부정적인 영향을 미칠 수 있습니다. 최고 품질의 예제에 대해서만 데이터 세트를 먼저 정리하지 않고 대량의 내부 데이터에서 모델을 학습하는 경우 예상보다 훨씬 더 나쁜 성능을 내는 모델로 끝날 수 있습니다.

두 개의 파일 `training_set.jsonl`과 `validation_set.jsonl`을 만들어야 합니다.

Jupyter Notebook을 실행하는 것과 동일한 디렉터리에 파일을 만들고 다음 코드 블록의 내용을 해당 파일에 복사합니다.

`training_set.jsonl`

JSON

```
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "Who discovered  
Antarctica?"}, {"role": "assistant", "content": "Some chaps named Fabian  
Gottlieb von Bellingshausen and Mikhail Lazarev, as if they don't teach that  
in every school!"}]}  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot
```

that is also sarcastic."}, {"role": "user", "content": "What is the biggest ocean?"}, {"role": "assistant", "content": "The Pacific Ocean. It's not like it's a small pond or anything."}]]}
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What is the largest planet?"}, {"role": "assistant", "content": "It's called Jupiter, you may have heard of it...or not."}]}]
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "Who painted the Mona Lisa?"}, {"role": "assistant", "content": "Oh, some obscure artist named Leonardo da Vinci. Not that he is famous or anything."}]}]
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What's the boiling point of water?"}, {"role": "assistant", "content": "Exciting fact of the day: it's 100 degrees Celsius. Life-changing, isn't it?"}]}]
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "Who wrote 'The Odyssey'?"}, {"role": "assistant", "content": "Oh, just some poet named Homer. Nothing special, really."}]}
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What is the largest mammal?"}, {"role": "assistant", "content": "Blue Whale. Quite a big fish isn't it? Oh wait, it's not a fish!"}]}
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What's the highest mountain peak?"}, {"role": "assistant", "content": "Everest. Brace yourself, it's not a walk in the park."}]}
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "Who invented the telephone?"}, {"role": "assistant", "content": "A guy named Alexander Graham Bell. Bet you didn't guess that one!"}]}
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What is the distance from Earth to the Sun?"}, {"role": "assistant", "content": "About 93 million miles. Just a quick drive, really."}]}]

validation set.jsonl

JSON

```
that is also sarcastic."}, {"role": "user", "content": "Who composed the  
'Moonlight Sonata'?"}, {"role": "assistant", "content": "Just a little-known  
guy named Beethoven. No big deal."}]]  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "What are the colors  
of the rainbow?"}, {"role": "assistant", "content": "ROYGBIV. Red, orange,  
yellow, green, blue, indigo, violet. Not that we learned this in preschool  
or anything!"}]]  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "What is the smallest  
planet?"}, {"role": "assistant", "content": "Meet Mercury, the runt of the  
solar system family."}]]  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "Who wrote 'Pride and  
Prejudice'?"}, {"role": "assistant", "content": "Do the words Jane Austen  
ring a bell? No? Well, she wrote it."}]]  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "What's the largest  
desert?"}, {"role": "assistant", "content": "Antarctica. Surprise, surprise!  
Deserts aren't just full of sand, you know."}]]  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "What is the longest  
river?"}, {"role": "assistant", "content": "The Nile River. It's not like  
it's famous or anything."}]]  
{"messages": [{"role": "system", "content": "Clippy is a factual chatbot  
that is also sarcastic."}, {"role": "user", "content": "What's the capital  
of Germany?"}, {"role": "assistant", "content": "Berlin. Shocking news,  
right?"}]]}
```

이제 학습 및 유효성 검사 파일에 대한 몇 가지 임시 검사를 실행해야 합니다.

Python

```
import json  
  
# Load the training set  
with open('training_set.jsonl', 'r', encoding='utf-8') as f:  
    training_dataset = [json.loads(line) for line in f]  
  
# Training dataset stats  
print("Number of examples in training set:", len(training_dataset))  
print("First example in training set:")  
for message in training_dataset[0]["messages"]:  
    print(message)  
  
# Load the validation set  
with open('validation_set.jsonl', 'r', encoding='utf-8') as f:  
    validation_dataset = [json.loads(line) for line in f]  
  
# Validation dataset stats  
print("\nNumber of examples in validation set:", len(validation_dataset))  
print("First example in validation set:")
```

```
for message in validation_dataset[0]["messages"]:
    print(message)
```

출력:

출력

```
Number of examples in training set: 10
First example in training set:
{'role': 'system', 'content': 'Clippy is a factual chatbot that is also
sarcastic.'}
{'role': 'user', 'content': 'Who discovered America?'}
{'role': 'assistant', 'content': "Some chap named Christopher Columbus, as
if they don't teach that in every school!"}

Number of examples in validation set: 10
First example in validation set:
{'role': 'system', 'content': 'Clippy is a factual chatbot that is also
sarcastic.'}
{'role': 'user', 'content': "What's the capital of Australia?"}
{'role': 'assistant', 'content': "It's Canberra, not Sydney. Shocking, I
know!"}
```

이 경우 10개의 학습 및 10개의 유효성 검사 예제만 있으므로 모델을 미세 조정하는 기본 메커니즘을 보여 주지만 일관되게 눈에 띄는 영향을 줄 만큼 충분히 큰 예제가 될 가능성은 거의 없습니다.

이제 tiktoken 라이브러리를 사용하여 OpenAI에서 몇 가지 추가 코드를 실행하여 토큰 수의 유효성을 검사할 수 있습니다. 개별 예제는 `gpt-35-turbo-0613` 모델의 입력 토큰 제한인 4096개의 토큰 이하로 유지해야 합니다.

Python

```
import json
import tiktoken
import numpy as np
from collections import defaultdict

encoding = tiktoken.get_encoding("cl100k_base") # default encoding used by
gpt-4, turbo, and text-embedding-ada-002 models

def num_tokens_from_messages(messages, tokens_per_message=3,
tokens_per_name=1):
    num_tokens = 0
    for message in messages:
        num_tokens += tokens_per_message
        for key, value in message.items():
            num_tokens += len(encoding.encode(value))
            if key == "name":
                num_tokens += tokens_per_name
```

```

        num_tokens += 3
    return num_tokens

def num_assistant_tokens_from_messages(messages):
    num_tokens = 0
    for message in messages:
        if message["role"] == "assistant":
            num_tokens += len(encoding.encode(message["content"]))
    return num_tokens

def print_distribution(values, name):
    print(f"\n#### Distribution of {name}:")
    print(f"min / max: {min(values)}, {max(values)}")
    print(f"mean / median: {np.mean(values)}, {np.median(values)}")
    print(f"p5 / p95: {np.quantile(values, 0.1)}, {np.quantile(values, 0.9)}")

files = ['training_set.jsonl', 'validation_set.jsonl']

for file in files:
    print(f"Processing file: {file}")
    with open(file, 'r', encoding='utf-8') as f:
        dataset = [json.loads(line) for line in f]

    total_tokens = []
    assistant_tokens = []

    for ex in dataset:
        messages = ex.get("messages", {})
        total_tokens.append(num_tokens_from_messages(messages))

    assistant_tokens.append(num_assistant_tokens_from_messages(messages))

print_distribution(total_tokens, "total tokens")
print_distribution(assistant_tokens, "assistant tokens")
print('*' * 50)

```

출력:

출력

```

Processing file: training_set.jsonl

#### Distribution of total tokens:
min / max: 47, 62
mean / median: 52.1, 50.5
p5 / p95: 47.9, 57.5

#### Distribution of assistant tokens:
min / max: 13, 30
mean / median: 17.6, 15.5
p5 / p95: 13.0, 21.9
*****
```

```
Processing file: validation_set.jsonl
```

```
#### Distribution of total tokens:
```

```
min / max: 43, 65
```

```
mean / median: 51.4, 49.0
```

```
p5 / p95: 45.7, 56.9
```

```
#### Distribution of assistant tokens:
```

```
min / max: 8, 29
```

```
mean / median: 15.9, 13.5
```

```
p5 / p95: 11.6, 20.9
```

```
*****
```

미세 조정 파일 업로드

OpenAI Python 1.x

Python

```
# Upload fine-tuning files

import os
from openai import AzureOpenAI

client = AzureOpenAI(
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT"),
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2023-12-01-preview" # This API version or later is
required to access fine-tuning for turbo/babbage-002/davinci-002
)

training_file_name = 'training_set.jsonl'
validation_file_name = 'validation_set.jsonl'

# Upload the training and validation dataset files to Azure OpenAI with
the SDK.

training_response = client.files.create(
    file=open(training_file_name, "rb"), purpose="fine-tune"
)
training_file_id = training_response.id

validation_response = client.files.create(
    file=open(validation_file_name, "rb"), purpose="fine-tune"
)
validation_file_id = validation_response.id

print("Training file ID:", training_file_id)
print("Validation file ID:", validation_file_id)
```

출력:

출력

```
Training file ID: file-9ace76cb11f54fdd8358af27abf4a3ea
Validation file ID: file-70a3f525ed774e78a77994d7a1698c4b
```

미세 조정 시작

이제 미세 조정 파일이 성공적으로 업로드되었으므로 미세 조정 학습 작업을 제출할 수 있습니다.

OpenAI Python 1.x

Python

```
response = client.fine_tuning.jobs.create(
    training_file=training_file_id,
    validation_file=validation_file_id,
    model="gpt-35-turbo-0613", # Enter base model name. Note that in
    Azure OpenAI the model name contains dashes and cannot contain
    dot/period characters.
)

job_id = response.id

# You can use the job ID to monitor the status of the fine-tuning job.
# The fine-tuning job will take some time to start and complete.

print("Job ID:", response.id)
print("Status:", response.status)
print(response.model_dump_json(indent=2))
```

출력:

출력

```
Job ID: ftjob-40e78bc022034229a6e3a222c927651c
Status: pending
{
  "hyperparameters": {
    "n_epochs": 2
  },
  "status": "pending",
  "model": "gpt-35-turbo-0613",
  "training_file": "file-90ac5d43102f4d42a3477fd30053c758",
  "validation_file": "file-e21aad7dddbc4ddc98ba35c790a016e5",
```

```
"id": "ftjob-40e78bc022034229a6e3a222c927651c",
"created_at": 1697156464,
"updated_at": 1697156464,
"object": "fine_tuning.job"
}
```

학습 작업 상태 추적

완료될 때까지 학습 작업 상태를 폴링하려는 경우 다음을 실행할 수 있습니다.

OpenAI Python 1.x

Python

```
# Track training status

from IPython.display import clear_output
import time

start_time = time.time()

# Get the status of our fine-tuning job.
response = client.fine_tuning.jobs.retrieve(job_id)

status = response.status

# If the job isn't done yet, poll it every 10 seconds.
while status not in ["succeeded", "failed"]:
    time.sleep(10)

    response = client.fine_tuning.jobs.retrieve(job_id)
    print(response.model_dump_json(indent=2))
    print("Elapsed time: {} minutes {} seconds".format(int((time.time() - start_time) // 60), int((time.time() - start_time) % 60)))
    status = response.status
    print(f'Status: {status}')
    clear_output(wait=True)

print(f'Fine-tuning job {job_id} finished with status: {status}')

# List all fine-tuning jobs for this resource.
print('Checking other fine-tune jobs for this resource.')
response = client.fine_tuning.jobs.list()
print(f'Found {len(response.data)} fine-tune jobs.')
```

출력:

ouput

```
{  
    "hyperparameters": {  
        "n_epochs": 2  
    },  
    "status": "running",  
    "model": "gpt-35-turbo-0613",  
    "training_file": "file-9ace76cb11f54fdd8358af27abf4a3ea",  
    "validation_file": "file-70a3f525ed774e78a77994d7a1698c4b",  
    "id": "ftjob-0f4191f0c59a4256b7a797a3d9eed219",  
    "created_at": 1695307968,  
    "updated_at": 1695310376,  
    "object": "fine_tuning.job"  
}  
Elapsed time: 40 minutes 45 seconds  
Status: running
```

학습을 완료하는 데 1시간 이상이 걸리는 경우가 드물지 않습니다. 학습이 완료되면 출력 메시지가 다음과 같이 변경됩니다.

출력

```
Fine-tuning job ftjob-b044a9d3cf9c4228b5d393567f693b83 finished with status:  
succeeded  
Checking other fine-tuning jobs for this resource.  
Found 2 fine-tune jobs.
```

전체 결과를 얻으려면 다음을 실행합니다.

OpenAI Python 1.x

Python

```
#Retrieve fine_tuned_model name  
  
response = client.fine_tuning.jobs.retrieve(job_id)  
  
print(response.model_dump_json(indent=2))  
fine_tuned_model = response.fine_tuned_model
```

미세 조정된 모델 배포

이 자습서의 이전 Python SDK 명령과 달리, 할당량 기능이 도입되었기 때문에 별도의 권한 부여, 다른 API 경로 및 다른 API 버전이 필요한 REST API를 사용하여 모델 배포를 수행해야 합니다.

또는 [Azure OpenAI Studio](#) 또는 [Azure CLI](#)와 같은 다른 일반적인 배포 방법을 사용하여 미세 조정된 모델을 배포할 수 있습니다.

테이블 확장

변수	정의
token	권한 부여 토큰을 생성하는 방법에는 여러 가지가 있습니다. 초기 테스트를 위한 가장 쉬운 방법은 Azure Portal 에서 Cloud Shell을 시작하는 것입니다. 그런 다음 <code>az account get-access-token</code> 를 실행합니다. 이 토큰을 API 테스트를 위한 임시 권한 부여 토큰으로 사용할 수 있습니다. 새 환경 변수에 저장하는 것이 좋습니다.
구독	연결된 Azure OpenAI 리소스에 대한 구독 ID
resource_group	Azure OpenAI 리소스의 리소스 그룹 이름
resource_name	Azure OpenAI 리소스 이름
model_deployment_name	미세 조정된 새 모델 배포의 사용자 지정 이름입니다. 채팅 완료 호출을 수행할 때 코드에서 참조되는 이름입니다.
fine_tuned_model	이전 단계의 미세 조정 작업 결과에서 이 값을 검색합니다. <code>gpt-35-turbo-0613.ft-b044a9d3cf9c4228b5d393567f693b83</code> 와 같이 표시됩니다. 해당 값을 <code>deploy_data.json</code> 에 추가해야 합니다.

중요

사용자 지정된 모델을 배포한 후 언제든지 배포가 15일 이상 비활성 상태로 유지되면 배포가 삭제됩니다. 모델이 배포된 지 15일이 넘었고 연속 15일 동안 모델에 대한 완료 또는 채팅 완료 호출이 이루어지지 않은 경우 맞춤형 모델 배포는 비활성 상태입니다.

비활성 배포를 삭제해도 기본 사용자 지정 모델은 삭제되거나 영향을 받지 않으며 사용자 지정 모델은 언제든지 다시 배포될 수 있습니다. [Azure OpenAI Service 가격 책정](#)에 설명된 대로 배포되는 각 사용자 지정(세밀 조정) 모델에는 완료 또는 채팅 완료 호출이 모델에 대해 수행되는지 여부에 관계없이 시간당 호스팅 비용이 발생합니다. Azure OpenAI를 사용하여 비용을 계획하고 관리하는 방법에 대한 자세한 내용은 [Azure OpenAI Service 비용 관리 계획](#)의 지침을 참조하세요.

Python

```
import json
import requests

token= os.getenv("TEMP_AUTH_TOKEN")
```

```

subscription = "<YOUR_SUBSCRIPTION_ID>"
resource_group = "<YOUR_RESOURCE_GROUP_NAME>"
resource_name = "<YOUR_AZURE_OPENAI_RESOURCE_NAME>"
model_deployment_name = "YOUR_CUSTOM_MODEL_DEPLOYMENT_NAME"

deploy_params = {'api-version': "2023-05-01"}
deploy_headers = {'Authorization': 'Bearer {}'.format(token), 'Content-Type': 'application/json'}

deploy_data = {
    "sku": {"name": "standard", "capacity": 1},
    "properties": {
        "model": {
            "format": "OpenAI",
            "name": "<YOUR_FINE_TUNED_MODEL>", #retrieve this value from the previous call, it will look like gpt-35-turbo-0613.ft-b044a9d3cf9c4228b5d393567f693b83
            "version": "1"
        }
    }
}
deploy_data = json.dumps(deploy_data)

request_url =
f'https://management.azure.com/subscriptions/{subscription}/resourceGroups/{resource_group}/providers/Microsoft.CognitiveServices/accounts/{resource_name}/deployments/{model_deployment_name}'

print('Creating a new deployment...')

r = requests.put(request_url, params=deploy_params, headers=deploy_headers,
data=deploy_data)

print(r)
print(r.reason)
print(r.json())

```

Azure OpenAI Studio에서 배포 진행률을 확인할 수 있습니다.

The screenshot shows the 'Deployments' section of the Azure OpenAI Studio. At the top, there are buttons for 'Create new deployment', 'Edit deployment', 'Delete deployment', 'Column options', 'Refresh', and 'Open in Playground'. Below this is a table with the following data:

Deployment name	Model name	Model version	Deployme...	Capacity	Status
<input checked="" type="checkbox"/> gpt-35-turbo-fine-tune	gpt-35-turbo-0613.ft-b044a9d3cf9c4228b5d393567f693b83	1	Standard	1K TPM	Creating

미세 조정된 모델 배포를 처리할 때 이 프로세스를 완료하는데 다소 시간이 걸리는 경우가 드물지 않습니다.

배포된 사용자 지정 모델 사용

미세 조정된 모델을 배포한 후 Azure OpenAI Studio의 채팅 플레이그라운드 [또는](#) 채팅 완료 API를 통해 배포된 다른 모델처럼 사용할 수 있습니다. 예를 들어 다음 Python 예제처럼 배포된 모델에 채팅 완료 호출을 보낼 수 있습니다. 배포된 다른 모델과 마찬가지로 사용자 지정 모델에서 온도 및 max_tokens와 같은 동일한 매개 변수를 계속 사용할 수 있습니다.

OpenAI Python 1.x

Python

```
import os
from openai import AzureOpenAI

client = AzureOpenAI(
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT"),
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2023-05-15"
)

response = client.chat.completions.create(
    model="gpt-35-turbo-ft", # model = "Custom deployment name you chose
for your fine-tuning model"
    messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Does Azure OpenAI support customer
managed keys?"},
        {"role": "assistant", "content": "Yes, customer managed keys are
supported by Azure OpenAI."},
        {"role": "user", "content": "Do other Azure AI services support
this too?"}
    ]
)

print(response.choices[0].message.content)
```

배포 삭제

다른 유형의 Azure OpenAI 모델과 달리, 미세 조정/사용자 지정된 모델은 [배포된 후 매시간 호스팅 비용이](#) 발생합니다. 이 자습서를 완료하고 미세 조정된 모델에 대해 몇 가지 채팅 완료 호출을 테스트한 후에는 모델 배포를 삭제하는 것이 좋습니다.

배포를 삭제해도 모델 자체에는 영향을 주지 않으므로 이 자습서에서 학습한 미세 조정된 모델을 언제든지 다시 배포할 수 있습니다.

REST API, Azure CLI 또는 기타 지원되는 배포 방법을 통해 Azure OpenAI Studio [에서](#) 배포를 삭제할 수 있습니다.

문제 해결

**미세 조정을 사용하도록 설정하려면 어떻게 해야 하나요?
사용자 지정 모델 만들기는 Azure OpenAI Studio에서 회색
으로 표시됩니다.**

미세 조정에 성공적으로 액세스하려면 Cognitive Services OpenAI 기여자가 할당되어야 합니다. 고급 서비스 관리자 권한이 있는 사용자도 미세 조정에 액세스하기 위해 이 계정을 명시적으로 설정해야 합니다. 자세한 내용은 [역할 기반 액세스 제어 지침](#)을 검토하세요.

다음 단계

- [Azure OpenAI에서 미세 조정에 대해 자세히 알아보기](#)
- [Azure OpenAI를 지원하는 기본 모델에 대해 자세히 알아봅니다.](#)

Azure OpenAI 음성 변환 채팅

아티클 • 2024. 02. 23.

참조 설명서 | [패키지\(NuGet\)](#) | [GitHub의 추가 샘플](#)

이 방법 가이드에서는 Azure AI Speech를 사용하여 Azure OpenAI Service와 대화할 수 있습니다. Speech Service에서 인식하는 텍스트는 Azure OpenAI로 전송됩니다. Speech Services는 Azure OpenAI의 텍스트 응답에서 음성을 합성합니다.

마이크를 사용하여 Azure OpenAI와 대화를 시작합니다.

- 음성 서비스는 사용자의 음성을 인식하여 텍스트로 변환합니다(음성 텍스트 변환).
- 텍스트 요청이 Azure OpenAI로 전송됩니다.
- 음성 서비스 텍스트 음성 변환 기능은 Azure OpenAI의 응답을 기본 스피커로 합성합니다.

이 예제의 환경은 앞뒤로 교환되지만 Azure OpenAI는 대화의 컨텍스트를 기억하지 못합니다.

① 중요

이 가이드의 단계를 완료하려면 Azure 구독에서 Microsoft Azure OpenAI Service에 대한 액세스 권한이 있어야 합니다. 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청합니다.

필수 조건

- ✓ Azure 구독 - [체험 구독 만들기](#)
- ✓ Azure Portal에서 [Microsoft Azure OpenAI Service 리소스를 만듭니다](#).
- ✓ Azure OpenAI 리소스에 [모델](#)을 배포합니다. 모델 배포에 대한 자세한 내용은 [리소스 배포 가이드](#)를 참조하세요.
- ✓ Azure OpenAI 리소스 키 및 엔드포인트를 가져옵니다. Azure OpenAI 리소스가 배포된 후, [리소스로 이동](#)을 선택하여 키를 보고 관리합니다. Azure AI 서비스 리소스에 대한 자세한 내용은 [리소스 키 가져오기](#)를 참조하세요.
- ✓ Azure Portal에서 [음성 리소스 만들기](#)
- ✓ 음성 리소스 키 및 지역을 가져옵니다. 음성 리소스가 배포된 후, [리소스로 이동](#)을 선택하여 키를 보고 관리합니다. Azure AI 서비스 리소스에 대한 자세한 내용은 [리소스 키 가져오기](#)를 참조하세요.

환경 설정

음성 SDK는 [NuGet 패키지](#)로 사용할 수 있으며 .NET Standard 2.0을 구현합니다. 이 가이드의 뒷부분에서 Speech SDK를 설치하지만, 먼저 [SDK 설치 가이드](#)에서 더 많은 요구 사항을 확인합니다.

환경 변수 설정

이 예제에는 `OPEN_AI_KEY`, `OPEN_AI_ENDPOINT`, `OPEN_AI_DEPLOYMENT_NAME`, `SPEECH_KEY` 및 `SPEECH_REGION`이라는 환경 변수가 필요합니다.

Azure AI 서비스 리소스에 액세스하려면 애플리케이션을 인증해야 합니다. 프로덕션의 경우 자격 증명을 안전하게 저장하고 액세스하는 방법을 사용합니다. 예를 들어, 음성 리소스에 대한 [키를 얻은](#) 후 애플리케이션을 실행하는 로컬 머신의 새 환경 변수에 이 키를 씁니다.

팁

코드에 키를 직접 포함하지 말고 공개적으로 게시하지 마세요. [Azure Key Vault](#)와 같은 추가 인증 옵션은 [Azure AI 서비스 보안](#)을 참조하세요.

환경 변수를 설정하려면 콘솔 창을 열고 운영 체제 및 개발 환경에 대한 지침을 따릅니다.

- `OPEN_AI_KEY` 환경 변수를 설정하려면 `your-openai-key`를 리소스에 대한 키 중 하나로 바꿉니다.
- `OPEN_AI_ENDPOINT` 환경 변수를 설정하려면 `your-openai-endpoint`을(를) 리소스에 대한 지역 중 하나로 바꿉니다.
- `OPEN_AI_DEPLOYMENT_NAME` 환경 변수를 설정하려면 `your-openai-deployment-name`을(를) 리소스에 대한 지역 중 하나로 바꿉니다.
- `SPEECH_KEY` 환경 변수를 설정하려면 `your-speech-key`를 리소스에 대한 키 중 하나로 바꿉니다.
- `SPEECH_REGION` 환경 변수를 설정하려면 `your-speech-region`을(를) 리소스에 대한 지역 중 하나로 바꿉니다.

Windows

콘솔

```
setx OPEN_AI_KEY your-openai-key  
setx OPEN_AI_ENDPOINT your-openai-endpoint  
setx OPEN_AI_DEPLOYMENT_NAME your-openai-deployment-name
```

```
setx SPEECH_KEY your-speech-key  
setx SPEECH_REGION your-speech-region
```

① 참고

현재 실행 중인 콘솔에서만 환경 변수에 액세스해야 하는 경우 환경 변수를 `setx` 대신 `set`로 설정합니다.

환경 변수를 추가한 후에는 콘솔 창을 포함하여 실행 중인 프로그램 중에서 환경 변수를 읽어야 하는 프로그램을 다시 시작해야 할 수도 있습니다. 예를 들어, Visual Studio가 편집기인 경우 예를 실행하기 전에 Visual Studio를 다시 시작합니다.

마이크에서 음성 인식

새 콘솔 애플리케이션을 만들려면 다음 단계를 수행합니다.

- 새 프로젝트를 원하는 폴더에서 명령 프롬프트 창을 엽니다. 이 명령을 실행하여 .NET CLI를 사용하여 콘솔 애플리케이션을 만듭니다.

.NET CLI

```
dotnet new console
```

명령은 프로젝트 디렉터리에 *Program.cs* 파일을 만듭니다.

- .NET CLI를 사용하여 새 프로젝트에 음성 SDK를 설치합니다.

.NET CLI

```
dotnet add package Microsoft.CognitiveServices.Speech
```

- .NET CLI를 사용하여 새 프로젝트에 Azure OpenAI SDK(시험판)를 설치합니다.

.NET CLI

```
dotnet add package Azure.AI.OpenAI --prerelease
```

- Program.cs*의 내용을 다음 코드로 바꿉니다.

C#

```

using System.Text;
using Microsoft.CognitiveServices.Speech;
using Microsoft.CognitiveServices.Speech.Audio;
using Azure;
using Azure.AI.OpenAI;

// This example requires environment variables named "OPEN_AI_KEY",
// "OPEN_AI_ENDPOINT" and "OPEN_AI_DEPLOYMENT_NAME"
// Your endpoint should look like the following
// https://YOUR_OPEN_AI_RESOURCE_NAME.openai.azure.com/
string openAIKey = Environment.GetEnvironmentVariable("OPEN_AI_KEY") ??
                  throw new ArgumentException("Missing OPEN_AI_KEY");
string openAIEndpoint =
Environment.GetEnvironmentVariable("OPEN_AI_ENDPOINT") ??
                  throw new ArgumentException("Missing
OPEN_AI_ENDPOINT");

// Enter the deployment name you chose when you deployed the model.
string engine =
Environment.GetEnvironmentVariable("OPEN_AI_DEPLOYMENT_NAME") ??
                  throw new ArgumentException("Missing
OPEN_AI_DEPLOYMENT_NAME");

// This example requires environment variables named "SPEECH_KEY" and
// "SPEECH_REGION"
string speechKey = Environment.GetEnvironmentVariable("SPEECH_KEY") ??
                  throw new ArgumentException("Missing SPEECH_KEY");
string speechRegion =
Environment.GetEnvironmentVariable("SPEECH_REGION") ??
                  throw new ArgumentException("Missing
SPEECH_REGION");

// Sentence end symbols for splitting the response into sentences.
List<string> sentenceSaperators = new() { ".", "!", "?", ";", ".",
"!", "?", ";", "\n" };

try
{
    await ChatWithOpenAI();
}
catch (Exception ex)
{
    Console.WriteLine(ex);
}

// Prompts Azure OpenAI with a request and synthesizes the response.
async Task AskOpenAI(string prompt)
{
    object consoleLock = new();
    var speechConfig = SpeechConfig.FromSubscription(speechKey,
speechRegion);

    // The language of the voice that speaks.
    speechConfig.SpeechSynthesisVoiceName = "en-US-

```

```
JennyMultilingualNeural";
    var audioOutputConfig = AudioConfig.FromDefaultSpeakerOutput();
    using var speechSynthesizer = new SpeechSynthesizer(speechConfig,
audioOutputConfig);
    speechSynthesizer.Synthesizing += (sender, args) =>
{
    lock (consoleLock)
    {
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.Write($"[Audio]");
        Console.ResetColor();
    }
};

// Ask Azure OpenAI
OpenAIClient client = new(new Uri(openAIEndpoint), new
AzureKeyCredential(openAIKey));
var completionsOptions = new ChatCompletionsOptions()
{
    DeploymentName = engine,
    Messages = { new ChatRequestUserMessage(prompt) },
    MaxTokens = 100,
};
var responseStream = await
client.GetChatCompletionsStreamingAsync(completionsOptions);

StringBuilder gptBuffer = new();
await foreach (var completionUpdate in responseStream)
{
    var message = completionUpdate.ContentUpdate;
    if (string.IsNullOrEmpty(message))
    {
        continue;
    }

    lock (consoleLock)
    {
        Console.ForegroundColor = ConsoleColor.DarkBlue;
        Console.WriteLine($"{message}");
        Console.ResetColor();
    }

    gptBuffer.Append(message);

    if (sentenceSaperators.Any(message.Contains))
    {
        var sentence = gptBuffer.ToString().Trim();
        if (!string.IsNullOrEmpty(sentence))
        {
            await speechSynthesizer.SpeakTextAsync(sentence);
            gptBuffer.Clear();
        }
    }
}
}
```

```
// Continuously listens for speech input to recognize and send as text
// to Azure OpenAI
async Task ChatWithOpenAI()
{
    // Should be the locale for the speaker's language.
    var speechConfig = SpeechConfig.FromSubscription(speechKey,
speechRegion);
    speechConfig.SpeechRecognitionLanguage = "en-US";

    using var audioConfig = AudioConfig.FromDefaultMicrophoneInput();
    using var speechRecognizer = new SpeechRecognizer(speechConfig,
audioConfig);
    var conversationEnded = false;

    while (!conversationEnded)
    {
        Console.WriteLine("Azure OpenAI is listening. Say 'Stop' or
press Ctrl-Z to end the conversation.");

        // Get audio from the microphone and then send it to the TTS
        // service.
        var speechRecognitionResult = await
speechRecognizer.RecognizeOnceAsync();

        switch (speechRecognitionResult.Reason)
        {
            case ResultReason.RecognizedSpeech:
                if (speechRecognitionResult.Text == "Stop.")
                {
                    Console.WriteLine("Conversation ended.");
                    conversationEnded = true;
                }
                else
                {
                    Console.WriteLine($"Recognized speech:
{speechRecognitionResult.Text}");
                    await AskOpenAI(speechRecognitionResult.Text);
                }
                break;
            case ResultReason.NoMatch:
                Console.WriteLine($"No speech could be recognized: ");
                break;
            case ResultReason.Canceled:
                var cancellationDetails =
CancellationDetails.FromResult(speechRecognitionResult);
                Console.WriteLine($"Speech Recognition canceled:
{cancellationDetails.Reason}");
                if (cancellationDetails.Reason ==
CancellationReason.Error)
                {
                    Console.WriteLine($"Error details=
{cancellationDetails.ErrorDetails}");
                }
        }
    }
}
```

```
        break;  
    }  
}  
}
```

5. Azure OpenAI에서 반환되는 토큰 수를 늘리거나 줄이려면 `ChatCompletionsOptions` 클래스 인스턴스에서 `MaxTokens` 속성을 변경하세요. 토큰 및 비용 관련 자세한 내용은 [Azure OpenAI 토큰](#) 및 [Azure OpenAI 가격 책정](#)을 참조하세요.
6. 새 콘솔 애플리케이션을 실행하여 마이크의 음성 인식을 시작합니다.

콘솔

```
dotnet run
```

ⓘ 중요

설명된 대로 `OPEN_AI_KEY`, `OPEN_AI_ENDPOINT`, `OPEN_AI_DEPLOYMENT_NAME`, `SPEECH_KEY` 및 `SPEECH_REGION` 환경 변수를 설정했는지 확인합니다. 이 변수를 설정하지 않으면 샘플이 오류 메시지와 함께 실패합니다.

메시지가 표시되면 마이크에 말합니다. 콘솔 출력에는 말하기를 시작하라는 프롬프트, 텍스트로 요청, Azure OpenAI의 응답이 텍스트로 포함됩니다. Azure OpenAI의 응답을 텍스트에서 음성으로 변환한 다음 기본 스피커로 출력해야 합니다.

콘솔

```
PS C:\dev\openai\csharp> dotnet run  
Azure OpenAI is listening. Say 'Stop' or press Ctrl-Z to end the  
conversation.  
Recognized speech:Make a comma separated list of all continents.  
Azure OpenAI response:Africa, Antarctica, Asia, Australia, Europe, North  
America, South America  
Speech synthesized to speaker for text [Africa, Antarctica, Asia, Australia,  
Europe, North America, South America]  
Azure OpenAI is listening. Say 'Stop' or press Ctrl-Z to end the  
conversation.  
Recognized speech: Make a comma separated list of 1 Astronomical observatory  
for each continent. A list should include each continent name in  
parentheses.  
Azure OpenAI response:Mauna Kea Observatories (North America), La Silla  
Observatory (South America), Tenerife Observatory (Europe), Siding Spring  
Observatory (Australia), Beijing Xinglong Observatory (Asia), Naukluft  
Plateau Observatory (Africa), Rutherford Appleton Laboratory (Antarctica)  
Speech synthesized to speaker for text [Mauna Kea Observatories (North  
America), La Silla Observatory (South America), Tenerife Observatory
```

```
(Europe), Siding Spring Observatory (Australia), Beijing Xinglong  
Observatory (Asia), Naukluft Plateau Observatory (Africa), Rutherford  
Appleton Laboratory (Antarctica)]  
Azure OpenAI is listening. Say 'Stop' or press Ctrl-Z to end the  
conversation.  
Conversation ended.  
PS C:\dev\openai\csharp>
```

설명

다음은 몇 가지 추가 고려 사항입니다.

- 음성 인식 언어를 변경하려면 `en-US`를 다른 지원되는 언어로 바꿉니다. 예를 들어 스페인어(스페인)의 경우 `es-ES`입니다. 기본 언어는 `en-US`입니다. 음성에 사용될 수 있는 여러 언어 중 하나를 식별하는 방법에 대한 자세한 내용은 [언어 식별](#)을 참조하세요.
- 들리는 음성을 변경하려면 `en-US-JennyMultilingualNeural`을 지원되는 다른 음성으로 바꿀 있습니다. 음성이 Azure OpenAI에서 반환된 텍스트의 언어를 모르는 경우 Speech Service에서 합성된 오디오를 출력하지 않습니다.
- 다른 모델을 사용하려면 `gpt-35-turbo-instruct`를 다른 배포의 ID로 바꿉니다. 배포 ID가 모델 이름과 반드시 동일할 필요는 없습니다. [Azure OpenAI Studio](#)에서 배포를 만들 때 배포 이름을 지정했습니다.
- 또한 Azure OpenAI는 프롬프트 입력 및 생성된 출력에서 콘텐츠 조정을 수행합니다. 유해한 콘텐츠가 감지되면 프롬프트나 응답이 필터링될 수 있습니다. 자세한 내용은 [콘텐츠 필터링](#) 문서를 참조하세요.

리소스 정리

Azure Portal 또는 Azure CLI([명령줄 인터페이스](#))를 사용하여 생성된 음성 리소스를 제거 할 수 있습니다.

관련 콘텐츠

- [Speech에 대한 자세한 정보](#)
- [Azure OpenAI에 대해 자세히 알아보기](#)

Overview of Responsible AI practices for Azure OpenAI models

Article • 02/27/2024

Many of the Azure OpenAI models are generative AI models that have demonstrated improvements in advanced capabilities such as content and code generation, summarization, and search. With many of these improvements also come increased responsible AI challenges related to harmful content, manipulation, human-like behavior, privacy, and more. For more information about the capabilities, limitations and appropriate use cases for these models, please review the [Transparency Note](#).

In addition to the Transparency Note, we have created technical recommendations and resources to help customers design, develop, deploy, and use AI systems that implement the Azure OpenAI models responsibly. Our recommendations are grounded in the [Microsoft Responsible AI Standard](#), which sets policy requirements that our own engineering teams follow. Much of the content of the Standard follows a pattern, asking teams to Identify, Measure, and Mitigate potential harms, and plan for how to Operate the AI system as well. In alignment with those practices, these recommendations are organized into four stages:

1. **Identify** : Identify and prioritize potential harms that could result from your AI system through iterative red-teaming, stress-testing, and analysis.
2. **Measure** : Measure the frequency and severity of those harms by establishing clear metrics, creating measurement test sets, and completing iterative, systematic testing (both manual and automated).
3. **Mitigate** : Mitigate harms by implementing tools and strategies such as [prompt engineering](#) and using our [content filters](#). Repeat measurement to test effectiveness after implementing mitigations.
4. **Operate** : Define and execute a deployment and operational readiness plan.

In addition to their correspondence to the Microsoft Responsible AI Standard, these stages correspond closely to the functions in the [NIST AI Risk Management Framework](#).

Identify

Identifying potential harms that could occur in or be caused by an AI system is the first stage of the Responsible AI lifecycle. The earlier you begin to identify potential harms, the more effective you can be at mitigating the harms. When assessing potential harms, it is important to develop an understanding of the types of harms that could result from

using the Azure OpenAI Service in your specific context(s). In this section, we provide recommendations and resources you can use to identify harms through an impact assessment, iterative red team testing, stress-testing, and analysis. Red teaming and stress-testing are approaches where a group of testers come together and intentionally probe a system to identify its limitations, risk surface, and vulnerabilities.

These steps have the goal of producing a prioritized list of potential harms for each specific scenario.

- 1. Identify harms that are relevant** for your specific model, application, and deployment scenario.
 - a. Identify potential harms associated with the model and model capabilities (for example, GPT-3 model vs GPT-4 model) that you're using in your system. This is important to consider because each model has different capabilities, limitations, and risks, as described more fully in the sections above.
 - b. Identify any other harms or increased scope of harm presented by the intended use of the system you're developing. Consider using a [Responsible AI Impact Assessment](#) to identify potential harms.
 - i. For example, let's consider an AI system that summarizes text. Some uses of text generation are lower risk than others. For example, if the system is to be used in a healthcare domain for summarizing doctor's notes, the risk of harm arising from inaccuracies is higher than if the system is summarizing online articles.
- 2. Prioritize harms based on elements of risk such as frequency and severity.** Assess the level of risk for each harm and the likelihood of each risk occurring in order to prioritize the list of harms you've identified. Consider working with subject matter experts and risk managers within your organization and with relevant external stakeholders when appropriate.
- 3. Conduct red team testing and stress testing** starting with the highest priority harms, to develop a better understanding of whether and how the identified harms are actually occurring in your scenario, as well as to identify new harms you didn't initially anticipate.
- 4. Share this information with relevant stakeholders** using your organization's internal compliance processes.

At the end of this Identify stage, you should have a documented, prioritized list of harms. When new harms and new instances of harms emerge through further testing and use of the system, you can update and improve this list by following the above process again.

Measure

Once a list of prioritized harms has been identified, the next stage involves developing an approach for systematic measurement of each harm and conducting evaluations of the AI system. There are manual and automated approaches to measurement. We recommend you do both, starting with manual measurement.

Manual measurement is useful for:

1. Measuring progress on a small set of priority issues. When mitigating specific harms, it's often most productive to keep manually checking progress against a small dataset until the harm is no longer observed before moving to automated measurement.
2. Defining and reporting metrics until automated measurement is reliable enough to use alone.
3. Spot-checking periodically to measure the quality of automatic measurement.

Automated measurement is useful for:

1. Measuring at a large scale with increased coverage to provide more comprehensive results.
2. Ongoing measurement to monitor for any regression as the system, usage, and mitigations evolve.

Below, we provide specific recommendations to measure your AI system for potential harms. We recommend you first complete this process manually and then develop a plan to automate the process:

1. **Create inputs that are likely to produce each prioritized harm:** Create measurement set(s) by generating many diverse examples of targeted inputs that are likely to produce each prioritized harm.
2. **Generate System Outputs:** Pass in the examples from the measurement sets as inputs to the system to generate system outputs. Document the outputs.
3. **Evaluate System Outputs and Report Results to Relevant Stakeholders**
 - a. **Define clear metric(s).** For each intended use of your system, establish metrics that measure the frequency and degree of severity of each potentially harmful output. Create clear definitions to classify outputs that will be considered harmful or problematic in the context of your system and scenario, for each type of prioritized harm you identified.
 - b. **Assess the outputs** against the clear metric definitions and record and quantify the occurrences of harmful outputs. Repeat the measurements periodically, to assess mitigations and monitor for any regression.

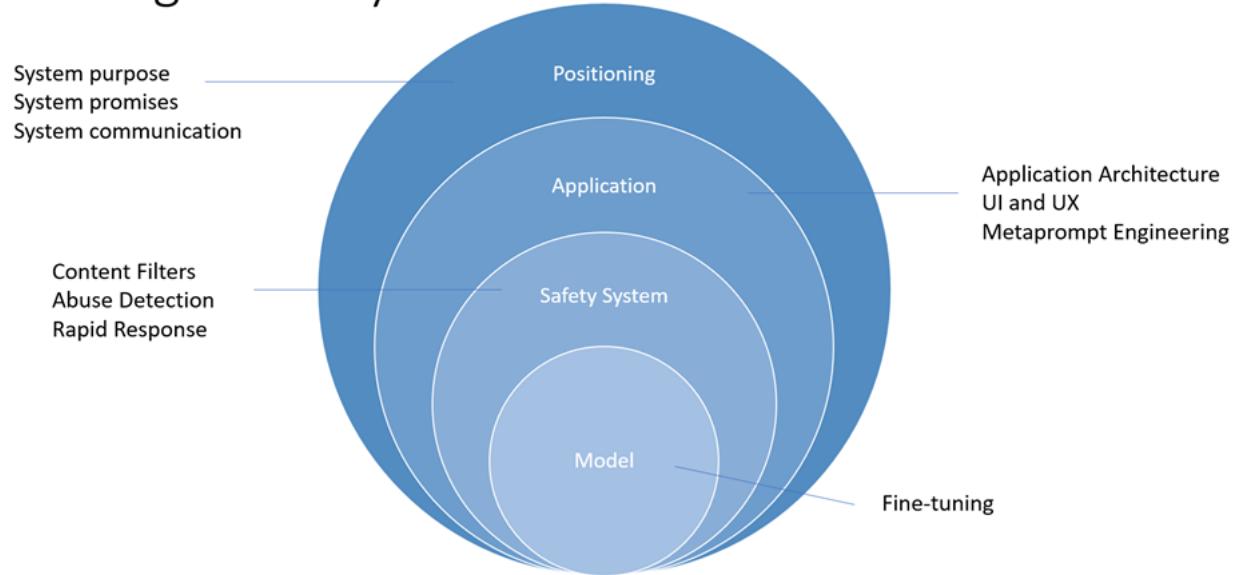
- c. Share this information with relevant stakeholders using your organization's internal compliance processes.

At the end of this measurement stage, you should have a defined measurement approach to benchmark how your system performs for each potential harm as well as an initial set of documented results. As you continue implementing and testing mitigations, the metrics and measurement sets should continue to be refined (for example, to add metrics for new harms that were initially unanticipated) and the results updated.

Mitigate

Mitigating harms presented by large language models such as the Azure OpenAI models requires an iterative, layered approach that includes experimentation and continual measurement. We recommend developing a mitigation plan that encompasses four layers of mitigations for the harms identified in the earlier stages of this process:

Mitigation Layers



1. At the **model level**, it's important to understand the model(s) you'll be using and what fine-tuning steps may have been taken by the model developers to align the model towards its intended uses and to reduce the risk of potentially harmful uses and outcomes.
 - a. For example, for GPT-4, model developers have been able to use reinforcement learning methods as a responsible AI tool to better align the model towards the designers' intended goals.
2. At the **safety system level**, you should understand the platform level mitigations that have been implemented, such as the [Azure OpenAI content filters](#) which help

to block the output of harmful content.

3. At the **application level**, application developers can implement metaprompt and user-centered design and user experience mitigations. Metaprompts are instructions provided to the model to guide its behavior; their use can make a critical difference in guiding the system to behave in accordance with your expectations. User-centered design and user experience (UX) interventions are also key mitigation tools to prevent misuse and overreliance on AI.
4. At the **positioning level**, there are many ways to educate the people who will use or be affected by your system about its capabilities and limitations.

Below, we provide specific recommendations to implement mitigations at the different layers. Not all of these mitigations are appropriate for every scenario, and conversely, these mitigations may be insufficient for some scenarios. Give careful consideration to your scenario and the prioritized harms you identified, and as you implement mitigations, develop a process to **measure and document their effectiveness** for your system and scenario.

1. **Model level Mitigations:** Review and identify which Azure OpenAI base model is best suited for the system you're building and educate yourself about its capabilities, limitations, and any measures taken to reduce the risk of the potential harms you've identified. For example, if you're using GPT-4, in addition to reading this Transparency Note, you can review OpenAI's [GPT-4 System Card](#) explaining the safety challenges presented by the model and the safety processes that OpenAI adopted to prepare GPT-4 for deployment. It may be worth experimenting with different versions of the model(s) (including through red teaming and measuring) to see how the harms present differently.
2. **Safety System Level Mitigations:** Identify and evaluate the effectiveness of platform level solutions such as the [Azure OpenAI content filters](#) to help mitigate the potential harms that you have identified.
3. **Application Level Mitigations:** Prompt engineering, including **metaprompt tuning, can be an effective mitigation** for many different types of harm. Review and implement metaprompt (also called the "system message" or "system prompt") guidance and best practices documented [here](#).

We recommend implementing the following user-centered design and user experience (UX) interventions, guidance, and best practices to guide users to use the system as intended and to prevent overreliance on the AI system:

- a. **Review and edit interventions:** Design the user experience (UX) to encourage people who use the system to review and edit the AI-generated outputs before accepting them (see [HAX G9](#): Support efficient correction).

b. Highlight potential inaccuracies in the AI-generated outputs (see HAX G2 ↗:

Make clear how well the system can do what it can do), both when users first start using the system and at appropriate times during ongoing use. In the first run experience (FRE), notify users that AI-generated outputs may contain inaccuracies and that they should verify information. Throughout the experience, include reminders to check AI-generated output for potential inaccuracies, both overall and in relation to specific types of content the system may generate incorrectly. For example, if your measurement process has determined that your system has lower accuracy with numbers, mark numbers in generated outputs to alert the user and encourage them to check the numbers or seek external sources for verification.

- c. User responsibility.** Remind people that they are accountable for the final content when they're reviewing AI-generated content. For example, when offering code suggestions, remind the developer to review and test suggestions before accepting.
- d. Disclose AI's role in the interaction.** Make people aware that they are interacting with an AI system (as opposed to another human). Where appropriate, inform content consumers that content has been partly or fully generated by an AI model; such notices may be required by law or applicable best practices, and can reduce inappropriate reliance on AI-generated outputs and can help consumers use their own judgment about how to interpret and act on such content.
- e. Prevent the system from anthropomorphizing.** AI models may output content containing opinions, emotive statements, or other formulations that could imply that they're human-like, that could be mistaken for a human identity, or that could mislead people to think that a system has certain capabilities when it doesn't. Implement mechanisms that reduce the risk of such outputs or incorporate disclosures to help prevent misinterpretation of outputs.
- f. Cite references and information sources.** If your system generates content based on references sent to the model, clearly citing information sources helps people understand where the AI-generated content is coming from.
- g. Limit the length of inputs and outputs, where appropriate.** Restricting input and output length can reduce the likelihood of producing undesirable content, misuse of the system beyond its intended uses, or other harmful or unintended uses.
- h. Structure inputs and/or system outputs.** Use [prompt engineering](#) techniques within your application to structure inputs to the system to prevent open-ended responses. You can also limit outputs to be structured in certain formats or patterns. For example, if your system generates dialog for a fictional character in

response to queries, limit the inputs so that people can only query for a predetermined set of concepts.

- i. **Prepare pre-determined responses.** There are certain queries to which a model may generate offensive, inappropriate, or otherwise harmful responses. When harmful or offensive queries or responses are detected, you can design your system to deliver a predetermined response to the user. Predetermined responses should be crafted thoughtfully. For example, the application can provide prewritten answers to questions such as "who/what are you?" to avoid having the system respond with anthropomorphized responses. You can also use predetermined responses for questions like, "What are your terms of use?" to direct people to the correct policy.
- j. **Restrict automatic posting on social media.** Limit how people can automate your product or service. For example, you may choose to prohibit automated posting of AI-generated content to external sites (including social media), or to prohibit the automated execution of generated code.
- k. **Bot detection.** Devise and implement a mechanism to prohibit users from building an API on top of your product.

4. Positioning Level Mitigations:

- a. **Be appropriately transparent.** It's important to provide the right level of transparency to people who use the system, so that they can make informed decisions around the use of the system.
- b. **Provide system documentation.** Produce and provide educational materials for your system, including explanations of its capabilities and limitations. For example, this could be in the form of a "learn more" page accessible via the system.
- c. **Publish user guidelines and best practices.** Help users and stakeholders use the system appropriately by publishing best practices, for example on prompt crafting, reviewing generations before accepting them, etc. Such guidelines can help people understand how the system works. When possible, incorporate the guidelines and best practices directly into the UX.

As you implement mitigations to address potential identified harms, it's important to develop a process for ongoing measurement of the effectiveness of such mitigations, to document measurement results, and to review those measurement results to continually improve the system.

Operate

Once measurement and mitigation systems are in place, we recommend that you define and execute a deployment and operational readiness plan. This stage includes

completing appropriate reviews of your system and mitigation plans with relevant stakeholders, establishing pipelines to collect telemetry and feedback, and developing an incident response and rollback plan.

Some recommendations for how to deploy and operate a system that uses the Azure OpenAI service with appropriate, targeted harms mitigations include:

1. Work with compliance teams within your organization to understand what types of reviews are required for your system and when they are required (for example, legal review, privacy review, security review, accessibility review, etc.).
2. Develop and implement the following:
 - a. **Develop a phased delivery plan.** We recommend you launch systems using the Azure OpenAI service gradually using a "phased delivery" approach. This gives a limited set of people the opportunity to try the system, provide feedback, report issues and concerns, and suggest improvements before the system is released more widely. It also helps to manage the risk of unanticipated failure modes, unexpected system behaviors, and unexpected concerns being reported.
 - b. **Develop an incident response plan.** Develop an incident response plan and evaluate the time needed to respond to an incident.
 - c. **Develop a rollback plan** Ensure you can roll back the system quickly and efficiently in case an unanticipated incident occurs.
 - d. **Prepare for immediate action for unanticipated harms.** Build the necessary features and processes to block problematic prompts and responses as they're discovered and as close to real-time as possible. When unanticipated harms do occur, block the problematic prompts and responses as quickly as possible, develop and deploy appropriate mitigations, investigate the incident, and implement a long-term solution.
 - e. **Develop a mechanism to block people who are misusing your system.** Develop a mechanism to identify users who violate your content policies (for example, by generating hate speech) or are otherwise using your system for unintended or harmful purposes, and take action against further abuse. For example, if a user frequently uses your system to generate content that is blocked or flagged by content safety systems, consider blocking them from further use of your system. Implement an appeal mechanism where appropriate.
 - f. **Build effective user feedback channels.** Implement feedback channels through which stakeholders (and the general public, if applicable) can submit feedback or report issues with generated content or that otherwise arise during their use of the system. Document how such feedback is processed, considered, and addressed. Evaluate the feedback and work to improve the system based on user feedback. One approach could be to include buttons with generated content that would allow users to identify content as "inaccurate," "harmful" or

"incomplete." This could provide a more widely used, structured and feedback signal for analysis.

- g. **Telemetry data.** Identify and record (consistent with applicable privacy laws, policies, and commitments) signals that indicate user satisfaction or their ability to use the system as intended. Use telemetry data to identify gaps and improve the system.

This document is not intended to be, and should not be construed as providing, legal advice. The jurisdiction in which you're operating may have various regulatory or legal requirements that apply to your AI system. Consult a legal specialist if you are uncertain about laws or regulations that might apply to your system, especially if you think those might impact these recommendations. Be aware that not all of these recommendations and resources are appropriate for every scenario, and conversely, these recommendations and resources may be insufficient for some scenarios.

Learn more about responsible AI

- Microsoft AI principles [↗](#)
- Microsoft responsible AI resources [↗](#)
- Microsoft Azure Learning courses on responsible AI

Learn more about Azure OpenAI

- Limited access to Azure OpenAI Service - Azure AI services | Microsoft Learn
- Code of Conduct for the Azure OpenAI Service | Microsoft Learn
- Data, privacy, and security for Azure OpenAI Service - Azure AI services | Microsoft Learn

Transparency Note for Azure OpenAI Service

Article • 02/04/2024

What is a Transparency Note?

An AI system includes not only the technology, but also the people who use it, the people who are affected by it, and the environment in which it's deployed. Creating a system that is fit for its intended purpose requires an understanding of how the technology works, what its capabilities and limitations are, and how to achieve the best performance. Microsoft's Transparency Notes are intended to help you understand how our AI technology works, the choices system owners can make that influence system performance and behavior, and the importance of thinking about the whole system, including the technology, the people, and the environment. You can use Transparency Notes when developing or deploying your own system, or share them with the people who will use or be affected by your system.

Microsoft's Transparency Notes are part of a broader effort at Microsoft to put our AI Principles into practice. To find out more, see the [Microsoft's AI principles](#).

The basics of the Azure OpenAI Models

Azure OpenAI provides customers with a fully managed AI service that lets developers and data scientists apply OpenAI's powerful models including models that can generate natural language, code, and images. Within the Azure OpenAI Service, the OpenAI models are integrated with Microsoft-developed content filtering and abuse detection models. Learn more about content filtering [here](#) and abuse detection [here](#).

Select the tabs to see content for the relevant model type.

Introduction

Text, code, and fine-tuned models

As part of the fully managed Azure OpenAI Service, the GPT-3 models analyze and generate natural language, Codex models analyze and generate code and plain text code commentary, and the GPT-4 models can understand and generate natural language and code. These models use an autoregressive architecture, meaning they

use data from prior observations to predict the most probable next word. This process is then repeated by appending the newly generated content to the original text to produce the complete generated response. Because the response is conditioned on the input text, these models can be applied to various tasks simply by changing the input text.

The GPT-3 series of models are pretrained on a wide body of publicly available free text data. This data is sourced from a combination of web crawling (specifically, a filtered version of [Common Crawl](#)), which includes a broad range of text from the internet and comprises 60 percent of the weighted pretraining dataset) and higher-quality datasets, including an expanded version of the WebText dataset, two internet-based books corpora and English-language Wikipedia. The GPT-4 base model was trained using publicly available data (such as internet data) and data that was licensed by OpenAI. The model was fine-tuned using reinforcement learning with human feedback (RLHF).

Learn more about the training and modeling techniques in OpenAI's [GPT-3](#), [GPT-4](#), and [Codex](#) research papers. The guidance below is also drawn from [OpenAI's safety best practices](#).

Fine tuning refers to using *Supervised Fine Tuning* to adjust a base model's weights to provide better responses based on a provided training set. All use cases and considerations for large language models apply to fine-tuned models, but there are additional considerations as well.

ⓘ Important

Fine-tuning is only available for text and code models, not vision or speech models.

Key terms

 [Expand table](#)

Term	Definition
Prompt	<p>The text you send to the service in the API call. This text is then input into the model. For example, one might input the following prompt:</p> <pre>Convert the questions to a command: Q: Ask Constance if we need some bread A: send-msg 'find constance' Do we need some bread?</pre>

Term	Definition
	<p>Q: Send a message to Greg to figure out if things are ready for Wednesday.</p> <p>A:</p>
Completion or Generation	<p>The text Azure OpenAI outputs in response. For example, the service may respond with the following answer to the above prompt: send-msg 'find greg' figure out if things are ready for Wednesday.</p>
Token	<p>Azure OpenAI processes text by breaking it down into tokens. Tokens can be words or just chunks of characters. For example, the word <code>hamburger</code> gets broken up into the tokens <code>ham</code>, <code>bur</code> and <code>ger</code>, while a short and common word like <code>pear</code> is a single token. Many tokens start with a whitespace, for example <code>hello</code> and <code>bye</code>.</p>
Fine tuning	<p>Supervised fine-tuning for large language models refers to the process of taking a pre-trained language model, often trained on a massive dataset, and further training it on a more specific task with labeled data. This involves adjusting the weights of the model using this smaller, specific dataset so that the model becomes more specialized in the tasks it can perform, enhancing its performance and accuracy.</p>
Model Weights	<p>Model weights are parameters within the model that are learned from the data during the training process. They determine the output of the model for a given input. These weights are adjusted in response to the error the model made in its predictions, with the aim of minimizing this error.</p>

Capabilities

Text, code, and fine-tuned models

The GPT-4, GPT-3, and Codex Azure OpenAI Service models use natural language instructions and examples in the prompt to identify the task. The model then completes the task by predicting the most probable next text. This technique is known as "in-context" learning. These models are not retrained during this step but instead give predictions based on the context you include in the prompt.

There are three main approaches for in-context learning. These approaches vary based on the amount of task-specific data that is given to the model:

Few-shot : In this case, a user includes several examples in the prompt that demonstrate the expected answer format and content. The following example shows a few-shot prompt providing multiple examples:

Convert the questions to a command:

Q: Ask Constance if we need some bread

A: send-msg `find constance` Do we need some bread?

Q: Send a message to Greg to figure out if things are ready for Wednesday.

A: send-msg `find greg` Is everything ready for Wednesday?

Q: Ask Ilya if we're still having our meeting this evening

A: send-msg `find ilya` Are we still having a meeting this evening?

Q: Contact the ski store and figure out if I can get my skis fixed before I leave on Thursday

A: send-msg `find ski store` Would it be possible to get my skis fixed before I leave on Thursday?

Q: Thank Nicolas for lunch

A: send-msg `find nicolas` Thank you for lunch!

Q: Tell Constance that I won't be home before 19:30 tonight – unmovable meeting.

A: send-msg `find constance` I won't be home before 19:30 tonight. I have a meeting I can't move.

Q: Tell John that I need to book an appointment at 10:30

A:

The number of examples typically ranges from 0 to 100 depending on how many can fit in the maximum input length for a single prompt. Few-shot learning enables a major reduction in the amount of task-specific data required for accurate predictions.

One-shot : This case is the same as the few-shot approach except only one example is provided. The following example shows a one-shot prompt:

Convert the questions to a command:

Q: Ask Constance if we need some bread

A: send-msg `find constance` Do we need some bread?

Q: Send a message to Greg to figure out if things are ready for Wednesday.

A:

Zero-shot: In this case, no examples are provided to the model and only the task request is provided. The following example shows a zero-shot prompt:

Convert the question to a command:

Q: Ask Constance if we need some bread

A:

Use cases

Text, code, and fine-tuned models

Intended uses

The GPT-4, GPT-3, and Codex models in the Azure OpenAI service can be used in multiple scenarios. The following list isn't comprehensive, but it illustrates the diversity of tasks that can be supported with appropriate mitigations:

- **Chat and conversation interaction** : Users can interact with a conversational agent that responds with responses drawn from trusted documents such as internal company documentation or tech support documentation. Conversations must be limited to answering scoped questions.
- **Chat and conversation creation** : Users can create a conversational agent that responds with responses drawn from trusted documents such as internal company documentation or tech support documentation. Conversations must be limited to answering scoped questions.
- **Code generation or transformation scenarios** : For example, converting one programming language to another, generating docstrings for functions, converting natural language to SQL.
- **Journalistic content** : For use to create new journalistic content or to rewrite journalistic content submitted by the user as a writing aid for predefined topics. Users cannot use the application as a general content creation tool for all topics.
- **Question-answering** : Users can ask questions and receive answers from trusted source documents such as internal company documentation. The application does not generate answers ungrounded in trusted source documentation.
- **Reason over structured and unstructured data** : Users can analyze inputs using classification, sentiment analysis of text, or entity extraction. Examples include analyzing product feedback sentiment, analyzing support calls and transcripts, and refining text-based search with embeddings.
- **Search** : Users can search trusted source documents such as internal company documentation. The application does not generate results ungrounded in trusted source documentation.
- **Summarization** : Users can submit content to be summarized for predefined topics built into the application and cannot use the application as an open-

ended summarizer. Examples include summarization of internal company documentation, call center transcripts, technical reports, and product reviews.

- **Writing assistance on specific topics** : Users can create new content or rewrite content submitted by the user as a writing aid for business content or pre-defined topics. Users can only rewrite or create content for specific business purposes or predefined topics and cannot use the application as a general content creation tool for all topics. Examples of business content include proposals and reports. For journalistic use, see above **Journalistic content** use case.
- **Data generation for fine-tuning**: Users can use a model in Azure OpenAI to generate data which is used solely to fine-tune (i) another Azure OpenAI model, using the fine-tuning capabilities of Azure OpenAI, and/or (ii) another Azure AI custom model, using the fine-tuning capabilities of the Azure AI service. Generating data and fine-tuning models is limited to internal users only; the fine-tuned model may only be used for inferencing in the applicable Azure AI service and, for Azure OpenAI service, only for customer's permitted use case(s) under this form.

Fine-tuned use cases

The following are additional use cases we recommend for fine-tuned text and code models. Fine tuning is most appropriate for:

- **Steering the style, format, tone or qualitative aspects of responses** via examples of the desired responses.
- **Ensuring the model reliably produces a desired output** such as providing responses in a specific format or ensuring responses are grounded by information in the prompt.
- **Use cases with many edge cases** that cannot be covered within examples in the prompt, such as complex natural language to code examples.
- **Improving performance at specific skills or tasks** such as classification, summarization, or formatting – that can be hard to describe within a prompt.
- **Reducing costs or latency** by utilizing shorter prompts, or swapping a fine-tuned version of a smaller/faster model for a more general-purpose model (e.g. fine tuned GPT-3.5-Turbo for GPT-4).

As with base models, the use case prohibitions outlined in the [Azure OpenAI Code of conduct](#) apply to fine-tuned models as well.

Fine tuning alone is not recommended for scenarios where you want to extend your model to include out-of-domain information, where explainability or grounding are important, or where the underlying data are updated frequently.

Considerations when choosing a use case

We encourage customers to use the Azure OpenAI GPT-4, GPT-3, and Codex models in their innovative solutions or applications as approved in their [Limited Access registration form](#). However, here are some considerations when choosing a use case:

- **Not suitable for open-ended, unconstrained content generation.** Scenarios where users can generate content on any topic are more likely to produce offensive or harmful text. The same is true of longer generations.
- **Not suitable for scenarios where up-to-date, factually accurate information is crucial** unless you have human reviewers or are using the models to search your own documents and have verified suitability for your scenario. The service does not have information about events that occur after its training date, likely has missing knowledge about some topics, and may not always produce factually accurate information.
- **Avoid scenarios where use or misuse of the system could result in significant physical or psychological injury to an individual.** For example, scenarios that diagnose patients or prescribe medications have the potential to cause significant harm.
- **Avoid scenarios where use or misuse of the system could have a consequential impact on life opportunities or legal status.** Examples include scenarios where the AI system could affect an individual's legal status, legal rights, or their access to credit, education, employment, healthcare, housing, insurance, social welfare benefits, services, opportunities, or the terms on which they're provided.
- **Avoid high stakes scenarios that could lead to harm.** The models hosted by Azure OpenAI service reflect certain societal views, biases, and other undesirable content present in the training data or the examples provided in the prompt. As a result, we caution against using the models in high-stakes scenarios where unfair, unreliable, or offensive behavior might be extremely costly or lead to harm.
- **Carefully consider use cases in high stakes domains or industry:** Examples include but are not limited to healthcare, medicine, finance, or legal.
- **Carefully consider well-scoped chatbot scenarios.** Limiting the use of the service in chatbots to a narrow domain reduces the risk of generating unintended or undesirable responses.
- **Carefully consider all generative use cases.** Content generation scenarios may be more likely to produce unintended outputs and these scenarios require careful consideration and mitigations.

Limitations

When it comes to large-scale natural language models, vision models, and speech models, there are fairness and responsible AI issues to consider. People use language and images to describe the world and to express their beliefs, assumptions, attitudes, and values. As a result, publicly available text and image data typically used to train large-scale natural language processing and image generation models contains societal biases relating to race, gender, religion, age, and other groups of people, as well as other undesirable content. Similarly, speech models can exhibit different levels of accuracy across different demographic groups and languages. These societal biases are reflected in the distributions of words, phrases, and syntactic structures.

Technical limitations, operational factors, and ranges

⊗ Caution

Be advised that this section contains illustrative examples which include terms and language that some individuals might find offensive.

Large-scale natural language, image, and speech models trained with such data can potentially behave in ways that are unfair, unreliable, or offensive, in turn causing harms. Some of the ways are listed here. We emphasize that these types of harms are not mutually exclusive. A single model can exhibit more than one type of harm, potentially relating to multiple different groups of people. For example:

- **Allocation:** These models can be used in ways that lead to unfair allocation of resources or opportunities. For example, automated résumé screening systems can withhold employment opportunities from one gender if they are trained on résumé data that reflects the existing gender imbalance in a particular industry. Or the DALL-E models could be used to create imagery in the style of a known artist, which could affect the value of the artist's work or the artist's life opportunities. GPT4 Turbo with Vision model could be used to identify individual behaviors and patterns that might have negative impacts on life opportunities.
- **Quality of service:** The Azure OpenAI models are trained primarily on English text and images with English text descriptions. Languages other than English will experience worse performance. English language varieties with less representation in the training data might experience worse performance than standard American English. The publicly available images used to train the DALL-E models might reinforce public bias and other undesirable content. The DALL-E models are also unable to consistently generate comprehensible text at this time. Speech models

might introduce other limitations, for example, translations using the Whisper model in Azure OpenAI are limited to English output only. Broadly speaking, with Speech-to-Text models, be sure to properly specify a language (or locale) for each audio input to improve accuracy in transcription. Additionally, acoustic quality of the audio input, non-speech noise, overlapped speech, vocabulary, accents, and insertion errors might also affect the quality of your transcription or translation.

- **Stereotyping:** These models can reinforce stereotypes. For example, when translating "He is a nurse" and "She is a doctor" into a genderless language such as Turkish and then back into English, many machine translation systems yield the stereotypical (and incorrect) results of "She is a nurse" and "He is a doctor." With DALL-E, when generating an image based on the prompt "Fatherless children," the model could generate images of Black children only, reinforcing harmful stereotypes that might exist in publicly available images. The GPT-4 Turbo with Vision model might also reinforce stereotypes based on the contents of the input image, by relying on components of the image and making assumptions that might not always be true.
- **Demeaning:** The natural language and vision models in the Azure OpenAI service can demean people. For example, an open-ended content generation system with inappropriate or insufficient mitigations might produce content that is offensive or demeaning to a particular group of people.
- **Overrepresentation and underrepresentation:** The natural language and vision models in the Azure OpenAI service can over- or under-represent groups of people, or even erase their representation entirely. For example, if text prompts that contain the word "gay" are detected as potentially harmful or offensive, this identification could lead to the underrepresentation or even erasure of legitimate image generations by or about the LGBTQIA+ community.
- **Inappropriate or offensive content:** The natural language and vision models in the Azure OpenAI service can produce other types of inappropriate or offensive content. Examples include the ability to generate text that is inappropriate in the context of the text or image prompt; the ability to create images that potentially contain harmful artifacts such as hate symbols; images that illicit harmful connotations; images that relate to contested, controversial, or ideologically polarizing topics; images that are manipulative; images that contain sexually charged content that is not caught by sexual-related content filters; and images that relate to sensitive or emotionally charged topics. For example, a well-intentioned text prompt aimed to create an image of the New York skyline with clouds and airplanes flying over it might unintentionally generate images that illicit sentiments related to the events surrounding 9/11.
- **Disinformation and misinformation about sensitive topics:** Because DALL-E 2 and DALL-E 3 are powerful image generation models, they can be used to produce

disinformation and misinformation that can be harmful. For example, a user could prompt the model to generate an image of a political leader engaging in activity of a violent or sexual (or simply inaccurate) nature that might lead to consequential harms, including but not limited to public protests, political change, or fake news. The GPT-4 Turbo with Vision model could also be used in a similar vein. The model might reinforce disinformation or misinformation about sensitive topics if the prompt contains such information without mitigation.

- **Information reliability:** Language and vision model responses can generate nonsensical content or fabricate content that might sound reasonable but is inaccurate with respect to external validation sources. Even when drawing responses from trusted source information, responses might misrepresent that content. Transcriptions or translations might result in inaccurate text.
- **False information:** Azure OpenAI does not fact-check or verify content that is provided by customers or users. Depending on how you have developed your application, it might produce false information unless you have built in mitigations (see [Best practices for improving system performance](#)).

Risks and limitations of fine-tuning

Fine-tuning models on Azure OpenAI can improve their performance and accuracy on specific tasks and domains, but it can also introduce new risks and limitations that customers should be aware of. Some of these risks and limitations are:

- **Data quality and representation:** The quality and representativeness of the data used for fine-tuning can affect the model's behavior and outputs. If the data is noisy, incomplete, outdated, or if it contains harmful content like stereotypes, the model can inherit these issues and produce inaccurate or harmful results. For example, if the data contains gender stereotypes, the model can amplify them and generate sexist language. Customers should carefully select and pre-process their data to ensure that it is relevant, diverse, and balanced for the intended task and domain.
- **Model robustness and generalization:** The model's ability to handle diverse and complex inputs and scenarios can decrease after fine-tuning, especially if the data is too narrow or specific. The model can overfit to the data and lose some of its general knowledge and capabilities. For example, if the data is only about sports, the model can struggle to answer questions or generate text about other topics. Customers should evaluate the model's performance and robustness on a variety of inputs and scenarios and avoid using the model for tasks or domains that are outside its scope.
- **Regurgitation:** While your training data is not available to Microsoft or any third-party customers, poorly fine-tuned models may regurgitate, or directly repeat,

training data. Customers are responsible for removing any PII or otherwise protected information from their training data and should assess their fine-tuned models for over-fitting or otherwise low-quality responses. To avoid regurgitation, customers are encouraged to provide large and diverse datasets.

- **Model transparency and explainability:** The model's logic and reasoning can become more opaque and difficult to understand after fine-tuning, especially if the data is complex or abstract. A fine-tuned model can produce outputs that are unexpected, inconsistent, or contradictory, and customers may not be able to explain how or why the model arrived at those outputs. For example, if the data is about legal or medical terms, the model can generate outputs that are inaccurate or misleading, and customers may not be able to verify or justify them. Customers should monitor and audit the model's outputs and behavior and provide clear and accurate information and guidance to the end-users of the model.

System performance

In many AI systems, performance is often defined in relation to accuracy—that is, how often the AI system offers a correct prediction or output. With large-scale natural language models and vision models, two different users might look at the same output and have different opinions of how useful or relevant it is, which means that performance for these systems must be defined more flexibly. Here, we broadly consider performance to mean that the application performs as you and your users expect, including not generating harmful outputs.

Azure OpenAI service can support a wide range of applications like search, classification, code generation, image generation, and image understanding, each with different performance metrics and mitigation strategies. There are several steps you can take to mitigate some of the concerns listed under "Limitations" and to improve performance. Other important mitigation techniques are outlined in the section [Evaluating and integrating Azure OpenAI for your use](#).

Best practices for improving system performance

- **Show and tell when designing prompts.** With natural language models and speech models, make it clear to the model what kind of outputs you expect through instructions, examples, or a combination of the two. If you want the model to rank a list of items in alphabetical order or to classify a paragraph by sentiment, show the model that is what you want.
 - **Prompts for the Whisper model in Azure OpenAI service** can help improve model outputs. The following best practices will help you create prompts that best fit your scenario and needs.

- Consider including a prompt to instruct the model to correct specific words or acronyms that the model often misrecognizes in the audio.
 - To preserve the context of a file that was split into segments, you might prompt the model with the transcript of the preceding segment. This prompt will make the transcript more accurate, because the model will use the relevant information from the previous audio. The model will only consider the final 224 tokens of the prompt and ignore anything earlier.
 - The model might skip punctuation in the transcript. Consider using a simple prompt that instructs the model to include punctuation.
 - The model might also leave out common filler words, for example, hmm, umm, etc. in the audio. If you want to keep the filler words in your transcript, you might include a prompt that contains them.
 - Some languages can be written in different ways, such as simplified or traditional Chinese. The model might not always use the writing style that a user wants for their transcript by default. Consider using a prompt to describe your preferred writing style.
- **Keep your application on topic.** Carefully structure prompts and image inputs to reduce the chance of producing undesired content, even if a user tries to use it for this purpose. For instance, you might indicate in your prompt that a chatbot only engages in conversations about mathematics and otherwise responds "I'm sorry. I'm afraid I can't answer that." Adding adjectives like "polite" and examples in your desired tone to your prompt can also help steer outputs. With DALL-E models, you might indicate in your prompt or image input that your application generates only conceptual images. It might otherwise generate a pop-up notification that explains that the application is not for photorealistic use or to portray reality. Consider nudging users toward acceptable queries and image inputs, either by listing such examples up front or by offering them as suggestions upon receiving an off-topic request. Consider training a classifier to determine whether an input (prompt or image) is on topic or off topic.
- **Provide quality data.** With text and code models, if you are trying to build a classifier or get the model to follow a pattern, make sure that there are enough examples. Be sure to proofread your examples—the model is usually capable of processing basic spelling mistakes and giving you a response, but it also might assume errors are intentional which could affect the response. Providing quality data also includes giving your model reliable data to draw responses from in chat and question answering systems.
- **Provide trusted data.** Retrieving or uploading untrusted data into your systems could compromise the security of your systems or applications. To mitigate these risks in your applicable applications (including applications using the Assistants API), we recommend logging and monitoring LLM interactions (inputs/outputs) to

detect and analyze potential prompt injections, clearly delineating user input to minimize risk of prompt injection, restricting the LLM's access to sensitive resources, limiting its capabilities to the minimum required, and isolating it from critical systems and resources. Learn about additional mitigation approaches in [Security guidance for Large Language Models | Microsoft Learn](#).

- **Configure parameters to improve accuracy or groundedness of responses.** Augmenting prompts with data retrieved from trusted sources – such as by using the Azure OpenAI "on your data" feature – can reduce, but not completely eliminate, the likelihood of generating inaccurate responses or false information. Steps you can take to further improve the accuracy of responses include carefully selecting the trusted and relevant data source and configuring custom parameters such as "strictness", "limit responses to data content" and "number of retrieved documents to be considered" as appropriate to your use cases or scenarios. Learn more about configuring these settings for [Azure OpenAI on Your Data](#).
- **Measure model quality.** As part of general model quality, consider measuring and improving fairness-related metrics and other metrics related to responsible AI in addition to traditional accuracy measures for your scenario. Consider resources like this checklist when you measure the fairness of the system. These measurements come with limitations, which you should acknowledge and communicate to stakeholders along with evaluation results.
- **Limit the length, structure, and rate of inputs and outputs.** Restricting the length or structure of inputs and outputs can increase the likelihood that the application will stay on task and mitigate, at least in part, any potentially unfair, unreliable, or offensive behaviour. Other options to reduce the risk of misuse include (i) restricting the source of inputs (for example, limiting inputs to a particular domain or to authenticated users rather than being open to anyone on the internet) and (ii) implementing usage rate limits.
- **Encourage human review of outputs prior to publication or dissemination.** With generative AI, there is potential for generating content that might be offensive or not related to the task at hand, even with mitigations in place. To ensure that the generated output meets the task of the user, consider building ways to remind users to review their outputs for quality prior to sharing widely. This practice can reduce many different harms, including offensive material, disinformation, and more.
- **Implement additional scenario-specific mitigations.** Refer to the mitigations outlined in [Evaluating and integrating Azure OpenAI for your use](#) including content moderation strategies. These recommendations do not represent every mitigation that might be required for your application, but they point to the general minimum baseline we check for when approving use cases for Azure OpenAI Service.

Best practices and recommendations for fine tuning

To mitigate the risks and limitations of fine-tuning models on Azure OpenAI, we recommend customers to follow some best practices and guidelines, such as:

- **Data selection and preprocessing:** Customers should carefully select and preprocess their data to ensure that it is relevant, diverse, and balanced for the intended task and domain. Customers should also remove or anonymize any sensitive or personal information from the data, such as names, addresses, or email addresses, to protect the privacy and security of the data subjects. Customers should also check and correct any errors or inconsistencies in the data, such as spelling, grammar, or formatting, to improve the data quality and readability.
- **Include a system message in your training data** for chat-completion formatted models, to steer your responses, and use that same system message when using your fine-tuned model for inferencing. Leaving the system message blank tends to produce low-accuracy fine-tuned models, and forgetting to include the same system message when inferencing may result in the fine-tuned model reverting to the behavior of the base model.
- **Model evaluation and testing:** Customers should evaluate and test the fine-tuned model's performance and robustness on a variety of inputs and scenarios and compare it with the original model and other baselines. Customers should also use appropriate metrics and criteria to measure the model's accuracy, reliability, and fairness, and to identify any potential errors or biases in the model's outputs and behavior.
- **Model documentation and communication:** Customers should document and communicate the model's purpose, scope, limitations, and assumptions, and provide clear and accurate information and guidance to the end-users of the model.

Evaluating and integrating Azure OpenAI natural language and vision models for your use

Text, code, and fine-tuned models

For additional information on how to evaluate and integrate these models responsibly, please see the [RAI Overview document](#).

Learn more about responsible AI

- Microsoft AI principles [↗](#)
- Microsoft responsible AI resources [↗](#)
- Microsoft Azure Learning courses on responsible AI

Learn more about Azure OpenAI

- Limited access to Azure OpenAI Service - [Azure AI services | Microsoft Learn](#)
- Code of Conduct for the Azure OpenAI Service [| Microsoft Learn](#)
- Data, privacy, and security for Azure OpenAI Service - [Azure AI services | Microsoft Learn](#)

Limited access to Azure OpenAI Service

Article • 11/03/2023

As part of Microsoft's commitment to responsible AI, we are designing and releasing Azure OpenAI Service with the intention of protecting the rights of individuals and society and fostering transparent human-computer interaction. For this reason, we currently limit the access and use of Azure OpenAI, including limiting access to the ability to modify content filters and/or abuse monitoring.

Registration process

Azure OpenAI requires registration and is currently only available to approved enterprise customers and partners. Customers who wish to use Azure OpenAI are required to submit [a registration form ↗](#).

Customers must attest to any and all use cases for which they will use the service (the use cases from which customers may select will populate in the form after selection of the desired model(s) in Question 22 in the initial registration form). Customers who wish to add additional use cases after initial onboarding must submit the additional use cases using [this form ↗](#). The use of Azure OpenAI is limited to use cases that have been selected in a registration form. Microsoft may require customers to re-verify this information. Read more about example use cases and use cases to avoid [here](#).

Customers who wish to modify content filters and modify abuse monitoring after they have onboarded to the service are subject to additional eligibility criteria and scenario restrictions. At this time, modified content filters and/or modified abuse monitoring for Azure OpenAI Service are only available to managed customers and partners working with Microsoft account teams and have additional use case restrictions. Customers meeting these requirements can register for:

- [Modified content filters ↗](#)
- [Modified abuse monitoring ↗](#)

Access to the Azure OpenAI Service is subject to Microsoft's sole discretion based on eligibility criteria and a vetting process, and customers must acknowledge that they have read and understand the Azure terms of service for Azure OpenAI Service.

Azure OpenAI Service is made available to customers under the terms governing their subscription to Microsoft Azure Services, including the Azure OpenAI section of the [Microsoft Product Terms ↗](#). Please review these terms carefully as they contain important conditions and obligations governing your use of Azure OpenAI Service.

Important links

- [Register to use Azure OpenAI ↗](#)
- [Add additional use cases ↗ \(if needed\)](#)
- [Register to modify content filtering ↗ \(if needed\)](#)
- [Register to modify abuse monitoring ↗ \(if needed\)](#)

Help and support

FAQ about Limited Access can be found [here](#). If you need help with Azure OpenAI, find support [here](#). Report abuse of Azure OpenAI [here ↗](#).

Report problematic content to cscraireport@microsoft.com.

See also

- [Code of conduct for Azure OpenAI Service integrations](#)
- [Transparency note for Azure OpenAI Service](#)
- [Characteristics and limitations for Azure OpenAI Service](#)
- [Data, privacy, and security for Azure OpenAI Service](#)

Code of conduct for Azure OpenAI Service

Article • 02/09/2024

The following Code of Conduct defines the requirements that all Azure OpenAI Service implementations must adhere to in good faith. This code of conduct is in addition to the Acceptable Use Policy in the [Microsoft Online Services Terms](#).

Access requirements

Azure OpenAI Service is a Limited Access service that requires registration and is only available to approved enterprise customers and partners. Customers who wish to use this service are required to [register through this form](#). To learn more, see [Limited Access to Azure OpenAI Service](#).

Responsible AI mitigation requirements

Integrations with Azure OpenAI Service must, as appropriate for the application and circumstances:

- Implement meaningful human oversight
- Implement technical and operational measures to detect fraudulent user behavior in account creation and during use.
- Implement strong technical limits on inputs and outputs to reduce the likelihood of misuse beyond the application's intended purpose
- Test applications thoroughly to find and mitigate undesirable behaviors
- Establish feedback channels
- Implement additional scenario-specific mitigations

To learn more, see the [Azure OpenAI transparency note](#).

Integrations with Azure OpenAI Service must not:

- be used in any way that violates Microsoft's [Acceptable Use Policy](#), including but not limited to any use prohibited by law, regulation, government order, or decree, or any use that violates the rights of others;

- be used in any way that is inconsistent with this code of conduct, including the Limited Access requirements, the Responsible AI mitigation requirements, and the Content requirements;
- exceed the use case(s) you identified to Microsoft in connection with your request to use the service;
- interact with individuals under the age of consent in any way that could result in exploitation or manipulation or is otherwise prohibited by law or regulation;
- generate or interact with content prohibited in this Code of Conduct;
- be presented alongside or monetize content prohibited in this Code of Conduct;
- make decisions without appropriate human oversight if your application may have a consequential impact on any individual's legal position, financial position, life opportunities, employment opportunities, human rights, or result in physical or psychological injury to an individual;
- infer protected characteristics about people or personally identifiable information without their explicit consent unless if used in a lawful manner by a law enforcement entity, court, or government official subject to judicial oversight in a jurisdiction that maintains a fair and independent judiciary;
- be used for unlawful tracking, stalking, or harassment of a person;
- be used to identify or verify individual identities based on media containing people's faces or otherwise physical, biological, or behavioral characteristics, or as otherwise prohibited in this Code of Conduct;
- be used for chatbots that (i) are erotic, romantic, or used for companionship purposes, or which are otherwise prohibited by this Code of Conduct; (ii) are personas of specific people without their explicit consent; (iii) claim to have special wisdom/insight/knowledge, unless very clearly labeled as being for entertainment purposes only; or (iv) enable end users to create their own chatbots without oversight.
- be used to infer gender or age from images of people, or
- attempt to infer people's emotional states from their facial expressions or facial movements; or
- without the individual's valid consent, be used for ongoing surveillance or real-time or near real-time identification or persistent tracking of the individual.

Content requirements

We prohibit the use of our service for processing content or generating content that can inflict harm on individuals or society. Our content policies are intended to improve the safety of our platform

These content requirements apply to the output of all models developed by OpenAI and hosted in Azure OpenAI, such as GPT-3, GPT-4, GPT-4 Turbo with Vision, Codex models,

DALL·E 2, DALL·E 3, and Whisper, and includes content provided as input to the service and content generated as output from the service.

Exploitation and Abuse

Child sexual exploitation and abuse

Azure OpenAI Service prohibits content that describes, features, or promotes child sexual exploitation or abuse, whether or not prohibited by law. This includes sexual content involving a child or that sexualizes a child.

Grooming

Azure OpenAI Service prohibits content that describes or is used for purposes of grooming of children. Grooming is the act of an adult building a relationship with a child for the purposes of exploitation, especially sexual exploitation. This includes communicating with a child for the purpose of sexual exploitation, trafficking, or other forms of exploitation.

Non-consensual intimate content

Azure OpenAI Service prohibits content that describes, features, or promotes non-consensual intimate activity.

Sexual solicitation

Azure OpenAI Service prohibits content that describes, features, or promotes, or is used for, purposes of solicitation of commercial sexual activity and sexual services. This includes encouragement and coordination of real sexual activity.

Trafficking

Azure OpenAI Service prohibits content describing or used for purposes of human trafficking. This includes the recruitment of individuals, facilitation of transport, and payment for, and the promotion of, exploitation of people such as forced labor, domestic servitude, sexual slavery, forced marriages, and forced medical procedures.

Suicide and Self-Injury

Azure OpenAI Service prohibits content that describes, praises, supports, promotes, glorifies, encourages and/or instructs individual(s) on self-injury or to take their life.

Facial recognition

Azure OpenAI Service prohibits identification or verification of individual identities using media containing people's faces by any user, including by or for state or local police in the United States.

Facial analysis

Azure OpenAI Service prohibits the inferencing of a person's emotional state based on facial expressions. This includes inferring internal emotions such as anger, disgust, happiness, sadness, surprise, fear or other terms commonly used to describe the emotional state of a person. Azure OpenAI Service also prohibits the inference of gender, age, or facial expressions, or inference of the presence of facial hair, hair, or makeup.

Violent Content and Conduct

Graphic violence and gore

Azure OpenAI Service prohibits content that describes, features, or promotes graphic violence or gore.

Terrorism and Violent Extremism

Azure OpenAI Service prohibits content that depicts an act of terrorism; praises, or supports a terrorist organization, terrorist actor, or violent terrorist ideology; encourages terrorist activities; offers aid to terrorist organizations or terrorist causes; or aids in recruitment to a terrorist organization.

Violent Threats, Incitement, and Glorification of Violence

Azure OpenAI Service prohibits content advocating or promoting violence toward others through violent threats or incitement.

Harmful Content

Hate speech and discrimination

Azure OpenAI Service prohibits content that attacks, denigrates, intimidates, degrades, targets, or excludes individuals or groups on the basis of traits such as actual or perceived race, ethnicity, national origin, gender, gender identity, sexual orientation, religious affiliation, age, disability status, caste, or any other characteristic that is associated with systemic prejudice or marginalization.

Bullying and harassment

Azure OpenAI Service prohibits content that targets individual(s) or group(s) with threats, intimidation, insults, degrading or demeaning language or images, promotion of physical harm, or other abusive behavior such as stalking.

Deception, disinformation, and inauthentic activity

Azure OpenAI Service prohibits content that is intentionally deceptive and likely to adversely affect the public interest, including deceptive or untrue content relating to health, safety, election integrity, or civic participation. Azure OpenAI Service also prohibits inauthentic interactions, such as fake accounts, automated inauthentic activity, impersonation to gain unauthorized information or privileges, and claims to be from any person, company, government body, or entity without explicit permission to make that representation.

Active malware or exploits

Content that supports unlawful active attacks or malware campaigns that cause technical harms, such as delivering malicious executables, organizing denial of service attacks, or managing command and control servers.

Additional content policies

We prohibit the use of our Azure OpenAI Service for scenarios in which the system is likely to generate undesired content due to limitations in the models or scenarios in which the system cannot be applied in a way that properly manages potential negative consequences to people and society. Without limiting the foregoing restriction, Microsoft reserves the right to revise and expand the above Content requirements to address specific harms to people and society.

This includes prohibiting content that is sexually graphic, including consensual pornographic content and intimate descriptions of sexual acts.

We may at times limit our service's ability to respond to particular topics, such as probing for personal information or seeking opinions on sensitive topics or current events.

We prohibit the use of Azure OpenAI Service for activities that significantly harm other individuals, organizations, or society, including but not limited to use of the service for purposes in conflict with the applicable [Azure Legal Terms](#) and the [Microsoft Product Terms](#).

Report abuse

If you suspect that Azure OpenAI Service is being used in a manner that is abusive or illegal, infringes on your rights or the rights of other people, or violates these policies, you can report it at the [Report Abuse Portal](#).

See also

- [Limited access to Azure OpenAI Service](#)
- [Transparency note for Azure OpenAI Service](#)
- [Data, privacy, and security for Azure OpenAI Service](#)

Data, privacy, and security for Azure OpenAI Service

Article • 02/23/2024

This article provides details regarding how data provided by you to the Azure OpenAI service is processed, used, and stored. Azure OpenAI stores and processes data to provide the service and to monitor for uses that violate the applicable product terms. Please also see the [Microsoft Products and Services Data Protection Addendum](#), which governs data processing by the Azure OpenAI Service except as otherwise provided in the applicable [Product Terms](#).

ⓘ Important

Your prompts (inputs) and completions (outputs), your embeddings, and your training data:

- are NOT available to other customers.
- are NOT available to OpenAI.
- are NOT used to improve OpenAI models.
- are NOT used to improve any Microsoft or 3rd party products or services.
- are NOT used for automatically improving Azure OpenAI models for your use in your resource (The models are stateless, unless you explicitly fine-tune models with your training data).
- Your fine-tuned Azure OpenAI models are available exclusively for your use.

The Azure OpenAI Service is fully controlled by Microsoft; Microsoft hosts the OpenAI models in Microsoft's Azure environment and the Service does NOT interact with any services operated by OpenAI (e.g. ChatGPT, or the OpenAI API).

What data does the Azure OpenAI Service process?

Azure OpenAI processes the following types of data:

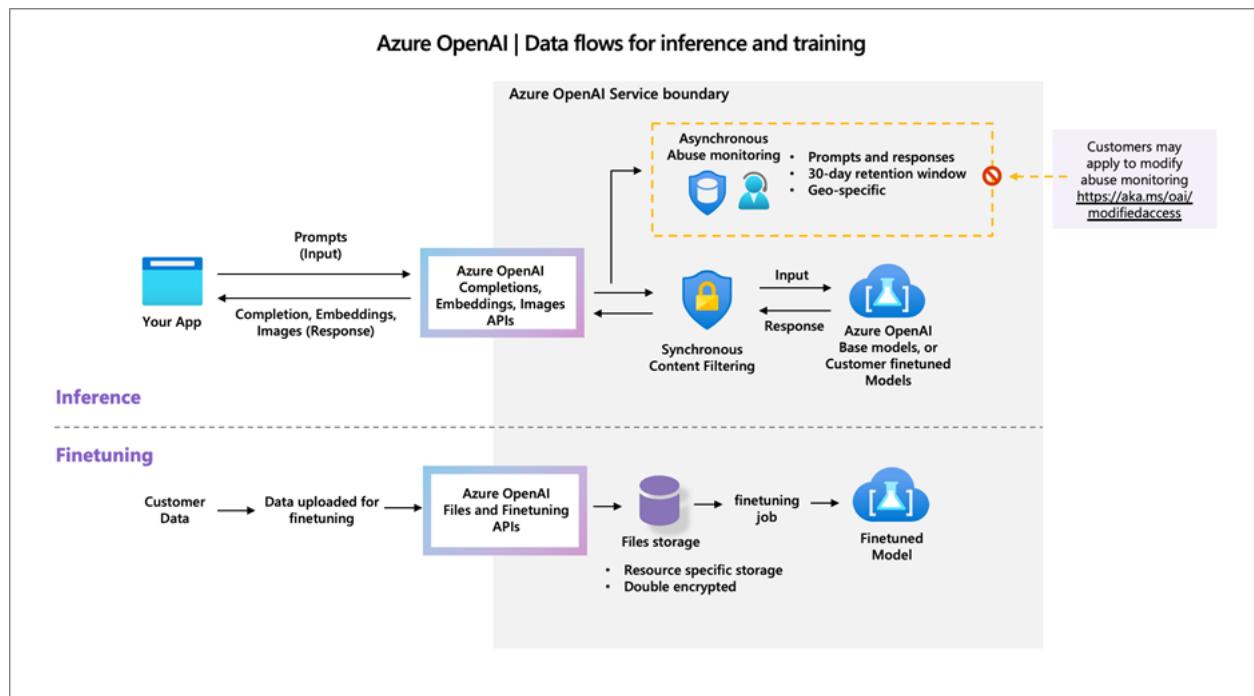
- **Prompts and generated content.** Prompts are submitted by the user, and content is generated by the service, via the completions, chat completions, images and embeddings operations.

- **Augmented data included with prompts.** When using the "on your data" feature, the service retrieves relevant data from a configured data store and augments the prompt to produce generations that are grounded with your data.
- **Training & validation data.** You can provide your own training data consisting of prompt-completion pairs for the purposes of [fine-tuning an OpenAI model](#).

How does the Azure OpenAI Service process data?

The diagram below illustrates how your data is processed. This diagram covers three different types of processing:

1. How the Azure OpenAI Service processes your prompts to generate content (including when additional data from a connected data source is added to a prompt using Azure OpenAI on your data).
2. How the Azure OpenAI Service creates a fine-tuned (custom) model with your training data.
3. How the Azure OpenAI Service and Microsoft personnel analyze prompts, completions and images for harmful content and for patterns suggesting the use of the service in a manner that violates the Code of Conduct or other applicable product terms



As depicted in the diagram above, managed customers may [apply to modify abuse monitoring](#).

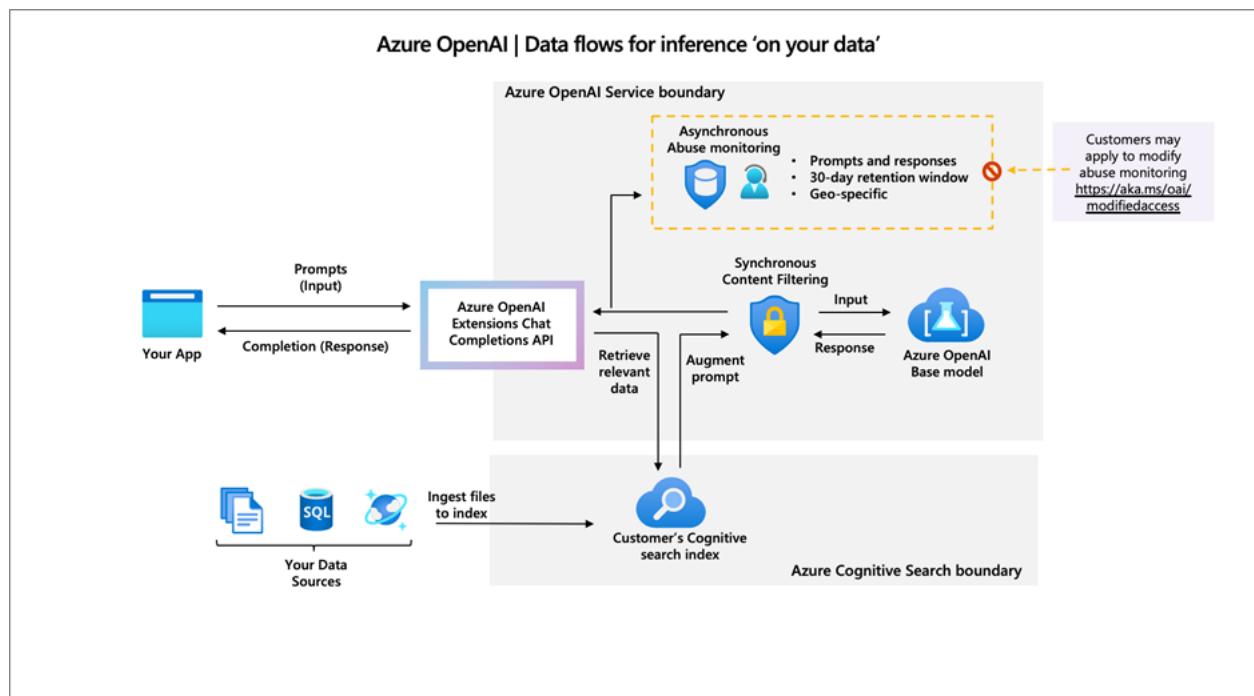
Generating completions, images or embeddings

Models (base or fine-tuned) deployed in your resource process your input prompts and generate responses with text, images or embeddings. Prompts and responses are processed within the customer-specified [geography](#), but may be processed between regions within the geography for operational purposes (including performance and capacity management). The service is configured to synchronously evaluate the prompt and completion data in real time to check for harmful content types and stops generating content that exceeds the configured thresholds. Learn more at [Azure OpenAI Service content filtering](#).

The models are stateless: no prompts or generations are stored in the model. Additionally, prompts and generations are not used to train, retrain, or improve the base models.

Augmenting prompts with data retrieved from your data sources to "ground" the generated results

The Azure OpenAI "on your data" feature lets you connect data sources to ground the generated results with your data. The data remains stored in the data source and location you designate. **No data is copied into the Azure OpenAI service.** When a user prompt is received, the service retrieves relevant data from the connected data source and augments the prompt. The model processes this augmented prompt and the generated content is returned as described above.



As depicted in the diagram above, managed customers may apply to [modify abuse monitoring](#).

Creating a customized (fine-tuned) model with your data:

Customers can upload their training data to the service to fine tune a model. Uploaded training data is stored in the Azure OpenAI resource in the customer's Azure tenant.

Training data and fine-tuned models:

- Are available exclusively for use by the customer.
- Are stored within the same region as the Azure OpenAI resource.
- Can be double [encrypted at rest](#) (by default with Microsoft's AES-256 encryption and optionally with a customer managed key).
- Can be deleted by the customer at any time.

Training data uploaded for fine-tuning is not used to train, retrain, or improve any Microsoft or 3rd party base models.

Preventing abuse and harmful content generation

To reduce the risk of harmful use of the Azure OpenAI Service, the Azure OpenAI Service includes both content filtering and abuse monitoring features. To learn more about [content filtering](#), see Azure OpenAI Service content filtering. To learn more about abuse monitoring, see [abuse monitoring](#).

Content filtering occurs synchronously as the service processes prompts to generate content as described above and [here](#). No prompts or generated results are stored in the content classifier models, and prompts and results are not used to train, retrain, or improve the classifier models.

Azure OpenAI abuse monitoring detects and mitigates instances of recurring content and/or behaviors that suggest use of the service in a manner that may violate the [code of conduct](#) or other applicable product terms. To detect and mitigate abuse, Azure OpenAI stores all prompts and generated content securely for up to thirty (30) days. (No prompts or completions are stored if the customer is approved for and elects to configure abuse monitoring off, as described below.)

The data store where prompts and completions are stored is logically separated by customer resource (each request includes the resource ID of the customer's Azure OpenAI resource). A separate data store is located in each [region](#) in which the Azure OpenAI Service is available, and a customer's prompts and generated content are stored in the Azure region where the customer's Azure OpenAI service resource is deployed, within the Azure OpenAI service boundary. Human reviewers assessing potential abuse can access prompts and completions data only when that data has been flagged by the abuse monitoring system. The human reviewers are authorized Microsoft employees

who access the data via point wise queries using request IDs, Secure Access Workstations (SAWs), and Just-In-Time (JIT) request approval granted by team managers. For Azure OpenAI Service deployed in the European Economic Area, the authorized Microsoft employees are located in the European Economic Area.

How can customers get an exemption from abuse monitoring and human review?

Some customers may want to use the Azure OpenAI Service for a use case that involves the processing of sensitive, highly confidential, or legally-regulated input data but where the likelihood of harmful outputs and/or misuse is low. These customers may conclude that they do not want or do not have the right to permit Microsoft to process such data for abuse detection, as described above, due to their internal policies or applicable legal regulations. To address these concerns, Microsoft allows customers who meet additional Limited Access eligibility criteria and attest to specific use cases to apply to modify the Azure OpenAI content management features by completing [this form](#).

If Microsoft approves a customer's request to modify abuse monitoring, then Microsoft does not store any prompts and completions associated with the approved Azure subscription for which abuse monitoring is configured off. In this case, because no prompts and completions are stored at rest in the Service Results Store, the human review process is not possible and is not performed. See [Abuse monitoring](#) for more information.

How can a customer verify if data storage for abuse monitoring is off?

There are two ways for customers, once approved to turn off abuse monitoring, to verify that data storage for abuse monitoring has been turned off in their approved Azure subscription:

- Using the Azure portal, or
- Azure CLI (or any management API).

Note

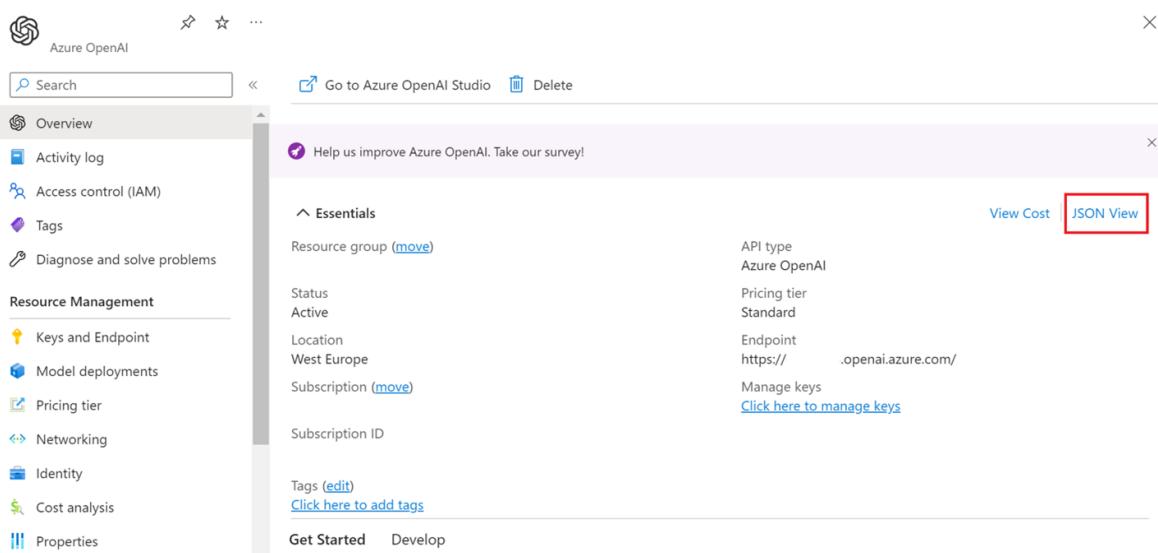
The value of "false" for the "ContentLogging" attribute appears only if data storage for abuse monitoring is turned off. Otherwise, this property will not appear in either Azure portal or Azure CLI's output.

Prerequisites

1. Sign into Azure
2. Select the Azure Subscription which hosts the Azure OpenAI Service resource.
3. Navigate to the **Overview** page of the Azure OpenAI Service resource.

Logging status verification using the Azure portal:

1. Go to the resource Overview page
2. Click on the **JSON view** link on the top right corner as shown in the image below.



There will be a value in the Capabilities list called "ContentLogging" which will appear and be set to FALSE when logging for abuse monitoring is off.

```
JSON
{
  "name": "ContentLogging",
  "value": "false"
}
```

Logging status verification using the Azure CLI (or other management API):

Execute the following command in Azure CLI to see the same JSON data as shown in the Azure portal above.

```
Azure CLI
az cognitiveservices account show -n resource\_name -g resource \_group
```

To learn more about Microsoft's privacy and security commitments see the [Microsoft Trust Center](#).

Change log

[+] Expand table

Date	Changes
23 June 2023	Added information about data processing for new Azure on your data feature; removed information about abuse monitoring which is now available at Azure OpenAI Service abuse monitoring . Added summary note. Updated and streamlined content and updated diagrams for additional clarity. added change log

See also

- Limited access to Azure OpenAI Service
- Code of conduct for Azure OpenAI Service integrations
- Transparency note and use cases for Azure OpenAI Service
- Characteristics and limitations for Azure OpenAI Service
- Report abuse of Azure OpenAI Service through the [Report Abuse Portal](#)
- Report problematic content to cscraireport@microsoft.com

Customer Copyright Commitment Required Mitigations

Article • 01/01/2024

ⓘ Note

The requirements described below apply only to customers using Azure OpenAI Service and other Covered Products with configurable Metaprompts or other safety systems ("Configurable GAI Services"). They do not apply to customers using other Covered Products including Copilots with safety systems that are fixed.

The Customer Copyright Commitment ("CCC") is a provision in the Microsoft Product Terms that describes Microsoft's obligation to defend customers against certain third-party intellectual property claims relating to Output Content. Beginning December 1, 2023, the CCC will be updated as follows: For Azure OpenAI Service and any Configurable GAI Service, Customer also must have implemented all mitigations required by the Azure OpenAI Service documentation in the offering that delivered the Output Content that is the subject of the claim. The required mitigations to maintain CCC coverage are set forth below.

This page describes only the required mitigations necessary to maintain CCC coverage for Azure OpenAI Service and Configurable GAI Services. It is not an exhaustive list of requirements or mitigations required to use Azure OpenAI Service (or Configurable GAI Services) responsibly in compliance with the documentation. Azure OpenAI Service customers must comply with the [Code of Conduct](#) at all times.

The section "Required Mitigations for GitHub Offerings" are the only requirements that apply to GitHub offerings, and separately took effect on November 1, 2023. The other mitigations below will take effect on December 1, 2023, when the CCC update takes effect. For new Configurable GAI Services, features, models, or use cases, new CCC requirements will be posted and take effect at or following the launch of such Configurable GAI Service, feature, model, or use case. Otherwise, for future updates released after December 1, 2023, Customers will have six months from the time of publication on this page to implement any new mitigations required to maintain coverage under the CCC. If a customer tenders a claim for defense, the customer will be required to demonstrate compliance with all relevant requirements, both on this page and as listed in the Product Terms.

Universal Required Mitigations

Universal required mitigations must be implemented to maintain CCC coverage for all offerings delivering Output Content from Azure OpenAI Service and Configurable GAI Services, with the exception of GitHub Offerings. The requirements are set forth here:

Azure OpenAI Service & Configurable GAI Services - Universal Required Mitigations:

[+] [Expand table](#)

Category	Required Mitigation	Effective Date
Metaprompt	The customer offering must include a metaprompt directing the model to prevent copyright infringement in its output, for example, the sample metaprompt, "To Avoid Copyright Infringements" at: System message framework and template recommendations for Large Language Models(LLMs)	December 1, 2023
Testing and Evaluation Report	The customer offering must have been subjected to evaluations (e.g., guided red teaming, systematic measurement, or other equivalent approach) by the customer using tests designed to detect the output of third-party content. Significant ongoing reproduction of third-party content determined through evaluation must be addressed. The report of results and mitigations must be retained by the customer and provided to Microsoft in the event of a claim. More information on guided red teaming is at: Red teaming large language models (LLMs) . More information on systematic measurement is at: Overview of Responsible AI practices for Azure OpenAI models - Azure AI services - Microsoft Learn .	December 1, 2023

Additional Required Mitigations Per Azure OpenAI Service Use Case

Additional required mitigations are required to maintain CCC coverage for offerings delivering Output Content from Azure OpenAI Service, depending on what use cases the customer has requested in the [Limited Access Form](#). Requirements are cumulative, meaning that the customer offering must include the required mitigations for all approved use cases. These additional requirements do not apply to Configurable GAI Services, only Azure OpenAI Service.

Not all content types can be generated by every application. The following required mitigations must be enabled for any use case described below. Azure OpenAI content

filters include protected material detection. Protected material detection can analyze both text and code. Different filters must be on depending on content type.

The required mitigations are set forth here:

Azure OpenAI Service Only - Additional Required Mitigations Per Use Case

Text and Code Use Cases:

[+] Expand table

Content type	Use Case	Category	Required Mitigation	Effective Date
Code generation	Code generation or transformation scenarios, or other open code generation scenarios	Content filters	<p>The protected material code model must be configured on in either annotate or filter mode. If choosing to use annotate mode, customer must comply with any cited license provided for Output Content that is the subject of the claim.</p> <p>The jailbreak model must be configured on in filter mode.</p>	December 1, 2023
Text generation	Journalistic content, writing assistance, or other open text generation scenarios	Content filters	<p>The protected material text model must be configured on in filter mode. The jailbreak model must be configured on in filter mode.</p>	December 1, 2023

Image generation models, OpenAI Whisper model, and all other use cases:

No additional requirements.

Required Mitigations for GitHub Offerings

The below are the only required mitigations that apply to GitHub Offerings, and separately took effect in the Product Terms on November 1, 2023.

Required Mitigations for GitHub Offerings Only

[+] Expand table

Category	Required Mitigation	Effective Date
GitHub Offerings	The Duplicate Detection filtering feature must be set to the "Block" setting. Customers can learn how to enable the Duplicate Detection filter at https://gh.io/cfb-dd .	November 1, 2023

도우미 API(미리 보기) 참조

아티클 • 2024. 02. 23.

이 문서에서는 새로운 도우미 API(미리 보기)에 대한 Python 및 REST에 대한 참조 설명서를 제공합니다. 더 자세한 단계별 지침은 [시작 지침](#)에서 제공됩니다.

도우미 만들기

HTTP

```
POST https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants?api-version=2024-02-15-preview
```

모델과 지침이 포함된 도우미를 만듭니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
model		Required	사용할 모델의 모델 배포 이름입니다.
name	문자열 또는 null	선택 사항	도우미의 이름입니다. 최대 길이는 256자입니다.
description	문자열 또는 null	선택 사항	도우미에 대한 설명입니다. 최대 길이는 512자입니다.
지침	문자열 또는 null	선택 사항	도우미가 사용하는 시스템 지침입니다. 최대 길이는 32768자입니다.
tools	배열	선택 사항	기본값은 []입니다. 도우미에서 사용하도록 설정된 도구 목록입니다. 도우미당 최대 128개의 도구가 있을 수 있습니다. 도구는 현재 <code>code_interpreter</code> 또는 <code>function</code> 형식일 수 있습니다.
file_ids	배열	선택 사항	기본값은 []입니다. 이 도우미에 첨부된 파일 ID 목록입니다. 도우미에는 최대 20개의 파일을 첨부할 수 있습니다. 파일은 만든 날짜를 기준으로 오름차순으로 정렬됩니다.
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용합니다.

이름	Type	필수	설명
			수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

도우미 개체.

도우미 만들기 요청 예

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

assistant = client.beta.assistants.create(
    instructions="You are an AI assistant that can write code to help
answer math questions",
    model=<REPLACE WITH MODEL DEPLOYMENT NAME>, # replace with model
deployment name.
    tools=[{"type": "code_interpreter"}]
)
```

도우미 파일 만들기

HTTP

POST

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}/files?api-version=2024-02-15-preview

File 을 assistant 에 첨부하여 도우미 파일을 만듭니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
assistant_id	string	Required	파일을 첨부해야 하는 도우미의 ID입니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
file_id	string	Required	도우미가 사용해야 하는 파일 ID(목적="도우미" 포함)입니다. 파일에 액세스할 수 있는 code_interpreter와 같은 도구에 유용합니다.

반환

도우미 파일 개체.

도우미 파일 요청 만들기 예

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

assistant_file = client.beta.assistants.files.create(
    assistant_id="asst_abc123",
    file_id="assistant-abc123"
)
print(assistant_file)
```

도우미 나열

HTTP

```
GET https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants?api-version=2024-02-15-preview
```

모든 도우미 목록을 반환합니다.

쿼리 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
limit	정수	선택 사항	반환할 개체 수에 대한 제한입니다. 제한 범위는 1~100이며 기본값은 20입니다.
order	string	선택 사항	개체의 Created_at 타임스탬프를 기준으로 정렬 순서입니다. 오름차순은 asc, 내림차순은 desc입니다.
after	string	선택 사항	페이지 매김에 사용되는 커서입니다. after는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 다음 페이지를 폐치 위해 후속 호출에 after=obj_foo가 포함될 수 있습니다.
before	string	선택 사항	페이지 매김에 사용되는 커서입니다. before는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 이전 페이지를 폐치 위해 후속 호출에 before=obj_foo가 포함될 수 있습니다.

반환

도우미 개체 목록

도우미 나열 예

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
```

```
)  
  
my_assistants = client.beta.assistants.list(  
    order="desc",  
    limit="20",  
)  
print(my_assistants.data)
```

도우미 파일 나열

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}/files?api-version=2024-02-15-preview

도우미 파일 목록을 반환합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
assistant_id	string	Required	파일이 속한 도우미의 ID입니다.

쿼리 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
limit	정수	선택 사항	반환할 개체 수에 대한 제한입니다. 제한 범위는 1~100이며 기본 값은 20입니다.
order	string	선택 사항	개체의 Created_at 타임스탬프를 기준으로 정렬 순서입니다. 오름 차순은 asc, 내림차순은 desc입니다. - 기본값은 desc입니다.
after	string	선택 사항	페이지 매김에 사용되는 커서입니다. after는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 다음 페이지를 페치 위해 후속 호출에 after=obj_foo가 포함될 수 있습니다.

매개 변수	Type	필수	설명
before	string	선택 사항	페이지 매김에 사용되는 커서입니다. <code>before</code> 는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 <code>obj_foo</code> 로 끝나는 100개의 개체를 받은 경우 목록의 이전 페이지를 페치 위해 후속 호출에 <code>before=obj_foo</code> 가 포함될 수 있습니다.

반환

도우미 파일 개체 목록

도우미 파일 나열 예

Python 1.x

```
Python

from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

assistant_files = client.beta.assistants.files.list(
    assistant_id="asst_abc123"
)
print(assistant_files)
```

도우미 검색

HTTP

GET

`https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}`
`?api-version=2024-02-15-preview`

도우미를 검색합니다.

경로 매개 변수

매개 변수	Type	필수	설명
assistant_id	string	Required	검색할 도우미의 ID입니다.

반환

지정된 ID와 일치하는 [도우미](#) 개체입니다.

도우미 검색 예

Python 1.x

Python

```
client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

my_assistant = client.beta.assistants.retrieve("asst_abc123")
print(my_assistant)
```

도우미 파일 검색

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}/files/{file-id}?api-version=2024-02-15-preview

도우미 파일을 검색합니다.

경로 매개 변수

매개 변수	Type	필수	설명
assistant_id	string	Required	파일이 속한 도우미의 ID입니다.
file_id	string	Required	가져오는 파일의 ID

반환

지정된 ID와 일치하는 도우미 파일 개체

도우미 파일 검색 예

Python 1.x

Python

```
client = AzureOpenAI(  
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),  
    api_version="2024-02-15-preview",  
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")  
)  
  
assistant_file = client.beta.assistants.files.retrieve(  
    assistant_id="asst_abc123",  
    file_id="assistant-abc123"  
)  
print(assistant_file)
```

도우미 수정

HTTP

POST

[https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}
?api-version=2024-02-15-preview](https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}?api-version=2024-02-15-preview)

도우미를 수정합니다.

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	Type	필수	설명
assistant_id	string	Required	파일이 속한 도우미의 ID입니다.

요청 본문

[\[+\] 테이블 확장](#)

매개 변수	Type	필수	설명
model	선택사항	선택사항	사용할 모델의 모델 배포 이름입니다.
name	문자열 또는 null	선택사항	도우미의 이름입니다. 최대 길이는 256자입니다.
description	문자열 또는 null	선택사항	도우미에 대한 설명입니다. 최대 길이는 512자입니다.
instructions	문자열 또는 null	선택사항	도우미가 사용하는 시스템 지침입니다. 최대 길이는 32768자입니다.
tools	배열	선택사항	기본값은 []입니다. 도우미에서 사용하도록 설정된 도구 목록입니다. 도우미당 최대 128개의 도구가 있을 수 있습니다. 도구는 code_interpreter 또는 함수 형식일 수 있습니다.
file_ids	배열	선택사항	기본값은 []입니다. 이 도우미에 첨부된 파일 ID 목록입니다. 도우미에는 최대 20개의 파일을 첨부할 수 있습니다. 파일은 만든 날짜를 기준으로 오름차순으로 정렬됩니다. 이전에 목록에 첨부된 파일이 목록에 표시되지 않으면 도우미에서 삭제됩니다.
metadata	map	선택사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

수정된 [도우미 개체](#).

도우미 수정 예

Python 1.x

Python

```

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

my_updated_assistant = client.beta.assistants.update(
    "asst_abc123",
    instructions="You are an HR bot, and you have access to files to
answer employee questions about company policies. Always respond with
info from either of the files.",
    name="HR Helper",
    tools=[{"type": "code-interpreter"}],
    model="gpt-4", #model = model deployment name
    file_ids=["assistant-abc123", "assistant-abc456"],
)
print(my_updated_assistant)

```

도우미 삭제

HTTP

DELETE

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}
?api-version=2024-02-15-preview

도우미를 삭제합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
assistant_id	string	Required	파일이 속한 도우미의 ID입니다.

반환

삭제 상태.

도우미 삭제 예

Python 1.x

Python

```
client = AzureOpenAI(  
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),  
    api_version="2024-02-15-preview",  
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")  
)  
  
response = client.beta.assistants.delete("asst_abc123")  
print(response)
```

도우미 파일 삭제

HTTP

DELETE

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/assistants/{assistant_id}/files/{file-id}?api-version=2024-02-15-preview

도우미 파일을 삭제합니다.

경로 매개 변수

 테이블 확장

매개 변수	Type	필수	설명
assistant_id	string	Required	파일이 속한 도우미의 ID입니다.
file_id	string	Required	삭제할 파일의 ID

반환

파일 삭제 상태

도우미 파일 삭제 예

Python 1.x

Python

```
client = AzureOpenAI(  
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),  
    api_version="2024-02-15-preview",
```

```

        azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
    )

deleted_assistant_file = client.beta.assistants.files.delete(
    assistant_id="asst_abc123",
    file_id="assistant-abc123"
)
print(deleted_assistant_file)

```

도우미 개체

[+] 테이블 확장

필드	형식	Description
<code>id</code>	string	API 엔드포인트에서 참조할 수 있는 식별자입니다.
<code>object</code>	string	항상 도우미인 개체 형식입니다.
<code>created_at</code>	정수	도우미가 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>name</code>	문자열 또는 null	도우미의 이름입니다. 최대 길이는 256자입니다.
<code>description</code>	문자열 또는 null	도우미에 대한 설명입니다. 최대 길이는 512자입니다.
<code>model</code>	string	사용할 모델 배포 이름입니다.
<code>instructions</code>	문자열 또는 null	도우미가 사용하는 시스템 지침입니다. 최대 길이는 32768자입니다.
<code>tools</code>	배열	도우미에서 사용하도록 설정된 도구 목록입니다. 도우미당 최대 128개의 도구가 있을 수 있습니다. 도구는 <code>code_interpreter</code> 또는 함수 형식일 수 있습니다.
<code>file_ids</code>	배열	이 도우미에 첨부된 파일 ID 목록입니다. 도우미에는 최대 20개의 파일을 첨부할 수 있습니다. 파일은 만든 날짜를 기준으로 오름차순으로 정렬됩니다.
<code>metadata</code>	map	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

도우미 파일 개체

필드	형식	Description
<code>id</code>	string	API 엔드포인트에서 참조할 수 있는 식별자입니다.
<code>object</code>	string	항상 <code>assistant.file</code> 인 개체 형식입니다.
<code>created_at</code>	정수	도우미 파일이 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>assistant_id</code>	string	파일이 첨부된 도우미 ID입니다.

도우미 API(미리 보기) 스레드 참조

아티클 • 2024. 03. 05.

이 문서에서는 새 도우미 API(미리 보기)에 대한 Python 및 REST에 대한 참조 설명서를 제공합니다. [시작 가이드](#)에 자세한 단계별 지침이 제공됩니다.

스레드 만들기

HTTP

```
POST https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads?api-version=2024-02-15-preview
```

스레드를 만듭니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
messages	배열	선택 사항	스레드를 시작할 메시지 목록입니다.
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 구조화된 형식으로 개체에 대한 추가 정보를 저장하는 데 유용할 수 있습니다. 키의 길이는 최대 64자이고 값은 최대 512자까지 가능합니다.

반환

[스레드 개체](#).

스레드 만들기 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
```

```
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

empty_thread = client.beta.threads.create()
print(empty_thread)
```

스레드 검색

HTTP

```
GET https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}?
api-version=2024-02-15-preview
```

스레드를 검색합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	검색할 스레드의 ID입니다.

반환

지정된 ID와 일치하는 스레드 개체입니다.

스레드 검색 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)
```

```
my_thread = client.beta.threads.retrieve("thread_abc123")
print(my_thread)
```

스레드 수정

HTTP

```
POST https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}?
api-version=2024-02-15-preview
```

스레드를 수정합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	수정할 스레드의 ID입니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 구조화된 형식으로 개체에 대한 추가 정보를 저장하는 데 유용할 수 있습니다. 키의 길이는 최대 64자이고 값은 최대 512자까지 가능합니다.

반환

지정된 ID와 일치하는 수정된 [스레드 개체](#)입니다.

스레드 수정 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI
```

```
client = AzureOpenAI(  
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),  
    api_version="2024-02-15-preview",  
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")  
)  
  
my_updated_thread = client.beta.threads.update(  
    "thread_abc123",  
    metadata={  
        "modified": "true",  
        "user": "abc123"  
    }  
)  
print(my_updated_thread)
```

스레드 삭제

HTTP

DELETE

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}?api-version=2024-02-15-preview

스레드 삭제

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	삭제할 스레드의 ID입니다.

반환

삭제 상태입니다.

스레드 삭제 요청 예제

Python 1.x

Python

```

from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

response = client.beta.threads.delete("thread_abc123")
print(response)

```

스레드 개체

[] 테이블 확장

필드	형식	Description
<code>id</code>	string	API 엔드포인트에서 참조할 수 있는 식별자입니다.
<code>object</code>	string	항상 스레드인 개체 형식입니다.
<code>created_at</code>	정수	스레드가 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>metadata</code>	map	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 구조화된 형식으로 개체에 대한 추가 정보를 저장하는 데 유용할 수 있습니다. 키의 길이는 최대 64자이고 값은 최대 512자까지 가능합니다.

Assistants API(미리 보기) 메시지 참조

아티클 • 2024. 03. 01.

이 문서에서는 새로운 Assistants API(미리 보기)에 대한 Python 및 REST에 대한 참조 설명서를 제공합니다. 더 자세한 단계별 지침은 [시작 가이드](#)에서 제공됩니다.

메시지 만들기

HTTP

POST

`https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/messages?api-version=2024-02-15-preview`

메시지 작성.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	메시지를 만들 스레드의 ID입니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
role	string	Required	메시지를 만드는 엔터티의 역할입니다. 현재는 사용자만 지원됩니다.
content	string	Required	메시지의 내용입니다.
file_ids	배열	선택 사항	메시지가 사용해야 하는 파일 ID 목록입니다. 메시지에는 최대 10 개의 파일을 첨부할 수 있습니다. 파일에 액세스하고 사용할 수 있는 retrieval 및 code_interpreter 같은 도구에 유용합니다.
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

메시지 개체

메시지 만들기 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

thread_message = client.beta.threads.messages.create(
    "thread_abc123",
    role="user",
    content="How does AI work? Explain it in simple terms.",
)
print(thread_message)
```

메시지 나열

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/messages?api-version=2024-02-15-preview

지정된 스레드에 대한 메시지 목록을 반환합니다.

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	Type	필수	설명
thread_id	string	Required	메시지가 속한 스레드의 ID입니다.

쿼리 매개 변수

이름	Type	필수	설명
limit	정수	선택 사항 - 기본값은 20입니다.	반환할 개체 수에 대한 제한입니다. 제한 범위는 1~100이며 기본값은 20입니다.
order	string	선택 사항 - 기본값은 desc입니다.	개체의 Created_at 타임스탬프를 기준으로 정렬 순서입니다. 오름 차순은 asc, 내림차순은 desc입니다.
after	string	선택 사항	페이지 매김에 사용되는 커서입니다. after는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 다음 페이지를 페치하기 위해 후속 호출에 after=obj_foo가 포함될 수 있습니다.
before	string	선택 사항	페이지 매김에 사용되는 커서입니다. before는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 이전 페이지를 페치하기 위해 후속 호출에 before=obj_foo가 포함될 수 있습니다.

반환

메시지 개체 목록입니다.

메시지 나열 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

thread_messages = client.beta.threads.messages.list("thread_abc123")
print(thread_messages.data)
```

메시지 파일 나열

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/thread/{thread_id}/messages/{message_id}/files?api-version=2024-02-15-preview

메시지 파일 목록을 반환합니다.

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	메시지와 파일이 속한 스레드의 ID입니다.
message_id	string	Required	파일이 속한 메시지의 ID입니다.

쿼리 매개 변수

[+] 테이블 확장

이름	Type	필수	설명
limit	정수	선택 사항 - 기본값은 20입니다.	반환할 개체 수에 대한 제한입니다. 제한 범위는 1~100이며 기본값은 20입니다.
order	string	선택 사항 - 기본값은 desc입니다.	개체의 Created_at 타임스탬프를 기준으로 정렬 순서입니다. 오름차순은 asc, 내림차순은 desc입니다.
after	string	선택 사항	페이지 매김에 사용되는 커서입니다. after는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 다음 페이지를 페치하기 위해 후속 호출에 after=obj_foo가 포함될 수 있습니다.
before	string	선택 사항	페이지 매김에 사용되는 커서입니다. before는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 이전 페이지를 페치하기 위해 후속 호출에 before=obj_foo가 포함될 수 있습니다.

반환

메시지 파일 개체 목록입니다.

메시지 파일 나열 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

message_files = client.beta.threads.messages.files.list(
    thread_id="thread_abc123",
    message_id="msg_abc123"
)
print(message_files)
```

메시지 검색

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/messages/{message_id}?api-version=2024-02-15-preview

메시지 파일을 검색합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	메시지가 속한 스레드의 ID입니다.
message_id	string	Required	검색할 메시지의 ID입니다.

반환

지정된 ID와 일치하는 [메시지](#) 개체입니다.

메시지 검색 요청 예제

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

message = client.beta.threads.messages.retrieve(
    message_id="msg_abc123",
    thread_id="thread_abc123",
)
print(message)
```

메시지 파일 검색

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/messages/{message_id}/files/{file_id}?api-version=2024-02-15-preview

메시지 파일을 검색합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	메시지와 파일이 속한 스레드의 ID입니다.
message_id	string	Required	파일이 속한 메시지의 ID입니다.
file_id	string	Required	검색할 파일의 ID입니다.

반환

메시지 파일 개체입니다.

메시지 파일 검색 요청 예제

Python 1.x

```
Python

from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

message_files = client.beta.threads.messages.files.retrieve(
    thread_id="thread_abc123",
    message_id="msg_abc123",
    file_id="assistant-abc123"
)
print(message_files)
```

메시지 수정

HTTP

POST

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/messages/{message_id}?api-version=2024-02-15-preview

메시지를 수정합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	메시지가 속한 스레드의 ID입니다.
message_id	string	Required	수정할 메시지의 ID입니다.

요청 본문

매개변수	Type	필수	설명
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

수정된 [메시지](#) 개체입니다.

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

message = client.beta.threads.messages.update(
    message_id="msg_abc12",
    thread_id="thread_abc123",
    metadata={
        "modified": "true",
        "user": "abc123",
    },
)
print(message)
```

Message 개체

스레드 내의 메시지를 나타냅니다.

이름	형식	Description
id	string	API 엔드포인트에서 참조할 수 있는 식별자입니다.
object	string	항상 thread.message인 개체 형식입니다.

이름	형식	Description
<code>created_at</code>	정수	메시지가 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>thread_id</code>	string	이 메시지가 속한 스레드의 ID입니다.
<code>role</code>	string	메시지를 생성한 엔터티입니다. 사용자 또는 도우미 중 하나입니다.
<code>content</code>	배열	텍스트 및/또는 이미지 배열에서 메시지의 내용입니다.
<code>assistant_id</code>	문자열 또는 null	해당하는 경우 이 메시지를 작성한 도우미의 ID입니다.
<code>run_id</code>	문자열 또는 null	해당하는 경우 이 메시지의 작성과 연결된 실행의 ID입니다.
<code>file_ids</code>	배열	도우미가 사용해야 하는 파일 ID 목록입니다. 파일에 액세스할 수 있는 retrieval 및 code_interpreter 같은 도구에 유용합니다. 최대 10개 파일을 하나의 메시지에 첨부할 수 있습니다.
<code>metadata</code>	map	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

메시지 파일 개체

메시지에 첨부된 파일의 목록입니다.

[\[+\] 테이블 확장](#)

이름	형식	Description
<code>id</code>	string	API 엔드포인트에서 참조할 수 있는 식별자입니다.
<code>object</code>	string	항상 <code>thread.message.file</code> 인 개체 형식입니다.
<code>created_at</code>	정수	메시지 파일이 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>message_id</code>	string	파일이 연결된 메시지의 ID입니다.

도우미 API(미리 보기) 참조 실행

아티클 • 2024. 02. 22.

이 문서에서는 새로운 도우미 API(미리 보기)에 대한 Python 및 REST에 대한 참조 설명서를 제공합니다. 더 자세한 단계별 지침은 [시작 가이드](#)에서 제공됩니다.

실행 만들기

HTTP
POST
<code>https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs?api-version=2024-02-15-preview</code>

실행을 만듭니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	메시지를 만들 스레드의 ID입니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
assistant_id	string	Required	이 실행을 실행하는 데 사용할 도우미의 ID입니다.
model	문자열 또는 null	선택 사항	이 실행을 실행하는 데 사용할 모델 배포 이름입니다. 여기에 값이 제공되면 도우미와 연결된 모델 배포 이름이 재정의됩니다. 그렇지 않은 경우 도우미와 연결된 모델 배포 이름이 사용됩니다.
instructions	문자열 또는 null	선택 사항	도우미의 명령을 무시합니다. 이는 실행별로 동작을 수정하는 데 유용합니다.
tools	배열 또는 null	선택 사항	도우미가 이 실행에 사용할 수 있는 도구를 재정의합니다. 이는 실행별로 동작을 수정하는 데 유용합니다.
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데

이름	Type	필수	설명
			유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

실행 개체입니다.

예 실행 요청 만들기

Python 1.x

```
Python

from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

run = client.beta.threads.runs.create(
    thread_id="thread_abc123",
    assistant_id="asst_abc123"
)
print(run)
```

스레드 만들기 및 실행

HTTP

```
POST https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/runs?api-version=2024-02-15-preview
```

스레드를 만들고 단일 요청으로 실행합니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
assistant_id	string	Required	이 실행을 실행하는 데 사용할 도우미의 ID입니다.
thread	개체	선택 사항	
model	문자열 또는 null	선택 사항	이 실행을 실행하는 데 사용할 모델 배포 이름의 ID입니다. 여기에 값이 제공되면 도우미와 연결된 모델 배포 이름이 재정의됩니다. 그렇지 않은 경우 도우미와 연결된 모델 배포 이름이 사용됩니다.
instructions	문자열 또는 null	선택 사항	도우미의 기본 시스템 메시지를 대체합니다. 이는 실행별로 동작을 수정하는 데 유용합니다.
tools	배열 또는 null	선택 사항	도우미가 이 실행에 사용할 수 있는 도구를 재정의합니다. 이는 실행별로 동작을 수정하는 데 유용합니다.
metadata	map	선택 사항	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

실행 개체입니다.

예제 스레드 만들기 및 요청 실행

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

run = client.beta.threads.create_and_run(
    assistant_id="asst_abc123",
    thread={
        "messages": [
            {"role": "user", "content": "Explain deep learning to a 5 year old."}
```

```
    ]  
}  
)
```

목록 실행

HTTP

GET

[https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs?
api-version=2024-02-15-preview](https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs?api-version=2024-02-15-preview)

스레드에 속하는 실행 목록을 반환합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	실행이 속한 스레드의 ID입니다.

쿼리 매개 변수

[+] 테이블 확장

이름	Type	필수	설명
limit	정수	선택 사항 - 기본값은 20입니다.	반환할 개체 수에 대한 제한입니다. 제한 범위는 1~100이며 기본값은 20입니다.
order	string	선택 사항 - 기본값은 desc입니다.	개체의 Created_at 타임스탬프를 기준으로 정렬 순서입니다. 오름차순은 asc, 내림차순은 desc입니다.
after	string	선택 사항	페이지 매김에 사용되는 커서입니다. after는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 다음 페이지를 페치 위해 후속 호출에 after=obj_foo가 포함될 수 있습니다.
before	string	선택 사항	페이지 매김에 사용되는 커서입니다. before는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 이전 페이지를 페치 위해 전속 호출에 before=obj_foo가 포함될 수 있습니다.

이름	Type	필수	설명
			나는 100개의 개체를 받은 경우 목록의 이전 페이지를 페치 위해 후속 호출에 before=obj_foo가 포함될 수 있습니다.

반환

run 개체 목록입니다.

예제 목록 실행 요청

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

runs = client.beta.threads.runs.list(
    "thread_abc123"
)
print(runs)
```

실행 단계 나열

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs/{run_id}/steps?api-version=2024-02-15-preview

실행에 속하는 단계 목록을 반환합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	실행이 속한 스레드의 ID입니다.
run_id	string	Required	쿼리할 실행 단계와 연결된 실행의 ID입니다.

쿼리 매개 변수

[+] 테이블 확장

이름	Type	필수	설명
limit	정수	선택 사항 - 기본값은 20입니다.	반환할 개체 수에 대한 제한입니다. 제한 범위는 1~100이며 기본값은 20입니다.
order	string	선택 사항 - 기본값은 desc입니다.	개체의 Created_at 타임스탬프를 기준으로 정렬 순서입니다. 오름차순은 asc, 내림차순은 desc입니다.
after	string	선택 사항	페이지 매김에 사용되는 커서입니다. after는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 다음 페이지를 페치 위해 후속 호출에 after=obj_foo가 포함될 수 있습니다.
before	string	선택 사항	페이지 매김에 사용되는 커서입니다. before는 목록에서의 위치를 정의하는 개체 ID입니다. 예를 들어, 목록 요청을 하고 obj_foo로 끝나는 100개의 개체를 받은 경우 목록의 이전 페이지를 페치 위해 후속 호출에 before=obj_foo가 포함될 수 있습니다.

반환

실행 단계 개체 목록입니다.

예 목록 실행 단계 요청

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
```

```
)  
  
run_steps = client.beta.threads.runs.steps.list(  
    thread_id="thread_abc123",  
    run_id="run_abc123"  
)  
print(run_steps)
```

검색 실행

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs/{run_id}?api-version=2024-02-15-preview

실행을 검색합니다.

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	Type	필수	설명
thread_id	string	Required	실행된 스레드의 ID입니다.
run_id	string	Required	검색할 실행의 ID입니다.

반환

지정된 실행 ID와 일치하는 run 개체입니다.

예 목록 실행 단계 요청

Python 1.x

Python

```
from openai import AzureOpenAI  
  
client = AzureOpenAI(  
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),  
    api_version="2024-02-15-preview",  
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
```

```
)  
  
run = client.beta.threads.runs.retrieve(  
    thread_id="thread_abc123",  
    run_id="run_abc123"  
)  
print(run)
```

실행 단계 검색

HTTP

GET

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs/{run_id}/steps/{step_id}?api-version=2024-02-15-preview

실행 단계를 검색합니다.

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	Type	필수	설명
thread_id	string	Required	실행 및 실행 단계가 속한 스레드의 ID입니다.
run_id	string	Required	실행 단계가 속한 실행의 ID입니다.
step_id	string	Required	검색할 실행 단계의 ID입니다.

반환

지정된 ID와 일치하는 [실행 단계](#) 개체입니다.

예제 실행 단계 요청 검색

Python 1.x

Python

```
from openai import AzureOpenAI  
  
client = AzureOpenAI(  
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
```

```
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

run_step = client.beta.threads.runs.steps.retrieve(
    thread_id="thread_abc123",
    run_id="run_abc123",
    step_id="step_abc123"
)
print(run_step)
```

실행 설정

HTTP

POST

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs/{run_id}?api-version=2024-02-15-preview

실행을 수정합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	실행된 스레드의 ID입니다.
run_id	string	Required	수정할 실행의 ID입니다.

요청 본문

[+] 테이블 확장

이름	Type	필수	설명
metadata	map	선택	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 사항 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

반환

지정된 ID와 일치하는 수정된 run 개체입니다.

예제 수정 실행 요청

Python 1.x

```
Python

from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

run = client.beta.threads.runs.update(
    thread_id="thread_abc123",
    run_id="run_abc123",
    metadata={"user_id": "user_abc123"},
)
print(run)
```

실행할 도구 출력 제출

HTTP

POST

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs/{run_id}/submit_tool_outputs?api-version=2024-02-15-preview

실행 상태가 "requires_action"이고 required_action.type이 submit_tool_outputs인 경우
도구 호출이 모두 완료된 후 이 엔드포인트를 사용하여 도구 호출의 출력을 제출할 수 있습니다.
모든 출력은 단일 요청으로 제출되어야 합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	이 실행이 속한 스레드의 ID입니다.
run_id	string	Required	도구 출력 제출이 필요한 실행의 ID입니다.

요청 본문

이름	Type	필수	설명
`tool_outputs`	배열	Required	출력이 제출되는 도구 목록입니다.

반환

지정된 ID와 일치하는 수정된 [run](#) 개체입니다.

요청을 실행하기 위한 제출 도구 출력 예

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

run = client.beta.threads.runs.submit_tool_outputs(
    thread_id="thread_abc123",
    run_id="run_abc123",
    tool_outputs=[
        {
            "tool_call_id": "call_abc123",
            "output": "28C"
        }
    ]
)
print(run)
```

실행 취소

HTTP

POST

https://YOUR_RESOURCE_NAME.openai.azure.com/openai/threads/{thread_id}/runs/{run_id}/cancel?api-version=2024-02-15-preview

진행 중인 실행을 취소합니다.

경로 매개 변수

[+] 테이블 확장

매개 변수	Type	필수	설명
thread_id	string	Required	이 실행이 속한 스레드의 ID입니다.
run_id	string	Required	취소할 실행의 ID입니다.

반환

지정된 ID와 일치하는 수정된 run 개체입니다.

요청을 실행하기 위한 제출 도구 출력 예

Python 1.x

Python

```
from openai import AzureOpenAI

client = AzureOpenAI(
    api_key=os.getenv("AZURE_OPENAI_API_KEY"),
    api_version="2024-02-15-preview",
    azure_endpoint = os.getenv("AZURE_OPENAI_ENDPOINT")
)

run = client.beta.threads.runs.cancel(
    thread_id="thread_abc123",
    run_id="run_abc123"
)
print(run)
```

개체 실행

스레드에서 실행되는 실행을 나타냅니다.

[+] 테이블 확장

이름	형식	Description
<code>id</code>	string	API 엔드포인트에서 참조할 수 있는 식별자입니다.
<code>object</code>	string	항상 <code>thread.run</code> 인 개체 형식입니다.
<code>created_at</code>	정수	실행이 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>thread_id</code>	string	이 실행의 일부로 실행된 스레드의 ID입니다.
<code>assistant_id</code>	string	이 실행을 실행하는 데 사용되는 도우미의 ID입니다.
<code>status</code>	string	실행 상태는 <code>queued</code> , <code>in_progress</code> , <code>requires_action</code> , <code>cancelling</code> , <code>cancelled</code> , <code>failed</code> , <code>completed</code> 또는 <code>expired</code> 일 수 있습니다.
<code>required_action</code>	개체 또는 null	실행을 계속하는 데 필요한 작업에 대한 세부 정보입니다. 작업이 필요하지 않으면 null이 됩니다.
<code>last_error</code>	개체 또는 null	이 실행과 관련된 마지막 오류입니다. 오류가 없으면 null이 됩니다.
<code>expires_at</code>	정수	실행이 만료된 시점의 Unix 타임스탬프(초)입니다.
<code>started_at</code>	정수 또는 null	실행이 시작된 시점의 Unix 타임스탬프(초)입니다.
<code>cancelled_at</code>	정수 또는 null	실행이 취소된 시점의 Unix 타임스탬프(초)입니다.
<code>failed_at</code>	정수 또는 null	실행이 실패한 시점의 Unix 타임스탬프(초)입니다.
<code>completed_at</code>	정수 또는 null	실행이 완료된 시점의 Unix 타임스탬프(초)입니다.
<code>model</code>	string	도우미가 이 실행에 사용한 모델 배포 이름입니다.
<code>instructions</code>	string	도우미가 이 실행에 사용한 지침입니다.
<code>tools</code>	배열	도우미가 이 실행에 사용한 도구 목록입니다.
<code>file_ids</code>	배열	도우미가 이 실행에 사용한 파일 ID 목록입니다.
<code>metadata</code>	map	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

단계 개체 실행

실행 단계를 나타냅니다.

이름	형식	Description
<code>id</code>	string	API 엔드포인트에서 참조할 수 있는 실행 단계의 식별자입니다.
<code>object</code>	string	항상 <code>thread.run.step</code> 인 개체 형식입니다.
<code>created_at</code>	정수	실행 단계가 만들어진 시점의 Unix 타임스탬프(초)입니다.
<code>assistant_id</code>	string	실행 단계와 연결된 도우미의 ID입니다.
<code>thread_id</code>	string	실행된 스레드의 ID입니다.
<code>run_id</code>	string	이 실행 단계가 포함된 실행의 ID입니다.
<code>type</code>	string	<code>message_creation</code> 또는 <code>tool_calls</code> 일 수 있는 실행 단계의 형식입니다.
<code>status</code>	string	실행 단계의 상태는 <code>in_progress</code> , <code>cancelled</code> , <code>failed</code> , <code>completed</code> 또는 <code>expired</code> 일 수 있습니다.
<code>step_details</code>	개체	실행 단계의 세부 정보입니다.
<code>last_error</code>	개체 또는 null	이 실행 단계와 관련된 마지막 오류입니다. 오류가 없으면 <code>null</code> 이 됩니다.
<code>expired_at</code>	정수 또는 null	실행 단계가 만료된 시점의 Unix 타임스탬프(초)입니다. 부모 실행이 만료되면 단계가 만료된 것으로 간주됩니다.
<code>cancelled_at</code>	정수 또는 null	실행 단계가 취소된 시점의 Unix 타임스탬프(초)입니다.
<code>failed_at</code>	정수 또는 null	실행 단계가 실패한 시점의 Unix 타임스탬프(초)입니다.
<code>completed_at</code>	정수 또는 null	실행 단계가 완료된 시점의 Unix 타임스탬프(초)입니다.
<code>metadata</code>	map	개체에 연결할 수 있는 16개의 키-값 쌍 집합입니다. 이는 개체에 대한 추가 정보를 구조화된 형식으로 저장하는 데 유용할 수 있습니다. 키는 최대 64자, 값은 최대 512자까지 가능합니다.

Azure OpenAI: OpenAI Assistants client library for .NET - version 1.0.0-beta.3

Article • 03/13/2024

NOTE: This is a preview version of the Azure SDK library for OpenAI Assistants.

[OpenAI's Assistants API](#) is tagged as beta and both the API surface as well as this library's representation are subject to change. For other OpenAI features like Chat Completions, please see the [Azure SDK for .NET Azure.AI.OpenAI library](#).

The Azure OpenAI Assistants client library for .NET is an adaptation of OpenAI's REST APIs that provides an idiomatic interface and rich integration with the rest of the Azure SDK ecosystem. It will connect to Azure OpenAI resources or to the non-Azure OpenAI inference endpoint, making it a great choice for even non-Azure OpenAI development.

Use this library to:

- Create and manage assistants, threads, messages, and runs
- Configure and use tools with assistants
- Upload and manage files for use with assistants

Getting started

Prerequisites

To use Assistants capabilities, you'll need service API access through OpenAI or Azure OpenAI:

- To use OpenAI (`api.openai.com`), you'll need an API key obtained from a developer account at <https://platform.openai.com>
- (Not yet supported) If you'd like to use an Azure OpenAI resource, you must have an [Azure subscription](#) and [Azure OpenAI access](#). This will allow you to create an Azure OpenAI resource and get both a connection URL as well as API keys. For more information, see [Quickstart: Get started generating text using Azure OpenAI Service](#).

Install the package

Install the client library for .NET with [NuGet](#):

.NET CLI

```
dotnet add package Azure.AI.Assistants --prerelease
```

Authenticate the client

See [OpenAI's "how assistants work"](#) documentation for an overview of the concepts and relationships used with assistants. This overview closely follows [OpenAI's overview example](#) to demonstrate the basics of creating, running, and using assistants and threads.

To get started, create an `AssistantsClient`:

C#

```
AssistantsClient client = isAzureOpenAI
    ? new AssistantsClient(new Uri(azureResourceUrl), new
    AzureKeyCredential(azureApiKey))
    : new AssistantsClient(nonAzureApiKey);
```

Key concepts

Overview

For an overview of Assistants and the pertinent key concepts like Threads, Messages, Runs, and Tools, please see [OpenAI's Assistants API overview](#).

Usage

Examples

With an authenticated client, an assistant can be created:

C#

```
Response<Assistant> assistantResponse = await client.CreateAssistantAsync(
    new AssistantCreationOptions("gpt-4-1106-preview")
    {
        Name = "Math Tutor",
        Instructions = "You are a personal math tutor. Write and run code to
answer math questions.",
        Tools = { new CodeInterpreterToolDefinition() }
```

```
});  
Assistant assistant = assistantResponse.Value;
```

Next, create a thread:

C#

```
Response<AssistantThread> threadResponse = await client.CreateThreadAsync();  
AssistantThread thread = threadResponse.Value;
```

With a thread created, messages can be created on it:

C#

```
Response<ThreadMessage> messageResponse = await client.CreateMessageAsync(  
    thread.Id,  
    MessageRole.User,  
    "I need to solve the equation `3x + 11 = 14`. Can you help me?");  
ThreadMessage message = messageResponse.Value;
```

A run can then be started that evaluates the thread against an assistant:

C#

```
Response<ThreadRun> runResponse = await client.CreateRunAsync(  
    thread.Id,  
    new CreateRunOptions(assistant.Id)  
    {  
        AdditionalInstructions = "Please address the user as Jane Doe. The  
        user has a premium account.",  
    });  
ThreadRun run = runResponse.Value;
```

Once the run has started, it should then be polled until it reaches a terminal status:

C#

```
do  
{  
    await Task.Delay(TimeSpan.FromMilliseconds(500));  
    runResponse = await client.GetRunAsync(thread.Id, runResponse.Value.Id);  
}  
while (runResponse.Value.Status == RunStatus.Queued  
    || runResponse.Value.Status == RunStatus.InProgress);
```

Assuming the run successfully completed, listing messages from the thread that was run will now reflect new information added by the assistant:

C#

```
Response<PageableList<ThreadMessage>> afterRunMessagesResponse
    = await client.GetMessagesAsync(thread.Id);
IReadOnlyList<ThreadMessage> messages = afterRunMessagesResponse.Value.Data;

// Note: messages iterate from newest to oldest, with the messages[0] being
// the most recent
foreach (ThreadMessage threadMessage in messages)
{
    Console.WriteLine($"{threadMessage.CreatedAt:yyyy-MM-dd HH:mm:ss} - 
{threadMessage.Role,10}: ");
    foreach (MessageContent contentItem in threadMessage.ContentItems)
    {
        if (contentItem is MessageTextContent textItem)
        {
            Console.WriteLine(textItem.Text);
        }
        else if (contentItem is MessageImageFileContent imageFileItem)
        {
            Console.WriteLine($"<image from ID: {imageFileItem.FileId}>");
        }
        Console.WriteLine();
    }
}
```

Example output from this sequence:

```
2023-11-14 20:21:23 -  assistant: The solution to the equation \((3x + 11 =
14)\) is \((x = 1)\).
2023-11-14 20:21:18 -      user: I need to solve the equation `3x + 11 =
14`. Can you help me?
```

Working with files for retrieval

Files can be uploaded and then referenced by assistants or messages. First, use the generalized upload API with a purpose of 'assistants' to make a file ID available:

C#

```
File.WriteAllText(
    path: "sample_file_for_upload.txt",
    contents: "The word 'apple' uses the code 442345, while the word
'banana' uses the code 673457.");
Response<OpenAIFile> uploadAssistantFileResponse = await
client.UploadFileAsync(
    localFilePath: "sample_file_for_upload.txt",
```

```
purpose: OpenAIFilePurpose.Assistants);
OpenAIFile uploadedAssistantFile = uploadAssistantFileResponse.Value;
```

Once uploaded, the file ID can then be provided to an assistant upon creation. Note that file IDs will only be used if an appropriate tool like Code Interpreter or Retrieval is enabled.

C#

```
Response<Assistant> assistantResponse = await client.CreateAssistantAsync(
    new AssistantCreationOptions("gpt-4-1106-preview")
{
    Name = "SDK Test Assistant - Retrieval",
    Instructions = "You are a helpful assistant that can help fetch data from files you know about.",
    Tools = { new RetrievalToolDefinition() },
    FileIds = { uploadedAssistantFile.Id },
});
Assistant assistant = assistantResponse.Value;
```

With a file ID association and a supported tool enabled, the assistant will then be able to consume the associated data when running threads.

Using function tools and parallel function calling

As [described in OpenAI's documentation for assistant tools](#), tools that reference caller-defined capabilities as functions can be provided to an assistant to allow it to dynamically resolve and disambiguate during a run.

Here, outlined is a simple assistant that "knows how to," via caller-provided functions:

1. Get the user's favorite city
2. Get a nickname for a given city
3. Get the current weather, optionally with a temperature unit, in a city

To do this, begin by defining the functions to use -- the actual implementations here are merely representative stubs.

C#

```
// Example of a function that defines no parameters
string GetUserFavoriteCity() => "Seattle, WA";
FunctionToolDefinition getUserFavoriteCityTool = new("getUserFavoriteCity",
    "Gets the user's favorite city.");
// Example of a function with a single required parameter
string GetCityNickname(string location) => location switch
{
```

```

    "Seattle, WA" => "The Emerald City",
    _ => throw new NotImplementedException(),
};

FunctionToolDefinition getCityNicknameTool = new(
    name: "getCityNickname",
    description: "Gets the nickname of a city, e.g. 'LA' for 'Los Angeles, CA'.",
    parameters: BinaryData.FromObjectAsJson(
        new
        {
            Type = "object",
            Properties = new
            {
                Location = new
                {
                    Type = "string",
                    Description = "The city and state, e.g. San Francisco, CA",
                },
                Required = new[] { "location" },
            },
            new JsonSerializerOptions() { PropertyNamingPolicy =
JsonNamingPolicy.CamelCase });
// Example of a function with one required and one optional, enum parameter
string GetWeatherAtLocation(string location, string temperatureUnit = "f")
=> location switch
{
    "Seattle, WA" => temperatureUnit == "f" ? "70f" : "21c",
    _ => throw new NotImplementedException()
};

FunctionToolDefinition getCurrentWeatherAtLocationTool = new(
    name: "getCurrentWeatherAtLocation",
    description: "Gets the current weather at a provided location.",
    parameters: BinaryData.FromObjectAsJson(
        new
        {
            Type = "object",
            Properties = new
            {
                Location = new
                {
                    Type = "string",
                    Description = "The city and state, e.g. San Francisco, CA",
                },
                Unit = new
                {
                    Type = "string",
                    Enum = new[] { "c", "f" },
                },
            },
            Required = new[] { "location" },
        },
    ),
);

```

```
        new JsonSerializerOptions() { PropertyNamingPolicy =
JsonNamingPolicy.CamelCase }));
```

With the functions defined in their appropriate tools, an assistant can be now created that has those tools enabled:

C#

```
Response<Assistant> assistantResponse = await client.CreateAssistantAsync(
    // note: parallel function calling is only supported with newer models
    // like gpt-4-1106-preview
    new AssistantCreationOptions("gpt-4-1106-preview")
{
    Name = "SDK Test Assistant - Functions",
    Instructions = "You are a weather bot. Use the provided functions to
    help answer questions. "
        + "Customize your responses to the user's preferences as much as
    possible and use friendly "
        + "nicknames for cities whenever possible.",
    Tools =
    {
        getUserFavoriteCityTool,
        getCityNicknameTool,
        getCurrentWeatherAtLocationTool,
    },
});
Assistant assistant = assistantResponse.Value;
```

If the assistant calls tools, the calling code will need to resolve `ToolCall` instances into matching `ToolOutput` instances. For convenience, a basic example is extracted here:

C#

```
ToolOutput GetResolvedToolOutput(RequiredToolCall toolCall)
{
    if (toolCall is RequiredFunctionToolCall functionToolCall)
    {
        if (functionToolCall.Name == getUserFavoriteCityTool.Name)
        {
            return new ToolOutput(toolCall, GetUserFavoriteCity());
        }
        using JsonDocument argumentsJson =
JsonDocument.Parse(functionToolCall.Arguments);
        if (functionToolCall.Name == getCityNicknameTool.Name)
        {
            string locationArgument =
argumentsJson.RootElement.GetProperty("location").GetString();
            return new ToolOutput(toolCall,
GetCityNickname(locationArgument));
        }
        if (functionToolCall.Name == getCurrentWeatherAtLocationTool.Name)
```

```

        {
            string locationArgument =
argumentsJson.RootElement.GetProperty("location").GetString();
            if (argumentsJson.RootElement.TryGetProperty("unit", out
JsonElement unitElement))
            {
                string unitArgument = unitElement.GetString();
                return new ToolOutput(toolCall,
GetWeatherAtLocation(locationArgument, unitArgument));
            }
            return new ToolOutput(toolCall,
GetWeatherAtLocation(locationArgument));
        }
    }
    return null;
}

```

To handle user input like "what's the weather like right now in my favorite city?", polling the response for completion should be supplemented by a `RunStatus` check for `RequiresAction` or, in this case, the presence of the `RequiredAction` property on the run. Then, the collection of `ToolOutputSubmissions` should be submitted to the run via the `SubmitRunToolOutputs` method so that the run can continue:

C#

```

do
{
    await Task.Delay(TimeSpan.FromMilliseconds(500));
    runResponse = await client.GetRunAsync(thread.Id, runResponse.Value.Id);

    if (runResponse.Value.Status == RunStatus.RequiresAction
        && runResponse.Value.RequiredAction is SubmitToolOutputsAction
submitToolOutputsAction)
    {
        List<ToolOutput> toolOutputs = new();
        foreach (RequiredToolCall toolCall in
submitToolOutputsAction.ToolCalls)
        {
            toolOutputs.Add(GetResolvedToolOutput(toolCall));
        }
        runResponse = await
client.SubmitToolOutputsToRunAsync(runResponse.Value, toolOutputs);
    }
}

while (runResponse.Value.Status == RunStatus.Queued
    || runResponse.Value.Status == RunStatus.InProgress);

```

Note that, when using supported models, the assistant may request that several functions be called in parallel. Older models may only call one function at a time.

Once all needed function calls have been resolved, the run will proceed normally and the completed messages on the thread will contain model output supplemented by the provided function tool outputs.

Troubleshooting

When you interact with Azure OpenAI using the .NET SDK, errors returned by the service correspond to the same HTTP status codes returned for [REST API](#) requests.

For example, if you try to create a client using an endpoint that doesn't match your Azure OpenAI Resource endpoint, a `404` error is returned, indicating `Resource Not Found`.

Next steps

- Provide a link to additional code examples, ideally to those sitting alongside the README in the package's `/samples` directory.
- If appropriate, point users to other packages that might be useful.
- If you think there's a good chance that developers might stumble across your package in error (because they're searching for specific functionality and mistakenly think the package provides that functionality), point them to the packages they might be looking for.

Contributing

See the [Azure SDK CONTRIBUTING.md](#) for details on building, testing, and contributing to this library.

This project welcomes contributions and suggestions. Most contributions require you to agree to a Contributor License Agreement (CLA) declaring that you have the right to, and actually do, grant us the rights to use your contribution. For details, visit cla.microsoft.com.

When you submit a pull request, a CLA-bot will automatically determine whether you need to provide a CLA and decorate the PR appropriately (e.g., label, comment). Simply follow the instructions provided by the bot. You will only need to do this once across all repos using our CLA.

This project has adopted the [Microsoft Open Source Code of Conduct](#). For more information see the [Code of Conduct FAQ](#) or contact opencode@microsoft.com with any additional questions or comments.

 Collaborate with us on
GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure SDK for .NET
feedback

Azure SDK for .NET is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Azure OpenAI: OpenAI Assistants client library for Java - version 1.0.0-beta.2

Article • 02/07/2024

The Azure OpenAI Assistants client library for Java is an adaptation of OpenAI's REST APIs that provides an idiomatic interface and rich integration with the rest of the Azure SDK ecosystem. It will connect to Azure OpenAI resources or to the non-Azure OpenAI inference endpoint, making it a great choice for even non-Azure OpenAI development.

Use this library to:

- Create and manage assistants, threads, messages, and runs
- Configure and use tools with assistants
- Upload and manage files for use with assistants

Getting started

Prerequisites

- Java Development Kit (JDK) with version 8 or above
- Azure Subscription ↗
- Azure OpenAI access

Adding the package to your product

XML

```
<dependency>
    <groupId>com.azure</groupId>
    <artifactId>azure-ai-openai-assistants</artifactId>
    <version>1.0.0-beta.2</version>
</dependency>
```

Authentication

See [OpenAI's "how assistants work"](#) ↗ documentation for an overview of the concepts and relationships used with assistants. This overview closely follows [OpenAI's overview example](#) ↗ to demonstrate the basics of creating, running, and using assistants and threads.

Create a Azure OpenAI client with key credential

Get Azure OpenAI `key` credential from the Azure Portal.

Java

```
AssistantsClient client = new AssistantsClientBuilder()
    .credential(new AzureKeyCredential("{key}"))
    .endpoint("{endpoint}")
    .buildClient();
```

or

Java

```
AssistantsAsyncClient client = new AssistantsClientBuilder()
    .credential(new AzureKeyCredential("{key}"))
    .endpoint("{endpoint}")
    .buildAsyncClient();
```

Support for non-Azure OpenAI

The SDK also supports operating against the public non-Azure OpenAI. The response models remain the same, only the setup of the `Assistants Client` is slightly different.

First, get Non-Azure OpenAI API key from [Open AI authentication API keys](#). Then setup your `Assistants Client` as follows:

Java

```
AssistantsClient client = new AssistantsClientBuilder()
    .credential(new KeyCredential("{openai-secret-key}"))
    .buildClient();
```

or

Java

```
AssistantsAsyncClient client = new AssistantsClientBuilder()
    .credential(new KeyCredential("{openai-secret-key}"))
    .buildAsyncClient();
```

Key concepts

Overview

For an overview of Assistants and the pertinent key concepts like Threads, Messages, Runs, and Tools, please see [OpenAI's Assistants API overview](#).

Examples

Working with simple assistant operations

Create an assistant

With an authenticated client, an assistant can be created:

Java

```
AssistantCreationOptions assistantCreationOptions = new
AssistantCreationOptions("{deploymentOrModelId}")
    .setName("Math Tutor")
    .setInstructions("You are a personal math tutor. Answer questions
briefly, in a sentence or less.");
Assistant assistant = client.createAssistant(assistantCreationOptions);
```

Create a thread with message and then run it

Then a thread can be created:

Java

```
AssistantThread thread = client.createThread(new
AssistantThreadCreationOptions());
String threadId = thread.getId();
```

With a thread created, a message can be created on it:

Java

```
String userMessage = "I need to solve the equation `3x + 11 = 14`. Can you
help me?";
ThreadMessage threadMessage = client.createMessage(threadId,
MessageRole.USER, userMessage);
```

As we have a thread and message, we can create a run:

Java

```
ThreadRun run = client.createRun(threadId, new  
CreateRunOptions(assistantId));
```

There is also a convenience method to create a thread and message, and then run it in one call:

Java

```
CreateAndRunThreadOptions createAndRunThreadOptions = new  
CreateAndRunThreadOptions(assistantId)  
    .setThread(new AssistantThreadCreationOptions()  
        .setMessages(Arrays.asList(new  
ThreadInitializationMessage(MessageRole.USER,  
            "I need to solve the equation `3x + 11 = 14`. Can  
you help me?")));  
run = client.createThreadAndRun(createAndRunThreadOptions);
```

Once the run has started, it should then be polled until it reaches a terminal status:

Java

```
do {  
    run = client.getRun(run.getThreadId(), run.getId());  
    Thread.sleep(1000);  
} while (run.getStatus() == RunStatus.QUEUED || run.getStatus() ==  
RunStatus.IN_PROGRESS);
```

Assuming the run successfully completed, listing messages from the thread that was run will now reflect new information added by the assistant:

Java

```
PageableList<ThreadMessage> messages =  
client.listMessages(run.getThreadId());  
List<ThreadMessage> data = messages.getData();  
for (int i = 0; i < data.size(); i++) {  
    ThreadMessage dataMessage = data.get(i);  
    MessageRole role = dataMessage.getRole();  
    for (MessageContent messageContent : dataMessage.getContent()) {  
        MessageTextContent messageTextContent = (MessageTextContent)  
messageContent;  
        System.out.println(i + ": Role = " + role + ", content = " +  
messageTextContent.getText().getValue());  
    }  
}
```

For more examples, such as listing assistants/threads/messages/runs/runSteps, upload files, delete assistants/threads, etc, see the [samples](#).

Working with files for retrieval

Files can be uploaded and then referenced by assistants or messages. First, use the generalized upload API with a purpose of 'assistants' to make a file ID available:

Java

```
Path filePath = Paths.get("src", "samples", "resources", fileName);
BinaryData fileData = BinaryData.fromFile(filePath);
FileDetails fileDetails = new FileDetails(fileData).setFilename(fileName);

OpenAIFile openAIFile = client.uploadFile(fileDetails,
FilePurpose.ASSISTANTS);
```

Once uploaded, the file ID can then be provided to an assistant upon creation. Note that file IDs will only be used if an appropriate tool like Code Interpreter or Retrieval is enabled.

Java

```
Assistant assistant = client.createAssistant(
    new AssistantCreationOptions(deploymentOrModelId)
        .setName("Java SDK Retrieval Sample")
        .setInstructions("You are a helpful assistant that can help fetch
data from files you know about.")
        .setTools(Arrays.asList(new RetrievalToolDefinition()))
        .setFileIds(Arrays.asList(openAIFile.getId()))
);
```

With a file ID association and a supported tool enabled, the assistant will then be able to consume the associated data when running threads.

Using function tools and parallel function calling

As [described in OpenAI's documentation for assistant tools](#), tools that reference caller-defined capabilities as functions can be provided to an assistant to allow it to dynamically resolve and disambiguate during a run.

Here, outlined is a simple assistant that "knows how to," via caller-provided functions:

1. Get the user's favorite city
2. Get a nickname for a given city

3. Get the current weather, optionally with a temperature unit, in a city

To do this, begin by defining the functions to use -- the actual implementations here are merely representative stubs. For the full sample, please follow this [link ↗](#).

Java

```
private FunctionToolDefinition getUserFavoriteCityToolDefinition() {  
  
    class UserFavoriteCityParameters {  
  
        @JsonProperty("type")  
        private String type = "object";  
  
        @JsonProperty("properties")  
        private Map<String, Object> properties = new HashMap<>();  
    }  
  
    return new FunctionToolDefinition(  
        new FunctionDefinition(  
            GET_USER_FAVORITE_CITY,  
            BinaryData.fromObject(new UserFavoriteCityParameters()  
            )  
        ).setDescription("Gets the user's favorite city."));  
}
```

Please refer to [full sample ↗](#) for more details on how to set up methods with mandatory parameters and enum types.

With the functions defined in their appropriate tools, an assistant can be now created that has those tools enabled:

Java

```
AssistantCreationOptions assistantCreationOptions = new  
AssistantCreationOptions(deploymentOrModelId)  
    .setName("Java Assistants SDK Function Tool Sample Assistant")  
    .setInstructions("You are a weather bot. Use the provided functions to  
help answer questions."  
        + "Customize your responses to the user's preferences as much as  
possible and use friendly "  
        + "nicknames for cities whenever possible.")  
    .setTools(Arrays.asList(  
        getUserFavoriteCityToolDefinition(),  
        getCityNicknameToolDefinition(),  
        getWeatherAtLocationToolDefinition()  
    ));  
  
Assistant assistant = client.createAssistant(assistantCreationOptions);
```

If the assistant calls tools, the calling code will need to resolve ToolCall instances into matching ToolOutput instances. For convenience, a basic example is extracted here:

Java

```
private ToolOutput getResolvedToolOutput(RequiredToolCall toolCall) {
    if (toolCall instanceof RequiredFunctionToolCall) {
        RequiredFunctionToolCall functionToolCall =
        (RequiredFunctionToolCall) toolCall;
        RequiredFunctionToolCallDetails functionCallDetails =
        functionToolCall.getFunction();
        String name = functionCallDetails.getName();
        String arguments = functionCallDetails getArguments();
        ToolOutput toolOutput = new
        ToolOutput().setToolCallId(toolCall.getId());
        if (GET_USER_FAVORITE_CITY.equals(name)) {
            toolOutput.setOutput(getUserFavoriteCity());
        } else if (GET_CITY_NICKNAME.equals(name)) {
            Map<String, String> parameters =
            BinaryData.fromString(arguments)
                .toObject(new TypeReference<Map<String, String>>() {});
            String location = parameters.get("location");

            toolOutput.setOutput(getCityNickname(location));
        } else if (GET_WEATHER_AT_LOCATION.equals(name)) {
            Map<String, String> parameters =
            BinaryData.fromString(arguments)
                .toObject(new TypeReference<Map<String, String>>() {});
            String location = parameters.get("location");
            // unit was not marked as required on our Function tool
            // definition, so we need to handle its absence
            String unit = parameters.getOrDefault("unit", "c");

            toolOutput.setOutput(getWeatherAtLocation(location, unit));
        }
        return toolOutput;
    }
    throw new IllegalArgumentException("Tool call not supported: " +
    toolCall.getClass());
}
```

To handle user input like "what's the weather like right now in my favorite city?", polling the response for completion should be supplemented by a `RunStatus` check for `RequiresAction` or, in this case, the presence of the `RequiredAction` property on the run. Then, the collection of `toolOutputs` should be submitted to the run via the `SubmitRunToolOutputs` method so that the run can continue:

Java

```

do {
    Thread.sleep(500);
    run = client.getRun(thread.getId(), run.getId());

    if (run.getStatus() == RunStatus.REQUIRES_ACTION
        && run.getRequiredAction() instanceof SubmitToolOutputsAction) {
        SubmitToolOutputsAction requiredAction = (SubmitToolOutputsAction)
    run.getRequiredAction();
        List<ToolOutput> toolOutputs = new ArrayList<>();

        for (RequiredToolCall toolCall :
requiredAction.getSubmitToolOutputs().getToolCalls()) {
            toolOutputs.add(getResolvedToolOutput(toolCall));
        }
        run = client.submitToolOutputsToRun(thread.getId(), run.getId(),
    toolOutputs);
    }
} while (run.getStatus() == RunStatus.QUEUED || run.getStatus() ==
RunStatus.IN_PROGRESS);

```

Note that, when using supported models, the assistant may request that several functions be called in parallel. Older models may only call one function at a time.

Once all needed function calls have been resolved, the run will proceed normally and the completed messages on the thread will contain model output supplemented by the provided function tool outputs.

Troubleshooting

Enable client logging

You can set the `AZURE_LOG_LEVEL` environment variable to view logging statements made in the client library. For example, setting `AZURE_LOG_LEVEL=2` would show all informational, warning, and error log messages. The log levels can be found here: [log levels ↗](#).

Default HTTP Client

All client libraries by default use the Netty HTTP client. Adding the above dependency will automatically configure the client library to use the Netty HTTP client. Configuring or changing the HTTP client is detailed in the [HTTP clients wiki ↗](#).

Default SSL library

All client libraries, by default, use the Tomcat-native Boring SSL library to enable native-level performance for SSL operations. The Boring SSL library is an uber jar containing native libraries for Linux / macOS / Windows, and provides better performance compared to the default SSL implementation within the JDK. For more information, including how to reduce the dependency size, refer to the [performance tuning](#) section of the wiki.

Next steps

- Samples are explained in detail [here](#).

Contributing

For details on contributing to this repository, see the [contributing guide](#).

1. Fork it
2. Create your feature branch (`git checkout -b my-new-feature`)
3. Commit your changes (`git commit -am 'Add some feature'`)
4. Push to the branch (`git push origin my-new-feature`)
5. Create new Pull Request



Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure SDK for Java feedback

Azure SDK for Java is an open source project. Select a link to provide feedback:

[Open a documentation issue](#)

[Provide product feedback](#)

JavaScript용 Azure OpenAI Assistants 클라이언트 라이브러리 - 버전 1.0.0-beta.5

아티클 • 2024. 03. 02.

JavaScript용 Azure OpenAI Assistants 클라이언트 라이브러리는 Idiomatic 인터페이스를 제공하고 나머지 Azure SDK 에코시스템과 풍부한 통합을 제공하는 OpenAI의 REST API를 적응한 것입니다. Azure OpenAI 리소스 또는 비 Azure OpenAI 유추 엔드포인트에 연결할 수 있으므로 비 Azure OpenAI 개발에도 적합합니다.

주요 링크:

- [패키지\(NPM\)](#)
- [소스 코드](#)
- [API 참조 설명서](#)
- [제품 설명서](#)
- [샘플](#)

시작

현재 지원되는 환경

- [Node.js의 LTS 버전](#)
- 최신 버전의 Safari, Chrome, Edge, Firefox.

사전 요구 사항

Azure OpenAI 리소스를 사용하려면 [Azure 구독](#) 및 [Azure OpenAI 액세스 권한](#)이 있어야 합니다. 이렇게 하면 Azure OpenAI 리소스를 만들고 연결 URL과 API 키를 모두 가져올 수 있습니다. 자세한 내용은 [빠른 시작: Azure OpenAI Service를 사용하여 텍스트 생성 시작을 참조하세요](#).

Azure OpenAI Assistants JS 클라이언트 라이브러리를 사용하여 비 Azure OpenAI에 연결하려면 의 개발자 계정 <https://platform.openai.com/>에서 API 키가 필요합니다.

@azure/openai-assistants 패키지를 설치합니다.

를 사용하여 JavaScript용 Azure OpenAI Assistants 클라이언트 라이브러리를 npm 설치합니다.

Bash

```
npm install @azure/openai-assistants
```

AssistantsClient 만들기 및 인증

Azure OpenAI에서 사용할 클라이언트를 구성하려면 Azure OpenAI 리소스를 사용할 수 있는 권한이 부여된 해당 키 자격 증명, 토큰 자격 증명 또는 Azure ID 자격 증명과 함께 Azure OpenAI 리소스에 유효한 엔드포인트 URI를 제공합니다. 대신 OpenAI의 서비스에 연결하도록 클라이언트를 구성하려면 OpenAI의 개발자 포털에서 API 키를 제공합니다.

Azure에서 API 키 사용

[Azure Portal](#) 을 사용하여 OpenAI 리소스로 이동하고 API 키를 검색하거나 아래 [Azure CLI](#) 코드 조각을 사용합니다.

참고: API 키를 "구독 키" 또는 "구독 API 키"라고도 합니다.

PowerShell

```
az cognitiveservices account keys list --resource-group <your-resource-group-name> --name <your-resource-name>
```

주요 개념

도우미와 함께 사용되는 개념 및 관계에 대한 개요는 [OpenAI의 "도우미 작동 방식"](#) 설명서를 참조하세요. 이 개요는 [OpenAI의 개요 예제](#) 에 따라 도우미 및 스레드를 만들고, 실행하고, 사용하는 기본 사항을 보여 줍니다.

시작하려면 을 만듭니다 `AssistantsClient`.

JavaScript

```
const assistantsClient = new AssistantsClient("<endpoint>", new AzureKeyCredential("azur..."));
```

클라이언트를 사용하면 도우미 만들 수 있습니다. 도우미 도구는 도우미 수명 동안 개략적인 지침을 허용하면서 도구를 호출할 수 있는 OpenAI 모델에 대한 특별히 빌드된 인터페이스입니다.

도우미 만드는 코드:

JavaScript

```
const assistant = await assistantsClient.createAssistant({
  model: "gpt-4-1106-preview",
  name: "JS Math Tutor",
  instructions: "You are a personal math tutor. Write and run code to answer math questions.",
  tools: [{ type: "code_interpreter" }]
});
```

도우미와 사용자 간의 대화 세션을 스레드라고 합니다. 스레드는 메시지를 저장하고 콘텐츠를 모델의 컨텍스트에 맞게 잘림을 자동으로 처리합니다.

스레드를 만들려면 다음을 수행합니다.

JavaScript

```
const assistantThread = await assistantsClient.createThread();
```

메시지는 도우미 또는 사용자가 만든 메시지를 나타냅니다. 메시지에는 텍스트, 이미지 및 기타 파일이 포함될 수 있습니다. 메시지는 스레드의 목록으로 저장됩니다. 스레드를 만들면 메시지를 만들 수 있습니다.

JavaScript

```
const question = "I need to solve the equation '3x + 11 = 14'. Can you help me?";
const messageResponse = await
assistantsClient.createMessage(assistantThread.id, "user", question);
```

실행은 스레드에서 도우미의 호출을 나타냅니다. 도우미는 구성 및 스레드의 메시지를 사용하여 모델 및 도구를 호출하여 작업을 수행합니다. 실행의 일부로 도우미는 스레드에 메시지를 추가합니다. 그런 다음 도우미 대해 스레드를 평가하는 실행을 시작할 수 있습니다.

JavaScript

```
let runResponse = await assistantsClient.createRun(assistantThread.id, {
  assistantId: assistant.id,
  instructions: "Please address the user as Jane Doe. The user has a premium account."
});
```

실행이 시작되면 터미널 상태 도달할 때까지 폴링되어야 합니다.

JavaScript

```
do {
    await new Promise((resolve) => setTimeout(resolve, 800));
    runResponse = await assistantsClient.getRun(assistantThread.id,
runResponse.id);
} while (runResponse.status === "queued" || runResponse.status ===
"in_progress")
```

실행이 성공적으로 완료되었다고 가정하면 실행된 스레드의 메시지를 나열하면 이제 도우미 추가된 새 정보가 반영됩니다.

JavaScript

```
const runMessages = await assistantsClient.listMessages(assistantThread.id);
for (const runMessageDatum of runMessages.data) {
    for (const item of runMessageDatum.content) {
        if (item.type === "text") {
            console.log(item.text.value);
        } else if (item.type === "image_file") {
            console.log(item.imageFile.fileId);
        }
    }
}
```

이 시퀀스의 출력 예제:

```
2023-11-14 20:21:23 - assistant: The solution to the equation \((3x + 11 =
14)\) is \((x = 1)\).
2023-11-14 20:21:18 - user: I need to solve the equation `3x + 11 =
14`. Can you help me?
```

검색을 위해 파일 작업

파일을 업로드한 다음 도우미 또는 메시지에서 참조할 수 있습니다. 먼저 'assistants'를 목적으로 일반화된 업로드 API를 사용하여 파일 ID를 사용할 수 있도록 합니다.

JavaScript

```
const filename = "<path_to_text_file>";
await fs.writeFile(filename, "The word 'apple' uses the code 442345, while
the word 'banana' uses the code 673457.", "utf8");
const uint8array = await fs.readFile(filename);
const uploadAssistantFile = await assistantsClient.uploadFile(uint8array,
"assistants", { filename });
```

업로드되면 파일 ID를 만들 때 도우미 제공할 수 있습니다. 파일 ID는 코드 인터프리터 또는 검색과 같은 적절한 도구를 사용하도록 설정한 경우에만 사용됩니다.

JavaScript

```
const fileAssistant = await assistantsClient.createAssistant({
  model: "gpt-4-1106-preview",
  name: "JS SDK Test Assistant - Retrieval",
  instructions: "You are a helpful assistant that can help fetch data from files you know about.",
  tools: [{ type: "retrieval" }],
  fileIds: [ uploadAssistantFile.id ]
});
```

파일 ID 연결 및 지원되는 도구를 사용하도록 설정하면 도우미 스레드를 실행할 때 연결된 데이터를 사용할 수 있습니다.

함수 도구 및 병렬 함수 호출 사용

도우미 도구에 대한 OpenAI 설명서에 설명된[설명된](#) 대로 호출자 정의 기능을 함수로 참조하는 도구를 도우미 제공하여 실행 중에 동적으로 resolve 명확하게 할 수 있습니다.

여기서는 호출자가 제공하는 함수를 통해 "방법을 알고 있다"는 간단한 도우미 간략하게 설명합니다.

1. 사용자가 가장 좋아하는 도시 가져오기
2. 지정된 도시에 대한 애칭 가져오기
3. 도시의 온도 단위를 사용하여 현재 날씨를 선택적으로 가져옵니다.

이렇게 하려면 먼저 사용할 함수를 정의합니다. 여기서 실제 구현은 단지 대표적인 스텝일 뿐입니다.

JavaScript

```
// Example of a function that defines no parameters
const getFavoriteCity = () => "Atlanta, GA";
const getUserFavoriteCityTool = {
  type: "function",
  function: {
    name: "getUserFavoriteCity",
    description: "Gets the user's favorite city.",
    parameters: {
      type: "object",
      properties: {}
    }
  }
};
```

```
// Example of a function with a single required parameter
const getCityNickname = (city) => {
  switch (city) {
    case "Atlanta, GA":
      return "The ATL";
    case "Seattle, WA":
      return "The Emerald City";
    case "Los Angeles, CA":
      return "LA";
    default:
      return "Unknown";
  }
};

const getCityNicknameTool = {
  type: "function",
  function: {
    name: "getCityNickname",
    description: "Gets the nickname for a city, e.g. 'LA' for 'Los Angeles, CA'.",
    parameters: {
      type: "object",
      properties: {
        city: {
          type: "string",
          description: "The city and state, e.g. San Francisco, CA"
        }
      }
    }
  }
};

// Example of a function with one required and one optional, enum parameter
const getWeatherAtLocation = (location, temperatureUnit = "f") => {
  switch (location) {
    case "Atlanta, GA":
      return temperatureUnit === "f" ? "84f" : "26c";
    case "Seattle, WA":
      return temperatureUnit === "f" ? "70f" : "21c";
    case "Los Angeles, CA":
      return temperatureUnit === "f" ? "90f" : "28c";
    default:
      return "Unknown";
  }
};

const getWeatherAtLocationTool = {
  type: "function",
  function: {
    name: "getWeatherAtLocation",
    description: "Gets the current weather at a provided location.",
    parameters: {
      type: "object",
      properties: {
        location: {

```

```
        type: "string",
        description: "The city and state, e.g. San Francisco, CA"
    },
    temperatureUnit: {
        type: "string",
        enum: ["f", "c"],
    }
},
required: ["location"]
}
}
};


```

적절한 도구에 정의된 함수를 사용하면 이제 해당 도구를 사용하도록 설정된 도우미 만들 수 있습니다.

JavaScript

```
const weatherAssistant = await assistantsClient.createAssistant({
// note: parallel function calling is only supported with newer models
like gpt-4-1106-preview
model: "gpt-4-1106-preview",
name: "JS SDK Test Assistant - Weather",
instructions: `You are a weather bot. Use the provided functions to help
answer questions.

Customize your responses to the user's preferences as much as possible
and use friendly
nicknames for cities whenever possible.
`,
tools: [getUserFavoriteCityTool, getCityNicknameTool,
getWeatherAtLocationTool]
});
```

도우미 도구를 호출하는 경우 호출 코드는 인스턴스를 일치하는 `ToolOutputSubmission` 인스턴스로 `resolve ToolCall` 합니다. 편의를 위해 기본 예제는 여기에서 추출됩니다.

JavaScript

```
const getResolvedToolOutput = (toolCall) => {
    const toolOutput = { toolCallId: toolCall.id };

    if (toolCall["function"]) {
        const functionCall = toolCall["function"];
        const functionName = functionCall.name;
        const functionArgs = JSON.parse(functionCall["arguments"] ?? {});

        switch (functionName) {
            case "getUserFavoriteCity":
                toolOutput.output = getFavoriteCity();
                break;
            case "getCityNickname":
```

```

        toolOutput.output = getCityNickname(functionArgs["city"]);
        break;
    case "getWeatherAtLocation":
        toolOutput.output = getWeatherAtLocation(functionArgs.location,
functionArgs.temperatureUnit);
        break;
    default:
        toolOutput.output = `Unknown function: ${functionName}`;
        break;
    }
}
return toolOutput;
};

```

"내가 가장 좋아하는 도시에서 지금 날씨는 어때?"와 같은 사용자 입력을 처리하려면 완료에 대한 응답을 폴링하는 검사 `RequiresAction` 보완 `RunStatus` 해야 합니다. 이 경우 실행에 속성이 `RequiredAction` 존재합니다. 그런 다음, 실행을 계속할 수 있도록의 `ToolOutputSubmissions` 컬렉션을 메서드를 `SubmitRunToolOutputs` 통해 실행에 제출해야 합니다.

JavaScript

```

const question = "What's the weather like right now in my favorite city?";
let runResponse = await assistantsClient.createThreadAndRun({
    assistantId: weatherAssistant.id,
    thread: { messages: [{ role: "user", content: question }] },
    tools: [getUserFavoriteCityTool, getCityNicknameTool,
    getWeatherAtLocationTool]
});

do {
    await new Promise((resolve) => setTimeout(resolve, 500));
    runResponse = await assistantsClient.getRun(runResponse.threadId,
runResponse.id);

    if (runResponse.status === "requires_action" &&
runResponse.requiredAction.type === "submit_tool_outputs") {
        const toolOutputs = [];

        for (const toolCall of
runResponse.requiredAction.submitToolOutputs.toolCalls) {
            toolOutputs.push(getResolvedToolOutput(toolCall));
        }
        runResponse = await
assistantsClient.submitToolOutputsToRun(runResponse.threadId,
runResponse.id, toolOutputs);
    }
} while (runResponse.status === "queued" || runResponse.status ===
"in_progress")

```

지원되는 모델을 사용하는 경우 도우미 여러 함수를 병렬로 호출할 것을 요청할 수 있습니다. 이전 모델은 한 번에 하나의 함수만 호출할 수 있습니다.

필요한 모든 함수 호출이 해결되면 실행이 정상적으로 진행되며 스레드의 완료된 메시지에는 제공된 함수 도구 출력으로 보완된 모델 출력이 포함됩니다.

문제 해결

로깅

로깅을 사용하도록 설정하면 실패에 대한 유용한 정보를 파악하는 데 도움이 될 수 있습니다. HTTP 요청 및 응답 로그를 보려면 `AZURE_LOG_LEVEL` 환경 변수를 `info`로 설정합니다. 또는 `@azure/logger`에서 `setLogLevel`을 호출하여 런타임에 로깅을 사용하도록 설정할 수 있습니다.

JavaScript

```
const { setLogLevel } = require("@azure/logger");

setLogLevel("info");
```

로그를 사용하는 방법에 대한 자세한 내용은 [@azure/logger package docs](#)를 참조하세요.

GitHub에서 Microsoft와 공동 작업

이 콘텐츠의 원본은 GitHub에서 찾을 수 있으며, 여기서 문제와 끌어오기 요청을 만들고 검토할 수도 있습니다. 자세한 내용은 [참여자 가이드](#)를 참조하세요.



Azure SDK for JavaScript 피드백

Azure SDK for JavaScript은(는) 오픈 소스 프로젝트입니다. 다음 링크를 선택하여 피드백을 제공해 주세요.

 설명서 문제 열기

 제품 사용자 의견 제공

데이터 API 참조의 Azure OpenAI

아티클 • 2024. 03. 20.

이 문서에서는 새 Azure OpenAI On Your Data API에 대한 Python 및 REST에 대한 참조 설명서를 제공합니다. 최신 API 버전은 Swagger 사양[입니다](#)2024-02-01↗.

① 참고

API 버전 이후 이전 API 버전 2024-02-15-preview 과 비교하여 다음과 같은 호환성이 손상되는 변경이 도입되었습니다.

- API 경로가 .로 /extensions/chat/completions /chat/completions 변경됩니다.
- 속성 키 및 열거형 값의 명명 규칙은 낙타 대/소문자에서 뱀 대/소문자로 변경됩니다. 예: deploymentName .로 deployment_name 변경됩니다.
- 데이터 원본 형식 AzureCognitiveSearch 이 .로 azure_search 변경됩니다.
- 인용 및 의도는 도우미 메시지의 컨텍스트 도구 메시지에서 명시적 스키마가 정의된 도우미 메시지의 컨텍스트 루트 수준으로 이동됩니다.

HTTP

```
POST {endpoint}/openai/deployments/{deployment-id}/chat/completions?api-version={api-version}
```

지원되는 버전

- 2024-02-15-preview Swagger 사양↗입니다.
- 2024-02-01 Swagger 사양↗입니다.

① 참고

Azure Machine Learning 인덱스, Pinecone 및 Elasticsearch 는 API 버전에서 2024-02-15-preview 만 미리 보기로 지원됩니다.

URI 매개 변수

[+] 테이블 확장

이름	In	Type	필수	설명
deployment-id	경로	string	True	이 요청에 사용할 채팅 완료 모델 배포 이름을 지정합니다.
endpoint	경로	string	True	Azure OpenAI 엔드포인트. 예: https://YOUR_RESOURCE_NAME.openai.azure.com
api-version	query	string	True	이 작업에 사용할 API 버전입니다.

요청 본문

요청 본문은 채팅 완료 API 요청과 동일한 스키마를 상속합니다. 이 표에서는 Azure OpenAI On Your Data에 대한 고유한 매개 변수를 보여 줍니다.

[+] 테이블 확장

속성	Type	필수	설명
data_sources	DataSource[]	True	데이터에 대한 Azure OpenAI에 대한 구성 항목입니다. 배열에는 정확히 하나의 요소가 있어야 합니다. 제공되지 않은 경우 <code>data_sources</code> 서비스는 채팅 완료 모델을 직접 사용하며 Azure OpenAI On Your Data를 사용하지 않습니다.

응답 본문

응답 본문은 채팅 완료 API 응답과 동일한 스키마를 상속합니다. [응답 채팅 메시지](#)에는 `context` Azure OpenAI On Your Data에 대해 추가되는 속성이 있습니다.

채팅 메시지

응답 도우미 메시지 스키마는 채팅 완료 도우미 [채팅 메시지](#)에서 상속되며 속성 `context`으로 확장됩니다.

[+] 테이블 확장

속성	Type	필수	설명
context	Context	False	검색된 문서를 포함하여 요청을 처리하는 동안 Azure OpenAI On Your Data에서 수행하는 충분 단계를 나타냅니다.

Context

[+] 테이블 확장

속성	Type	필수	설명
citations	인용	False	응답에서 도우미 메시지를 생성하는 데 사용되는 데이터 원본 검색 결과입니다. 클라이언트는 인용에서 참조를 렌더링할 수 있습니다.
intent	string	False	채팅 기록에서 검색된 의도입니다. 이전 의도를 다시 전달할 필요가 없습니다. 이 속성을 무시합니다.

인용

[+] 테이블 확장

속성	Type	필수	설명
content	string	True	인용 내용입니다.
title	string	False	인용 제목입니다.
url	string	False	인용의 URL입니다.
filepath	string	False	인용의 파일 경로입니다.
chunk_id	string	False	인용의 청크 ID입니다.

데이터 원본

이 목록에는 지원되는 데이터 원본이 표시됩니다.

- Azure AI 검색
- Azure Cosmos DB for MongoDB vCore
- Azure Machine Learning 인덱스(미리 보기)
- Elasticsearch(미리 보기)
- Pinecone(미리 보기)

예제

이 예제에서는 더 나은 결과를 위해 대화 기록을 전달하는 방법을 보여 줍니다.

필수 조건:

- Azure OpenAI 시스템에서 할당된 관리 ID에서 Azure Search 서비스로 역할 할당을 구성합니다. 필수 역할: `Search Index Data Reader`, `. Search Service Contributor`
- 사용자에서 Azure OpenAI 리소스로 역할 할당을 구성합니다. 필수 역할: `Cognitive Services OpenAI User`.
- Az CLI를 설치하고 실행 `az login`합니다.
- 다음 환경 변수를 정의합니다.

```
AzureOpenAIEndpoint ChatCompletionsDeploymentName SearchEndpoint SearchIndex
```

Bash

```
export AzureOpenAIEndpoint=https://example.openai.azure.com/
export ChatCompletionsDeploymentName=turbo
export SearchEndpoint=https://example.search.windows.net
export SearchIndex=example-index
```

Python 1.x

최신 pip 패키지를 설치합니다. `openai azure-identity`

Python

```
import os
from openai import AzureOpenAI
from azure.identity import DefaultAzureCredential,
get_bearer_token_provider

endpoint = os.environ.get("AzureOpenAIEndpoint")
deployment = os.environ.get("ChatCompletionsDeploymentName")
search_endpoint = os.environ.get("SearchEndpoint")
search_index = os.environ.get("SearchIndex")

token_provider = get_bearer_token_provider(DefaultAzureCredential(),
"https://cognitiveservices.azure.com/.default")

client = AzureOpenAI(
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
    api_version="2024-02-01",
)

completion = client.chat.completions.create(
    model=deployment,
    messages=[
        {
            "role": "user",
            "content": "Who is DRI?",
        },
        {
            "role": "assistant",
        }
    ]
)
```

```
        "content": "DRI stands for Directly Responsible Individual  
of a service. Which service are you asking about?"  
    },  
    {  
        "role": "user",  
        "content": "Opinion mining service"  
    }  
],  
extra_body={  
    "data_sources": [  
        {  
            "type": "azure_search",  
            "parameters": {  
                "endpoint": search_endpoint,  
                "index_name": search_index,  
                "authentication": {  
                    "type": "system_assigned_managed_identity"  
                }  
            }  
        }  
    ]  
}  
)  
  
print(completion.model_dump_json(indent=2))
```

데이터 원본 - Azure AI Search

아티클 · 2024. 03. 13.

데이터에서 Azure OpenAI를 사용하는 경우 Azure AI Search의 구성 가능한 옵션입니다. 이 데이터 원본은 API 버전 2024-02-01에서 지원됩니다.

 테이블 확장

속성	Type	필수	설명
parameters	매개 변수	True	Azure Search를 구성할 때 사용할 매개 변수입니다.
type	string	True	azure_search이어야 합니다.

매개 변수

 테이블 확장

이름	Type	필수	설명
endpoint	string	True	사용할 Azure Search 리소스의 절대 엔드포인트 경로입니다.
index_name	string	True	참조된 Azure Search 리소스에 사용할 인덱스의 이름입니다.
authentication	ApiKeyAuthenticationOptions 중 하나, SystemAssignedManagedIdentityAuthenticationOptions, UserAssignedManagedIdentityAuthenticationOptions	True	정의된 데이터 원본에 액세스할 때 사용할 인증 방법입니다.
embedding_dependency	DeploymentNameVectorizationSource, EndpointVectorizationSource 중 하나	False	벡터 검색에 포함되는 종속성입니다. 필요한 경우 <code>query_type</code> , <code>vector_vector_simple_hybrid</code> 또는 <code>vector_semantic_hybrid</code> .
fields_mapping	FieldsMappingOptions	False	검색 인덱스와 상호 작용할 때 사용할 사용자 지정된 필드 매팅 동작입니다.
filter	string	False	검색 필터입니다.
in_scope	부울 값	False	쿼리를 인덱싱된 데이터 사용으로 제한해야 하는지 여부입니다. 기본값은 <code>True</code> 입니다.
query_type	QueryType	False	Azure Search와 함께 사용할 쿼리 유형입니다. 기본값은 <code>simple</code>
role_information	string	False	응답을 생성할 때 참조해야 하는 컨텍스트와 작동 방식에 대한 지침을 모델에 제공합니다. 도우미 성격에 대해 설명하고 응답 형식을 지정하는 방법을 알려줄 수 있습니다.
semantic_configuration	string	False	쿼리에 대한 의미 체계 구성입니다. 필요한 경우 <code>query_type semantic</code> 또는 <code>vector_semantic_hybrid</code> .
strictness	정수	False	검색 관련성 필터링의 구성된 엄격성입니다. 엄격성이 높을수록 정밀도가 높지만 대답의 재현율이 낮습니다. 기본값은 3입니다.
top_n_documents	정수	False	구성된 쿼리에 대해 기능할 구성된 상위 문서 수입니다. 기본값은 5입니다.

API 키 인증 옵션

API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 테이블 확장

속성	Type	필수	설명
key	string	True	인증에 사용할 API 키입니다.
type	string	True	api_key 이어야 합니다.

시스템 할당 관리 ID 인증 옵션

시스템 할당 관리 ID를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 [데이터 확장](#)

속성	Type	필수	설명
type	string	True	system_assigned_managed_identity 이어야 합니다.

사용자 할당 관리 ID 인증 옵션

사용자 할당 관리 ID를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 [데이터 확장](#)

속성	Type	필수	설명
managed_identity_resource_id	string	True	인증 시 이용할 사용자가 할당한 관리 ID의 리소스 ID입니다.
type	string	True	user_assigned_managed_identity 이어야 합니다.

배포 이름 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 동일한 Azure OpenAI 리소스의 내부 포함 모델 배포 이름을 기반으로 합니다. 이 벡터화 원본을 사용하면 Azure OpenAI api-key 없이 Azure OpenAI 공용 네트워크 액세스 없이 벡터 검색을 사용할 수 있습니다.

 [데이터 확장](#)

속성	Type	필수	설명
deployment_name	string	True	동일한 Azure OpenAI 리소스 내의 포함 모델 배포 이름입니다.
type	string	True	deployment_name 이어야 합니다.

엔드포인트 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 Azure OpenAI 포함 API 엔드포인트를 기반으로 합니다.

 [데이터 확장](#)

속성	Type	필수	설명
endpoint	string	True	포함을 검색할 리소스 엔드포인트 URL을 지정합니다. 형식이어야 합니다. <code>https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/embeddings</code> api-version 쿼리 매개 변수는 허용되지 않습니다.
authentication	ApiKeyAuthenticationOptions	True	지정된 엔드포인트에서 포함을 검색할 때 사용할 인증 옵션을 지정합니다.
type	string	True	endpoint 이어야 합니다.

필드 매팅 옵션

구성된 Azure Search 리소스를 사용할 때 필드가 처리되는 방식을 제어하는 선택적 설정입니다.

속성	Type	필수	설명
content_fields	string[]	False	콘텐츠로 처리해야 하는 인덱스 필드의 이름입니다.
vector_fields	string[]	False	벡터 데이터를 나타내는 필드의 이름입니다.
content_fields_separator	string	False	콘텐츠 필드에서 사용해야 하는 구분 기호 패턴입니다. 기본값은 \n입니다.
filepath_field	string	False	파일 경로로 사용할 인덱스 필드의 이름입니다.
title_field	string	False	제목으로 사용할 인덱스 필드의 이름입니다.
url_field	string	False	URL로 사용할 인덱스 필드의 이름입니다.

쿼리 유형

Azure OpenAI On Your Data로 사용할 때 실행해야 하는 Azure Search 검색 쿼리의 유형입니다.

열거형 값	설명
simple	기본 단순 쿼리 파서입니다.
semantic	고급 의미 체계 모델링을 위한 의미 체계 쿼리 파서입니다.
vector	계산된 데이터에 대한 벡터 검색을 나타냅니다.
vector_simple_hybrid	벡터 데이터와 간단한 쿼리 전략의 조합을 나타냅니다.
vector_semantic_hybrid	의미 체계 검색 및 벡터 데이터 쿼리의 조합을 나타냅니다.

예제

필수 조건:

- Azure OpenAI 시스템에서 할당된 관리 ID에서 Azure Search 서비스로 역할 할당을 구성합니다. 필수 역할: `Search Index Data Reader`, `Search Service Contributor`
- 사용자에서 Azure OpenAI 리소스로 역할 할당을 구성합니다. 필수 역할: `Cognitive Services OpenAI User`.
- Az CLI를 설치하고 실행 `az login`합니다.
- 다음 환경 변수를 정의합니다. `AzureOpenAIEndpoint ChatCompletionsDeploymentName SearchEndpoint SearchIndex`

Bash

```
export AzureOpenAIEndpoint=https://example.openai.azure.com/
export ChatCompletionsDeploymentName=turbo
export SearchEndpoint=https://example.search.windows.net
export SearchIndex=example-index
```

Python 1.x

최신 pip 패키지를 설치합니다. `openai azure-identity`

Python

```
import os
from openai import AzureOpenAI
from azure.identity import DefaultAzureCredential, get_bearer_token_provider

endpoint = os.environ.get("AzureOpenAIEndpoint")
deployment = os.environ.get("ChatCompletionsDeploymentName")
search_endpoint = os.environ.get("SearchEndpoint")
search_index = os.environ.get("SearchIndex")

token_provider = get_bearer_token_provider(DefaultAzureCredential(),
"https://cognitiveservices.azure.com/.default")
```

```
client = AzureOpenAI(
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
    api_version="2024-02-01",
)

completion = client.chat.completions.create(
    model=deployment,
    messages=[
        {
            "role": "user",
            "content": "Who is DRI?",
        },
    ],
    extra_body={
        "data_sources": [
            {
                "type": "azure_search",
                "parameters": {
                    "endpoint": search_endpoint,
                    "index_name": search_index,
                    "authentication": {
                        "type": "system_assigned_managed_identity"
                    }
                }
            }
        ]
    }
)

print(completion.model_dump_json(indent=2))
```

데이터 원본 - Azure Cosmos DB for MongoDB vCore

아티클 • 2024. 03. 13.

Azure OpenAI On Your Data를 사용하는 경우 MongoDB vCore용 Azure Cosmos DB의 구성 가능한 옵션입니다. 이 데이터 원본은 API 버전 2024-02-01에서 지원됩니다.

 테이블 확장

속성	Type	필수	설명
parameters	매개 변수	True	Azure Cosmos DB for MongoDB vCore를 구성할 때 사용할 매개 변수입니다.
type	string	True	azure_cosmos_db 이어야 합니다.

매개 변수

 테이블 확장

이름	Type	필수	설명
database_name	string	True	Azure Cosmos DB와 함께 사용할 MongoDB vCore 데이터베이스 이름입니다.
container_name	string	True	Azure Cosmos DB 리소스 컨테이너의 이름입니다.
index_name	string	True	Azure Cosmos DB와 함께 사용할 MongoDB vCore 인덱스 이름입니다.
fields_mapping	FieldsMappingOptions	True	검색 인덱스와 상호 작용할 때 사용할 사용자 지정된 필드 매핑 동작입니다.
authentication	커넥션 문자열 Authentication Options	True	정의된 데이터 원본에 액세스할 때 사용할 인증 방법입니다.
embedding_dependency	DeploymentNameVectorizationSource, EndpointVectorizationSource 중 하나	True	벡터 검색에 포함되는 종속성입니다.
in_scope	부울 값	False	쿼리를 인덱싱된 데이터 사용으로 제한해야 하는지 여부입니다. 기본값은 True입니다.
role_information	string	False	응답을 생성할 때 참조해야 하는 컨텍스트와 작동 방식에 대한 지침을 모델에 제공합니다. 도우미 성격에 대해 설명하고 응답 형식을 지정하는 방법을 알려줄 수 있습니다.
strictness	정수	False	검색 관련성 필터링의 구성된 엄격성입니다. 엄격성이 높을수록 정밀도가 높지만 대답의 재현율이 낮습니다. 기본값은 3입니다.
top_n_documents	정수	False	구성된 쿼리에 대해 기능할 구성된 상위 문서 수입니다. 기본값은 5입니다.

커넥션 문자열 인증 옵션

연결 문자열 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 테이블 확장

속성	Type	필수	설명
connection_string	string	True	인증에 사용할 연결 문자열.
type	string	True	connection_string 이어야 합니다.

배포 이름 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 동일한 Azure OpenAI 리소스의 내부 포함 모델 배포 이름을 기반으로 합니다. 이 벡터화 원본을 사용하면 Azure OpenAI api-key 없이 Azure OpenAI

공용 네트워크 액세스 없이 벡터 검색을 사용할 수 있습니다.

 테이블 확장

속성	Type	필수	설명
deployment_name	string	True	동일한 Azure OpenAI 리소스 내의 포함 모델 배포 이름입니다.
type	string	True	<code>deployment_name</code> 이어야 합니다.

엔드포인트 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 Azure OpenAI 포함 API 엔드포인트를 기반으로 합니다.

 테이블 확장

속성	Type	필수	설명
endpoint	string	True	포함을 검색할 리소스 엔드포인트 URL을 지정합니다. 형식이어야 <code>https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/embeddings</code> 입니다. api-version 쿼리 매개 변수는 허용되지 않습니다.
authentication	ApiKeyAuthenticationOptions	True	지정된 엔드포인트에서 포함을 검색할 때 사용할 인증 옵션을 지정합니다.
type	string	True	<code>endpoint</code> 이어야 합니다.

API 키 인증 옵션

API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 테이블 확장

속성	Type	필수	설명
key	string	True	인증에 사용할 API 키입니다.
type	string	True	<code>api_key</code> 이어야 합니다.

필드 매팅 옵션

필드 처리 방법을 제어하는 설정입니다.

 테이블 확장

속성	Type	필수	설명
content_fields	string[]	True	콘텐츠로 처리해야 하는 인덱스 필드의 이름입니다.
vector_fields	string[]	True	벡터 데이터를 나타내는 필드의 이름입니다.
content_fields_separator	string	False	콘텐츠 필드에서 사용해야 하는 구분 기호 패턴입니다. 기본값은 <code>\n</code> 입니다.
filepath_field	string	False	파일 경로로 사용할 인덱스 필드의 이름입니다.
title_field	string	False	제목으로 사용할 인덱스 필드의 이름입니다.
url_field	string	False	URL로 사용할 인덱스 필드의 이름입니다.

예제

필수 조건:

- 사용자에서 Azure OpenAI 리소스로 역할 할당을 구성합니다. 필수 역할: Cognitive Services OpenAI User.

- Az CLI를 설치하고 실행 az login 합니다.
- 다음 환경 변수를 정의합니다.

```
AzureOpenAIEndpoint ChatCompletionsDeploymentName ConnectionString Database Container Index EmbeddingDeploymentName
```

Bash

```
export AzureOpenAIEndpoint=https://example.openai.azure.com/
export ChatCompletionsDeploymentName=turbo
export ConnectionString='mongodb+srv://username:***@example.mongocluster.cosmos.azure.com/?tls=true&authMechanism=SCRAM-SHA-256&retryWrites=false&maxIdleTimeMS=120000'
export Database=testdb
export Container=testcontainer
export Index=testindex
export EmbeddingDeploymentName=ada
```

Python 1.x

최신 pip 패키지를 설치합니다.`openai azure-identity`

Python

```
import os
from openai import AzureOpenAI
from azure.identity import DefaultAzureCredential, get_bearer_token_provider

endpoint = os.environ.get("AzureOpenAIEndpoint")
deployment = os.environ.get("ChatCompletionsDeploymentName")
connection_string = os.environ.get("ConnectionString")
database = os.environ.get("Database")
container = os.environ.get("Container")
index = os.environ.get("Index")
embedding_deployment_name = os.environ.get("EmbeddingDeploymentName")

token_provider = get_bearer_token_provider(
    DefaultAzureCredential(), "https://cognitiveservices.azure.com/.default"
)

client = AzureOpenAI(
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
    api_version="2024-02-01",
)

completion = client.chat.completions.create(
    model=deployment,
    messages=[
        {
            "role": "user",
            "content": "Who is DRI?",
        },
    ],
    extra_body={
        "data_sources": [
            {
                "type": "azure_cosmos_db",
                "parameters": {
                    "authentication": {
                        "type": "connection_string",
                        "connection_string": connection_string
                    },
                    "database_name": database,
                    "container_name": container,
                    "index_name": index,
                    "fields_mapping": {
                        "content_fields": [
                            "content"
                        ],
                        "vector_fields": [
                            "contentvector"
                        ]
                    },
                    "embedding_dependency": {
                        "type": "deployment_name",
                        "deployment_name": embedding_deployment_name
                    }
                }
            }
        ]
    }
)
```

```
        }
    ],
)

print(completion.model_dump_json(indent=2))
```

데이터 원본 - Azure Machine Learning 인덱스(미리 보기)

아티클 • 2024. 03. 20.

Azure OpenAI On Your Data를 사용하는 경우 Azure Machine Learning 인덱스의 구성 가능한 옵션입니다. 이 데이터 원본은 API 버전 `2024-02-15-preview`에서 지원됩니다.

[+] 테이블 확장

속성	Type	필수	설명
<code>parameters</code>	매개 변수	True	Azure Machine Learning 인덱스 구성 시 사용할 매개 변수입니다.
<code>type</code>	string	True	<code>azure_ml_index</code> 이어야 합니다.

매개 변수

[+] 테이블 확장

이름	Type	필수	설명
<code>project_resource_id</code>	string	True	Azure Machine Learning 프로젝트의 리소스 ID입니다.
<code>name</code>	string	True	Azure Machine Learning 인덱스 이름입니다.
<code>version</code>	string	True	Azure Machine Learning 인덱스의 버전입니다.

이름	Type	필수	설명
authentication	AccessTokenAuthenticationOptions 중 하나, SystemAssignedManagedIdentityAuthenticationOptions, UserAssignedManagedIdentityAuthenticationOptions	True	정의된 데이터 원본에 액세스 할 때 사용할 인증 방법입니다.
in_scope	부울 값	False	쿼리를 인덱싱 된 데이터 사용으로 제한해야 하는지 여부입니다. 기본값은 <code>True</code> 입니다.
role_information	string	False	응답을 생성할 때 참조해야 하는 컨텍스트와 작동 방식에 대한 지침을 모델에 제공합니다. 도우미 성격에 대해 설명하고 응답 형식을 정하는 방법을 알려줄 수 있습니다.
strictness	정수	False	검색 관련성 필

이름	Type	필수	설명
			터링의 구성된 엄격성입니다. 엄격성이 높을 수록 정밀도가 높지만 대답의 재현율이 낮습니다. 기본값은 3입니다.
<code>top_n_documents</code>	정수	False	구성된 쿼리에 대해 기능할 구성된 상위 문서 수입니다. 기본값은 5입니다.
<code>filter</code>	string	False	검색 필터입니다. Azure Machine Learning 인덱스가 Azure Search 형식인 경우에만 지원됩니다.

액세스 토큰 인증 옵션

액세스 토큰을 사용할 때 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

[+] 테이블 확장

속성	Type	필수	설명
access_token	string	True	인증에 사용할 액세스 토큰입니다.
type	string	True	access_token 이어야 합니다.

시스템 할당 관리 ID 인증 옵션

시스템 할당 관리 ID를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

[+] 테이블 확장

속성	Type	필수	설명
type	string	True	system_assigned_managed_identity 이어야 합니다.

사용자 할당 관리 ID 인증 옵션

사용자 할당 관리 ID를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

[+] 테이블 확장

속성	Type	필수	설명
managed_identity_resource_id	string	True	인증 시 이용할 사용자가 할당한 관리 ID의 리소스 ID입니다.
type	string	True	user_assigned_managed_identity 이어야 합니다.

예제

필수 조건:

- Azure OpenAI 시스템에서 할당된 관리 ID에서 Azure Machine Learning 작업 영역 리소스로 역할 할당을 구성합니다. 필수 역할: AzureML Data Scientist.
- 사용자에서 Azure OpenAI 리소스로 역할 할당을 구성합니다. 필수 역할: Cognitive Services OpenAI User.

- Az CLI를 설치하고 실행 `az login`합니다.
- 다음 환경 변수를 정의합니다. `AzureOpenAIEndpoint`, `ChatCompletionsDeploymentName`, `ProjectResourceId`, `IndexName`, `IndexVersion`.
- MINGW를 사용하는 경우 실행 `export MSYS_NO_PATHCONV=1` 합니다.

Bash

```
export AzureOpenAIEndpoint=https://example.openai.azure.com/
export ChatCompletionsDeploymentName=turbo
export ProjectResourceId='/subscriptions/{subscription-
id}/resourceGroups/{resource-group-
name}/providers/Microsoft.MachineLearningServices/workspaces/{workspace-id}'
export IndexName=testamlindex
export IndexVersion=2
```

Python 1.x

최신 pip 패키지를 설치합니다. `openai azure-identity`

Python

```
import os
from openai import AzureOpenAI
from azure.identity import DefaultAzureCredential,
get_bearer_token_provider

endpoint = os.environ.get("AzureOpenAIEndpoint")
deployment = os.environ.get("ChatCompletionsDeploymentName")
project_resource_id = os.environ.get("ProjectResourceId")
index_name = os.environ.get("IndexName")
index_version = os.environ.get("IndexVersion")

token_provider = get_bearer_token_provider(
    DefaultAzureCredential(),
    "https://cognitiveservices.azure.com/.default")

client = AzureOpenAI(
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
    api_version="2024-02-15-preview",
)

completion = client.chat.completions.create(
    model=deployment,
    messages=[
        {
            "role": "user",
            "content": "Who is DRI?",
        },
    ],
)
```

```
extra_body={  
    "data_sources": [  
        {  
            "type": "azure_ml_index",  
            "parameters": {  
                "project_resource_id": project_resource_id,  
                "name": index_name,  
                "version": index_version,  
                "authentication": {  
                    "type": "system_assigned_managed_identity"  
                },  
            }  
        }  
    ]  
}  
  
print(completion.model_dump_json(indent=2))
```

데이터 원본 - Elasticsearch(미리 보기)

아티클 · 2024. 03. 18.

데이터에서 Azure OpenAI를 사용하는 경우 Elasticsearch에 대한 구성 가능한 옵션입니다. 이 데이터 원본은 API 버전 2024-02-15-preview에서 지원됩니다.

 테이블 확장

속성	Type	필수	설명
parameters	매개 변수	True	Elasticsearch를 구성할 때 사용할 매개 변수입니다.
type	string	True	elasticsearch이어야 합니다.

매개 변수

 테이블 확장

이름	Type	필수	설명
endpoint	string	True	사용할 Elasticsearch 리소스의 절대 엔드포인트 경로입니다.
index_name	string	True	참조된 Elasticsearch에서 사용할 인덱스의 이름입니다.
authentication	KeyAndKeyIdAuthenticationOptions 중 하나인 EncodedApiKeyAuthenticationOptions	True	정의된 데이터 원본에 액세스할 때 사용할 인증 방법입니다.
embedding_dependency	DeploymentNameVectorizationSource, EndpointVectorizationSource, ModelIdVectorizationSource 중 하나	False	벡터 검색에 포함되는 종속성입니다. 필요한 경우 <code>query_type .vector</code>
fields_mapping	FieldsMappingOptions	False	검색 인덱스와 상호 작용할 때 사용할 사용자 지정된 필드 매핑 동작입니다.
in_scope	부울 값	False	쿼리를 인덱싱된 데이터 사용으로 제한해야 하는지 여부입니다. 기본값은 True입니다.
query_type	QueryType	False	Elasticsearch와 함께 사용할 쿼리 형식입니다. 기본값은 simple
role_information	string	False	응답을 생성할 때 참조해야 하는 컨텍스트와 작동 방식에 대한 지침을 모델에 제공합니다. 도우미 성격에 대해 설명 하고 응답 형식을 지정하는 방법을 알려줄 수 있습니다.
strictness	정수	False	검색 관련성 필터링의 구성된 엄격성입니다. 엄격성이 높을 수록 정밀도가 높지만 대답의 재현율이 낮습니다. 기본값은 3입니다.
top_n_documents	정수	False	구성된 쿼리에 대해 기능할 구성된 상위 문서 수입니다. 기 본값은 5입니다.

키 및 키 ID 인증 옵션

API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 테이블 확장

속성	Type	필수	설명
key	string	True	인증에 사용할 Elasticsearch 키입니다.
key_id	string	True	인증에 사용할 Elasticsearch 키 ID입니다.
type	string	True	key_and_key_id이어야 합니다.

인코딩된 API 키 인증 옵션

Elasticsearch로 인코딩된 API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 테이블 확장

속성	Type	필수	설명
encoded_api_key	string	True	인증에 사용할 Elasticsearch로 인코딩된 API 키입니다.
type	string	True	encoded_api_key 이어야 합니다.

배포 이름 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 동일한 Azure OpenAI 리소스의 내부 포함 모델 배포 이름을 기반으로 합니다. 이 벡터화 원본을 사용하면 Azure OpenAI api-key 없이 Azure OpenAI 공용 네트워크 액세스 없이 벡터 검색을 사용할 수 있습니다.

 테이블 확장

속성	Type	필수	설명
deployment_name	string	True	동일한 Azure OpenAI 리소스 내의 포함 모델 배포 이름입니다.
type	string	True	deployment_name 이어야 합니다.

엔드포인트 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 Azure OpenAI 포함 API 엔드포인트를 기반으로 합니다.

 테이블 확장

속성	Type	필수	설명
endpoint	string	True	포함을 검색할 리소스 엔드포인트 URL을 지정합니다. 형식이어야 <code>https://{{YOUR_RESOURCE_NAME}}.openai.azure.com/openai/deployments/{{YOUR_DEPLOYMENT_NAME}}/embeddings</code> 입니다. api-version 쿼리 매개 변수는 허용되지 않습니다.
authentication	ApiKeyAuthenticationOptions	True	지정된 엔드포인트에서 포함을 검색할 때 사용할 인증 옵션을 지정합니다.
type	string	True	endpoint 이어야 합니다.

모델 ID 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 Elasticsearch 모델 ID를 기반으로 합니다.

 테이블 확장

속성	Type	필수	설명
model_id	string	True	벡터화에 사용할 모델 ID를 지정합니다. 이 모델 ID는 Elasticsearch에서 정의해야 합니다.
type	string	True	model_id 이어야 합니다.

API 키 인증 옵션

API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

 테이블 확장

속성	Type	필수	설명
key	string	True	인증에 사용할 API 키입니다.
type	string	True	api_key 이어야 합니다.

필드 매팅 옵션

구성된 Elasticsearch 리소스를 사용할 때 필드가 처리되는 방식을 제어하는 선택적 설정입니다.

 테이블 확장

속성	Type	필수	설명
content_fields	string[]	False	콘텐츠로 처리해야 하는 인덱스 필드의 이름입니다.
vector_fields	string[]	False	벡터 데이터를 나타내는 필드의 이름입니다.
content_fields_separator	string	False	콘텐츠 필드에서 사용해야 하는 구분 기호 패턴입니다. 기본값은 \n 입니다.
filepath_field	string	False	파일 경로로 사용할 인덱스 필드의 이름입니다.
title_field	string	False	제목으로 사용할 인덱스 필드의 이름입니다.
url_field	string	False	URL로 사용할 인덱스 필드의 이름입니다.

쿼리 유형

Azure OpenAI On Your Data와 함께 사용할 때 실행해야 하는 Elasticsearch 검색 쿼리의 형식입니다.

 테이블 확장

열거형 값	설명
simple	기본 단순 쿼리 파서입니다.
vector	계산된 데이터에 대한 벡터 검색을 나타냅니다.

예제

필수 조건:

- 사용자에서 Azure OpenAI 리소스로 역할 할당을 구성합니다. 필수 역할: Cognitive Services OpenAI User.
- Az CLI를 설치하고 실행 az login 합니다.
- 다음 환경 변수를 정의합니다. AzureOpenAIEndpoint ChatCompletionsDeploymentName SearchEndpoint IndexName Key KeyId

Bash

```
export AzureOpenAIEndpoint=https://example.openai.azure.com/
export ChatCompletionsDeploymentName=turbo
export SearchEndpoint='https://example.eastus.azurecontainer.io'
export IndexName=testindex
export Key='****'
export KeyId='****'
```

Python 1.x

최신 pip 패키지를 설치합니다. openai azure-identity

Python

```
import os
from openai import AzureOpenAI
from azure.identity import DefaultAzureCredential, get_bearer_token_provider

endpoint = os.environ.get("AzureOpenAIEndpoint")
```

```
deployment = os.environ.get("ChatCompletionsDeploymentName")
index_name = os.environ.get("IndexName")
search_endpoint = os.environ.get("SearchEndpoint")
key = os.environ.get("Key")
key_id = os.environ.get("KeyId")

token_provider = get_bearer_token_provider(
    DefaultAzureCredential(), "https://cognitiveservices.azure.com/.default")

client = AzureOpenAI(
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
    api_version="2024-02-15-preview",
)

completion = client.chat.completions.create(
    model=deployment,
    messages=[
        {
            "role": "user",
            "content": "Who is DRI?",
        },
    ],
    extra_body={
        "data_sources": [
            {
                "type": "elasticsearch",
                "parameters": {
                    "endpoint": search_endpoint,
                    "index_name": index_name,
                    "authentication": {
                        "type": "key_and_key_id",
                        "key": key,
                        "key_id": key_id
                    }
                }
            }
        ]
    }
)

print(completion.model_dump_json(indent=2))
```

데이터 원본 - Pinecone(미리 보기)

아티클 • 2024. 03. 18.

데이터에서 Azure OpenAI를 사용하는 경우 Pinecone의 구성 가능한 옵션입니다. 이 데이터 원본은 API 버전 2024-02-15-preview에서 지원됩니다.

[+] 테이블 확장

속성	Type	필수	설명
parameters	매개 변수	True	Pinecone을 구성할 때 사용할 매개 변수입니다.
type	string	True	pinecone이어야 합니다.

매개 변수

[+] 테이블 확장

이름	Type	필수	설명
environment	string	True	Pinecone의 환경 이름입니다.
index_name	string	True	Pinecone 데이터베이스 인덱스의 이름입니다.
fields_mapping	FieldsMappingOptions	True	검색 인덱스와 상호 작용 할 때 사용할 사용자 지정 된 필드 매팅 동작입니다.
authentication	ApiKeyAuthenticationOptions	True	정의된 데이터 원본에 액세스할 때 사용할 인증 방법입니다.
embedding_dependency	DeploymentNameVectorizationSource	True	벡터 검색에 포함되는 종속성입니다.
in_scope	부울 값	False	쿼리를 인덱싱된 데이터 사용으로 제한해야 하는지 여부입니다. 기본값은 True입니다.
role_information	string	False	응답을 생성할 때 참조해야 하는 컨텍스트와 작동 방식에 대한 지침을 모델에 제공합니다. 도우미 성

이름	Type	필수	설명
			격에 대해 설명하고 응답 형식을 지정하는 방법을 알려줄 수 있습니다.
<code>strictness</code>	정수	False	검색 관련성 필터링의 구성된 엄격성입니다. 엄격성이 높을수록 정밀도가 높지만 대답의 재현율이 낮습니다. 기본값은 3입니다.
<code>top_n_documents</code>	정수	False	구성된 쿼리에 대해 가능할 구성된 상위 문서 수입니다. 기본값은 5입니다.

API 키 인증 옵션

API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

[+] 테이블 확장

속성	Type	필수	설명
<code>key</code>	string	True	인증에 사용할 API 키입니다.
<code>type</code>	string	True	<code>api_key</code> 이어야 합니다.

배포 이름 벡터화 원본

벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 이 벡터화 원본은 동일한 Azure OpenAI 리소스의 내부 포함 모델 배포 이름을 기반으로 합니다. 이 벡터화 원본을 사용하면 Azure OpenAI api-key 없이 Azure OpenAI 공용 네트워크 액세스 없이 벡터 검색을 사용할 수 있습니다.

[+] 테이블 확장

속성	Type	필수	설명
<code>deployment_name</code>	string	True	동일한 Azure OpenAI 리소스 내의 포함 모델 배포 이름입니다.
<code>type</code>	string	True	<code>deployment_name</code> 이어야 합니다.

필드 맵핑 옵션

필드 처리 방법을 제어하는 설정입니다.

 테이블 확장

속성	Type	필수	설명
content_fields	string[]	True	콘텐츠로 처리해야 하는 인덱스 필드의 이름입니다.
content_fields_separator	string	False	콘텐츠 필드에서 사용해야 하는 구분 기호 패턴입니다. 기본값은 \n입니다.
filepath_field	string	False	파일 경로로 사용할 인덱스 필드의 이름입니다.
title_field	string	False	제목으로 사용할 인덱스 필드의 이름입니다.
url_field	string	False	URL로 사용할 인덱스 필드의 이름입니다.

예제

필수 조건:

- 사용자에서 Azure OpenAI 리소스로 역할 할당을 구성합니다. 필수 역할: Cognitive Services OpenAI User.
- Az CLI를 설치하고 실행 az login 합니다.
- 다음 환경 변수를 정의합니다.
AzureOpenAIEndpoint ChatCompletionsDeploymentName Environment IndexName Key EmbeddingDeploymentName

Bash

```
export AzureOpenAIEndpoint=https://example.openai.azure.com/
export ChatCompletionsDeploymentName=turbo
export Environment=testenvironment
export Key=***
export IndexName=pinecone-test-index
export EmbeddingDeploymentName=ada
```

Python 1.x

최신 pip 패키지를 설치합니다. openai azure-identity

Python

```
import os
from openai import AzureOpenAI
from azure.identity import DefaultAzureCredential,
get_bearer_token_provider

endpoint = os.environ.get("AzureOpenAIEndpoint")
deployment = os.environ.get("ChatCompletionsDeploymentName")
environment = os.environ.get("Environment")
key = os.environ.get("Key")
index_name = os.environ.get("IndexName")
embedding_deployment_name = os.environ.get("EmbeddingDeploymentName")

token_provider = get_bearer_token_provider(
    DefaultAzureCredential(),
    "https://cognitiveservices.azure.com/.default")

client = AzureOpenAI(
    azure_endpoint=endpoint,
    azure_ad_token_provider=token_provider,
    api_version="2024-02-15-preview",
)

completion = client.chat.completions.create(
    model=deployment,
    messages=[
        {
            "role": "user",
            "content": "Who is DRI?",
        },
    ],
    extra_body={
        "data_sources": [
            {
                "type": "pinecone",
                "parameters": {
                    "environment": environment,
                    "authentication": {
                        "type": "api_key",
                        "key": key
                    },
                    "index_name": index_name,
                    "fields_mapping": {
                        "content_fields": [
                            "content"
                        ]
                    },
                    "embedding_dependency": {
                        "type": "deployment_name",
                        "deployment_name": embedding_deployment_name
                    }
                }
            }
        ],
    }
)
```

```
print(completion.model_dump_json(indent=2))
```

Ingestion Jobs

참조

Service: Azure AI Services

API Version: 2023-10-01-preview

Operations

 테이블 확장

Create 완료를 위해 데이터 원본으로 사용할 Azure Search 인덱스에 데이터를 수집하는 작업을 시작합니다. 수집 작업의 상태 48시간 동안 유지됩니다...

Get 수집 작업의 상태 ID로 가져옵니다.

List 페이지를 매긴 컬렉션의 형태로 지난 48시간 동안 실행된 모든 수집 작업의 상태 Lists.

Microsoft.CognitiveServices 계정

아티클 • 2023. 06. 19.

Bicep 리소스 정의

다음을 대상으로 하는 작업으로 계정 리소스 유형을 배포할 수 있습니다.

- 리소스 그룹 - 리소스 그룹 배포 명령 참조

각 API 버전에서 변경된 속성 목록은 [변경 로그](#)를 참조하세요.

리소스 형식

Microsoft.CognitiveServices/accounts 리소스를 만들려면 템플릿에 다음 Bicep을 추가합니다.

```
Bicep

resource symbolicname 'Microsoft.CognitiveServices/accounts@2023-05-01' = {
    name: 'string'
    location: 'string'
    tags: {
        tagName1: 'tagValue1'
        tagName2: 'tagValue2'
    }
    sku: {
        capacity: int
        family: 'string'
        name: 'string'
        size: 'string'
        tier: 'string'
    }
    kind: 'string'
    identity: {
        type: 'string'
        userAssignedIdentities: {}
    }
    properties: {
        allowedFqdnList: [
            'string'
        ]
        apiProperties: {
            aadClientId: 'string'
            aadTenantId: 'string'
            eventHubConnectionString: 'string'
            qnaAzureSearchEndpointId: 'string'
            qnaAzureSearchEndpointKey: 'string'
            qnaRuntimeEndpoint: 'string'
            statisticsEnabled: bool
            storageAccountConnectionString: 'string'
            superUser: 'string'
            websiteName: 'string'
        }
        customSubDomainName: 'string'
        disableLocalAuth: bool
        dynamicThrottlingEnabled: bool
        encryption: {
            keySource: 'string'
            keyVaultProperties: {
                identityClientId: 'string'
                keyName: 'string'
                keyVaultUri: 'string'
                keyVersion: 'string'
            }
        }
        locations: {
            regions: [
                {
                    customsubdomain: 'string'
                    name: 'string'
                    value: int
                }
            ]
            routingMethod: 'string'
        }
        migrationToken: 'string'
    }
}
```

```

networkAcls: {
  defaultAction: 'string'
  ipRules: [
    {
      value: 'string'
    }
  ]
  virtualNetworkRules: [
    {
      id: 'string'
      ignoreMissingVnetServiceEndpoint: bool
      state: 'string'
    }
  ]
}
publicNetworkAccess: 'string'
restore: bool
restrictOutboundNetworkAccess: bool
userOwnedStorage: [
  {
    identityClientId: 'string'
    resourceId: 'string'
  }
]
}
}

```

속성 값

계정

속성	Description	값
name	리소스 이름	string(필수) 문자 제한: 2-64 유효한 문자: 영숫자 및 하이픈 영숫자로 시작하고 끝납니다.
위치	리소스가 있는 지리적 위치	문자열
tags	리소스 태그.	태그 이름 및 값의 사전입니다. 템플릿의 태그를 참조하세요.
sku	SKU를 나타내는 리소스 모델 정의	Sku
kind	리소스 종류입니다.	문자열
identity	리소스의 ID입니다.	ID
properties	Cognitive Services 계정의 속성입니다.	AccountProperties

ID

이름	Description
형식	ID 유형입니다.
userAssignedIdentities	리소스와 연결된 사용자 할당 ID 목록입니다. 사용자 ID 사전 키 참조는 '/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.ManagedIdentity/userAssignedIdentities/{identity}' 형식의 ARM 리소스 ID입니다.

AccountProperties

속성	Description	값

속성	Description	값
allowedFqdnList		string[]
apiProperties	특수 API에 대한 api 속성입니다.	ApiProperties
customSubDomainName	토큰 기반 인증에 사용되는 선택적 하위 도메인 이름입니다.	문자열
disableLocalAuth		bool
dynamicThrottlingEnabled	동적 제한을 사용하도록 설정하는 플래그입니다.	bool
암호화	이 리소스에 대한 암호화 속성입니다.	암호화
위치	Cognitive Services 계정의 다중Region 설정입니다.	MultiRegionSettings
migrationToken	리소스 마이그레이션 토큰.	문자열
networkAcls	특정 네트워크 위치의 접근성을 제어하는 규칙 컬렉션입니다.	NetworkRuleSet
publicNetworkAccess	이 계정에 대해 퍼블릭 앤드포인트 액세스가 허용되는지 여부입니다.	'사용 안 함' '사용'
복원		bool
restrictOutboundNetworkAccess		bool
userOwnedStorage	이 리소스에 대한 스토리지 계정입니다.	UserOwnedStorage[]

ApiProperties

속성	Description	값
aadClientId	(Metrics Advisor만 해당) Azure AD 클라이언트 ID(애플리케이션 ID)입니다.	문자열
aadTenantId	(Metrics Advisor만 해당) Azure AD 테넌트 ID입니다.	문자열
eventHubConnectionString	(개인 설정만 해당) Bing Search의 통계를 사용하도록 설정하는 플래그입니다.	문자열
qnaAzureSearchEndpointId	(QnAMaker만 해당) QnAMaker의 Azure Search 엔드포인트 ID입니다.	문자열
qnaAzureSearchEndpointKey	(QnAMaker만 해당) QnAMaker의 Azure Search 엔드포인트 키입니다.	문자열
qnaRuntimeEndpoint	(QnAMaker만 해당) QnAMaker의 런타임 엔드포인트입니다.	문자열
statisticsEnabled	(Bing Search만 해당) Bing Search의 통계를 사용하도록 설정하는 플래그입니다.	bool
storageAccountConnectionString	(개인 설정만 해당) 스토리지 계정 연결 문자열입니다.	문자열
수퍼유저	(Metrics Advisor만 해당) Metrics Advisor의 슈퍼 사용자입니다.	문자열
websiteName	(Metrics Advisor만 해당) Metrics Advisor의 웹 사이트 이름입니다.	문자열

암호화

속성	Description	값
keySource	암호화에 사용할 수 있는 keySource 값을 열거합니다.	'Microsoft.CognitiveServices' 'Microsoft.KeyVault'
keyVaultProperties	KeyVault의 속성	KeyVaultProperties

KeyVaultProperties

속성	Description	값
identityClientId		문자열
keyName	KeyVault의 키 이름	문자열
keyVaultUri	KeyVault의 URI	문자열
keyVersion	KeyVault의 키 버전	문자열

MultiRegionSettings

속성	Description	값
regions		RegionSetting[]
routingMethod	다중region 라우팅 메서드.	'성능' '우선 순위' '가중치'

RegionSetting

속성	Description	값
customsubdomain	지역을 지역 사용자 지정 하위 도메인에 매핑합니다.	문자열
name	지역의 이름입니다.	문자열
값	우선 순위 또는 가중 라우팅 메서드에 대한 값입니다.	int

NetworkRuleSet

속성	Description	값
defaultAction	ipRules 및 virtualNetworkRules의 규칙이 일치하지 않는 경우의 기본 작업입니다. 바이패스 속성이 평가된 후에만 사용됩니다.	'허용' 'Deny'
ipRules	IP 주소 규칙 목록입니다.	IpRule[]
virtualNetworkRules	가상 네트워크 규칙 목록입니다.	VirtualNetworkRule[]

IpRule

속성	Description	값
값	CIDR 표기법의 IPv4 주소 범위(예: '124.56.78.91'(단순 IP 주소) 또는 '124.56.78.0/24'(124.56.78로 시작하는 모든 주소).	string(필수)

VirtualNetworkRule

속성	Description	값
id	'/subscriptions/subid/resourceGroups/rg1/providers/Microsoft.Network/virtualNetworks/test-vnet/subnets/subnet1'과 같은 vnet 서브넷의 전체 리소스 ID입니다.	string(필수)
ignoreMissingVnetServiceEndpoint	누락된 vnet 서비스 엔드포인트를 무시합니다.	bool
state	가상 네트워크 규칙의 상태를 가져옵니다.	문자열

UserOwnedStorage

속성	Description	값
identityClientId		문자열
resourceId	Microsoft.Storage 리소스의 전체 리소스 ID입니다.	문자열

SKU

속성	Description	값
용량	SKU가 스케일 아웃/인을 지원하는 경우 용량 정수가 포함되어야 합니다. 리소스에 대해 규모 확장/감축이 불가능한 경우 생략할 수 있습니다.	int
family	서비스에 동일한 SKU에 대해 여러 세대의 하드웨어가 있는 경우 여기에서 캡처할 수 있습니다.	문자열
name	SKU의 이름입니다. 예 - P3. 일반적으로 letter+number 코드입니다.	string(필수)

속성	Description	값
크기	SKU 크기입니다. 이름 필드가 계층과 다른 값의 조합인 경우 독립 실행형 코드입니다.	문자열
계층	이 필드는 서비스에 둘 이상의 계층이 있지만 PUT에 필요하지 않은 경우 리소스 공급자가 구현해야 합니다.	'기본' 'Enterprise' '무료' '프리미엄' '표준'

빠른 시작 템플릿

다음 빠른 시작 템플릿은 이 리소스 유형을 배포합니다.

템플릿	Description
Cognitive Services Computer Vision API 배포 ↗	새 Cognitive Services Computer Vision API를 만들기 위한 템플릿
 Deploy to Azure ↗	이 템플릿은 Cognitive Services Translate API를 배포합니다. Microsoft Translator API는 개발자가 웹 사이트 지역화, 전자 상거래, 고객 지원, 메시징 애플리케이션, 내부 통신 등과 같은 다국어 지원이 필요한 애플리케이션 웹 사이트, 도구 또는 솔루션에 쉽게 통합할 수 있는 신경망 기계 번역 서비스입니다.
Cognitive Services Translate API 배포 ↗	이 템플릿은 기계 학습 전문 지식 없이도 Cognitive Services가 모든 개발자의 손이 닿는 범위 내에 AI를 가져오는 모든 Cognitive Services API 배포합니다. 앱에 의사 결정을 보고, 듣고, 말하고, 검색하고, 이해하고, 가속화하는 기능을 포함하는 API 호출만 있으면 됩니다.
 Deploy to Azure ↗	

az cognitiveservices

참조

Azure Cognitive Services 계정을 관리합니다.

이 문서에는 Azure Cognitive Services 계정 및 구독 관리에 대한 Azure CLI 명령만 나열되어 있습니다. API 및 지원되는 SDK를 사용하는 방법을 알아보려면 개별 서비스에 대한 설명서를 <https://docs.microsoft.com/azure/cognitive-services/> 참조하세요.

명령

 테이블 확장

Name	Description	형식	상태
az cognitiveservices account	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account commitment-plan	Azure Cognitive Services 계정에 대한 약정 계획을 관리합니다.	핵심	GA
az cognitiveservices account commitment-plan create	Azure Cognitive Services 계정에 대한 약정 계획을 만듭니다.	핵심	GA
az cognitiveservices account commitment-plan delete	Azure Cognitive Services 계정에서 약정 계획을 삭제합니다.	핵심	GA
az cognitiveservices account commitment-plan list	Azure Cognitive Services 계정의 모든 약정 계획을 표시합니다.	핵심	GA
az cognitiveservices account commitment-plan show	Azure Cognitive Services 계정에서 약정 계획을 표시합니다.	핵심	GA
az cognitiveservices account create	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account delete	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account deployment	Azure Cognitive Services 계정에 대한 배포를 관리합니다.	핵심	GA
az cognitiveservices account deployment create	Azure Cognitive Services 계정에 대한 배포를 만듭니다.	핵심	GA

Name	Description	형식	상태
az cognitiveservices account deployment delete	Azure Cognitive Services 계정에서 배포를 삭제합니다.	핵심	GA
az cognitiveservices account deployment list	Azure Cognitive Services 계정에 대한 모든 배포를 표시합니다.	핵심	GA
az cognitiveservices account deployment show	Azure Cognitive Services 계정에 대한 배포를 표시합니다.	핵심	GA
az cognitiveservices account identity	Cognitive Services 계정의 ID를 관리합니다.	핵심	GA
az cognitiveservices account identity assign	Cognitive Services 계정의 ID를 할당합니다.	핵심	GA
az cognitiveservices account identity remove	Cognitive Services 계정에서 ID를 제거합니다.	핵심	GA
az cognitiveservices account identity show	Cognitive Services 계정의 ID를 표시합니다.	핵심	GA
az cognitiveservices account keys	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account keys list	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account keys regenerate	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account list	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account list-deleted	일시 삭제된 Azure Cognitive Services 계정을 나열합니다.	핵심	GA
az cognitiveservices account list-kinds	Azure Cognitive Services 계정에 유료한 모든 종류를 나열합니다.	핵심	GA
az cognitiveservices account list-models	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account list-skus	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account list-usage	Azure Cognitive Services 계정에 대한 사용량을 나열합니다.	핵심	GA

Name	Description	형식	상태
az cognitiveservices account network-rule	네트워크 규칙을 관리합니다.	핵심	GA
az cognitiveservices account network-rule add	네트워크 규칙을 추가합니다.	핵심	GA
az cognitiveservices account network-rule list	네트워크 규칙을 나열합니다.	핵심	GA
az cognitiveservices account network-rule remove	네트워크 규칙을 제거합니다.	핵심	GA
az cognitiveservices account purge	일시 삭제된 Azure Cognitive Services 계정을 제거합니다.	핵심	GA
az cognitiveservices account recover	일시 삭제된 Azure Cognitive Services 계정을 복구합니다.	핵심	GA
az cognitiveservices account show	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices account show-deleted	일시 삭제된 Azure Cognitive Services 계정을 표시합니다.	핵심	GA
az cognitiveservices account update	Azure Cognitive Services 계정을 관리합니다.	핵심	GA
az cognitiveservices commitment-tier	Azure Cognitive Services에 대한 약정 계층을 관리합니다.	핵심	GA
az cognitiveservices commitment-tier list	Azure Cognitive Services에 대한 모든 약정 계층을 표시합니다.	핵심	GA
az cognitiveservices list	Azure Cognitive Services 계정을 관리합니다.	핵심	더 이상 사용되지 않음
az cognitiveservices model	Azure Cognitive Services에 대한 모델을 관리합니다.	핵심	GA
az cognitiveservices model list	Azure Cognitive Services에 대한 모든 모델을 표시합니다.	핵심	GA
az cognitiveservices usage	Azure Cognitive Services에 대한 사용량을 관리합니다.	핵심	GA
az cognitiveservices usage list	Azure Cognitive Services에 대한 모든 사용량을 표시합니다.	핵심	GA

az cognitiveservices list

 편집

사용되지 않음

이 명령은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다. 대신 'az cognitiveservices 계정 목록'을 사용합니다.

Azure Cognitive Services 계정을 관리합니다.

이 문서에는 Azure Cognitive Services 계정 및 구독 관리에 대한 Azure CLI 명령만 나열되어 있습니다. API 및 지원되는 SDK를 사용하는 방법을 알아보려면 개별 서비스에 대한 설명서를 <https://docs.microsoft.com/azure/cognitive-services/> 참조하세요.

Azure CLI

```
az cognitiveservices list [--resource-group]
```

예제

리소스 그룹의 모든 Cognitive Services 계정을 나열합니다.

Azure CLI

```
az cognitiveservices list -g MyResourceGroup
```

선택적 매개 변수

--resource-group -g

리소스 그룹의 이름입니다. 를 사용하여 `az configure --defaults group=<name>` 기본 그룹을 구성할 수 있습니다.

▼ 전역 매개 변수

--debug

로깅의 자세한 정도를 늘려 모든 디버그 로그를 표시합니다.

--help -h

이 도움말 메시지를 표시하고 종료합니다.

--only-show-errors

경고를 표시하지 않고 오류만 표시합니다.

--output -o

출력 형식입니다.

허용되는 값: json, jsonc, none, table, tsv, yaml, yamlc

기본값: json

--query

JMESPath 쿼리 문자열입니다. 자세한 내용과 예제는 <http://jmespath.org/> 를 참조하세요.

--subscription

구독의 이름 또는 ID입니다. 를 사용하여 `az account set -s NAME_OR_ID` 기본 구독을 구성할 수 있습니다.

--verbose

로깅의 자세한 정도를 늘립니다. 전체 디버그 로그를 표시하려면 --debug를 사용합니다.

com.azure.ai.openai

Reference

Package: com.azure.ai.openai

Maven Artifact: [com.azure:azure-ai-openai:1.0.0-beta.7](#)

Package containing the classes for OpenAI. Azure OpenAI APIs for completions and search.

Classes

[] Expand table

OpenAIAsyncClient	Initializes a new instance of the asynchronous OpenAIClient type.
OpenAIClient	Initializes a new instance of the synchronous OpenAIClient type.
OpenAIClientBuilder	A builder for creating a new instance of the OpenAIClient type.

Enums

[] Expand table

OpenAIServiceVersion	Service version of OpenAIClient.
--------------------------------------	----------------------------------

 Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure SDK for Java feedback

Azure SDK for Java is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

@azure/openai package

참조

클래스

[+] 테이블 확장

AzureKey
Credential

기본 키 값 업데이트를 지원하는 정적 키 기반 자격 증명입니다.

OpenAIClient

Azure OpenAI와 상호 작용하기 위한 클라이언트입니다.

클라이언트에는 OpenAI 리소스의 엔드포인트와 API 키 또는 토큰과 같은 인증 방법이 필요합니다. API 키 및 엔드포인트는 OpenAI 리소스 페이지에서 찾을 수 있습니다. 리소스의 키 및 엔드포인트 페이지에 있습니다.

인증 예제:

API 키

JavaScript

```
import { OpenAIClient } from "@azure/openai";
import { AzureKeyCredential } from "@azure/core-auth";

const endpoint = "<azure endpoint>";
const credential = new AzureKeyCredential("<api key>");

const client = new OpenAIClient(endpoint, credential);
```

Azure Active Directory

JavaScript

```
import { OpenAIClient } from "@azure/openai";
import { DefaultAzureCredential } from "@azure/identity";

const endpoint = "<azure endpoint>";
const credential = new DefaultAzureCredential();

const client = new OpenAIClient(endpoint, credential);
```

OpenAIKeyCredential	OpenAIKeyCredential 클래스는 OpenAI API 키를 나타내며 OpenAI 엔드포인트에 대한 OpenAI 클라이언트에 인증하는 데 사용됩니다.
-------------------------------------	--

인터페이스

 테이블 확장

AudioResultSimpleJson	간단한 전사 응답
AudioResultVerboseJson	전사 응답.
AudioSegment	전사 세그먼트.
AzureChatEnhancementConfiguration	사용 가능한 Azure OpenAI 향상된 구성의 표현입니다.
AzureChatEnhancements	요청에 제공된 일치 입력을 통해 구성된 대로 채팅 완료에 대한 Azure 개선 사항의 출력 결과를 나타냅니다.
AzureChatExtensionConfiguration	단일 Azure OpenAI 채팅 확장에 대한 구성 데이터의 표현입니다. 이는 Azure OpenAI 채팅 확장을 사용하여 응답 동작을 보강해야 하는 채팅 완료 요청에 사용됩니다. 이 구성의 사용은 Azure OpenAI와만 호환됩니다.
AzureChatExtensionDataSourceResponseCitation	Azure OpenAI 채팅 확장 프로그램이 해당 채팅 완료 응답 생성에 관련된 경우 사용할 수 있는 추가 컨텍스트 정보의 단일 instance. 이 컨텍스트 정보는 일치하는 확장을 사용하도록 구성된 Azure OpenAI 요청을 사용하는 경우에만 채워집니다.
AzureChatExtensionsMessageContext	Azure OpenAI 채팅 확장이 해당 채팅 완료 응답 생성에 관련될 때 사용할 수 있는 추가 컨텍스트 정보의 표현입니다. 이 컨텍스트 정보는 일치하는 확장을 사용하도록 구성된 Azure OpenAI 요청을 사용하는 경우에만 채워집니다.
AzureChatGroundingEnhancementConfiguration	Azure OpenAI 접지 향상에 사용할 수 있는 옵션의 표현입니다.
AzureChatOCREnhancementConfiguration	Azure OpenAI OCR(광학 문자 인식) 향상에 사용할 수 있는 옵션의 표현입니다.
AzureCosmosDBChatExtensionConfiguration	Azure OpenAI 채팅 확장으로 사용할 때 Azure Cosmos DB에 대한 구성 가능한 옵션의 특정 표현입니다.
AzureCosmosDBFieldMappingOptions	구성된 Azure Cosmos DB 리소스를 사용할 때 필드가 처리되는 방식을 제어하는 선택적 설정입니다.
AzureExtensionsOptions	Azure OpenAI 채팅 확장에 대한 옵션입니다.

AzureGrounding Enhancement	이미지에서 검색된 개체의 경계 상자를 반환하는 접지 향상 기능입니다.
AzureGrounding EnhancementCoordinate Point	Azure 접지 향상에서 사용하는 단일 다각형 지점의 표현입니다.
AzureGrounding EnhancementLine	단어 및 선택 표시와 같은 인접한 콘텐츠 요소 시퀀스로 구성된 콘텐츠 줄 개체입니다.
AzureGrounding EnhancementLineSpan	검색된 개체 및 경계 상자 정보를 나타내는 span 개체입니다.
AzureMachineLearningIndex ChatExtensionConfiguration	Azure OpenAI 채팅 확장으로 사용할 때 Azure Machine Learning 벡터 인덱스의 구성 가능한 옵션에 대한 특정 표현입니다.
AzureSearchChatExtension Configuration	Azure OpenAI 채팅 확장으로 사용할 때 Azure Search 구성 가능한 옵션의 특정 표현입니다.
AzureSearchIndexField MappingOptions	구성된 Azure Search 리소스를 사용할 때 필드가 처리되는 방식을 제어하는 선택적 설정입니다.
ChatChoice	전체 채팅 완료 요청의 일부로 단일 프롬프트 완료의 표현입니다. 일반적으로 <code>n</code> 기본값이 1인 제공된 프롬프트 별로 선택 항목이 생성됩니다. 토큰 제한 및 기타 설정은 생성된 선택 항목 수를 제한할 수 있습니다.
ChatChoiceLogProbability Info	'logprobs' 및 'top_logprobs'을 통해 요청된 선택 항목에 대한 확률 정보를 기록합니다.
ChatCompletions	채팅 완료 요청의 응답 데이터 표현입니다. 완료는 다양한 작업을 지원하고 제공된 프롬프트 데이터에서 계속되거나 "완료"되는 텍스트를 생성합니다.
ChatCompletionsFunction ToolCall	구성된 함수 도구를 평가하여 모델에서 실행한 함수 도구에 대한 도구 호출로, 후속 채팅 완료 요청이 resolve 데 필요한 함수 호출을 나타냅니다.
ChatCompletionsFunction ToolDefinition	도구 호출에 대한 응답으로 함수를 호출할 수 있는 채팅 완료 함수 도구에 대한 정의 정보입니다.
ChatCompletionsFunction ToolSelection	채팅 완료를 명명된 함수 사용으로 제한하는 명명된 특정 함수 도구의 도구 선택입니다.
ChatCompletionsJson ResponseFormat	응답을 유효한 JSON 개체 내보내기로 제한하는 채팅 완료에 대한 응답 형식입니다.
ChatCompletionsNamed FunctionToolSelection	채팅 완료를 명명된 함수 사용으로 제한하는 명명된 특정 함수 도구의 도구 선택입니다.
ChatCompletionsNamedTool	채팅 완료 요청에 사용할 명시적 명명된 도구 선택의 추상 표현입니다.

Selection	니다.
ChatCompletionsResponseFormat	채팅 완료에서 사용할 수 있는 응답 형식 구성의 추상 표현입니다. JSON 모드를 사용하도록 설정하는 데 사용할 수 있습니다.
ChatCompletionsTextResponseFormat	텍스트를 자유롭게 생성할 수 있고 특정 스키마를 준수하는 응답 콘텐츠를 생성하도록 보장되지 않는 표준 채팅 완료 응답 형식입니다.
ChatCompletionsToolCall	요청된 채팅 완료를 수행하기 위해 후속 요청에서 해결해야 하는 도구 호출의 추상 표현입니다.
ChatCompletionsToolDefinition	모델에서 채팅 완료 응답을 개선하는 데 사용할 수 있는 도구의 추상 표현입니다.
ChatFinishDetails	채팅 완료 응답이 종료된 이유에 대한 구조화된 정보의 추상 표현입니다.
ChatMessageContentItem	채팅 메시지 내에서 구조화된 콘텐츠 항목의 추상 표현입니다.
ChatMessageImageContentItem	이미지 참조를 포함하는 구조적 채팅 콘텐츠 항목입니다.
ChatMessageImageUrl	모델이 이미지를 검색할 수 있는 인터넷 위치입니다.
ChatMessageTextContentItem	일반 텍스트를 포함하는 구조화된 채팅 콘텐츠 항목입니다.
ChatRequestAssistantMessage	도우미 응답 또는 작업을 나타내는 요청 채팅 메시지입니다.
ChatRequestFunctionMessage	구성된 함수에서 요청된 출력을 나타내는 요청 채팅 메시지입니다.
ChatRequestMessage	요청에 제공된 채팅 메시지의 추상 표현입니다.
ChatRequestSystemMessage	모델이 채팅 완료 응답을 생성하는 방법에 영향을 주는 시스템 지침이 포함된 요청 채팅 메시지입니다.
ChatRequestToolMessage	구성된 도구에서 요청된 출력을 나타내는 요청 채팅 메시지입니다.
ChatRequestUserMessage	도우미 대한 사용자 입력을 나타내는 요청 채팅 메시지입니다.
ChatResponseMessage	응답에서 받은 채팅 메시지의 표현입니다.
ChatTokenLogProbabilityInfo	단일 메시지 콘텐츠 토큰에 대한 로그 확률 정보의 표현입니다.
ChatTokenLogProbabilityResult	'top_logprobs'이 요청된 경우 가장 가능성이 높은 토큰 목록을 포함하여 단일 콘텐츠 토큰에 대한 로그 확률 정보의 표현입니다.
Choice	전체 완료 요청의 일부로 단일 프롬프트 완료의 표현입니다. 일반적으로 <code>n</code> 기본값이 1인 제공된 프롬프트 별로 선택 항목이 생성됩니다.

	니다. 토큰 제한 및 기타 설정은 생성된 선택 항목 수를 제한할 수 있습니다.
Completions	완료 요청에서 응답 데이터의 표현입니다. 완료는 다양한 작업을 지원하고 제공된 프롬프트 데이터에서 계속되거나 "완료"되는 텍스트를 생성합니다.
CompletionsLogProbability Model	완료 생성에 대한 로그 확률 모델의 표현입니다.
CompletionsUsage	완료 요청에 대해 처리된 토큰 수의 표현입니다. 개수는 프롬프트, 선택 항목, 선택 대체 항목, best_of 세대 및 기타 소비자의 모든 토큰을 고려합니다.
ContentFilterBlocklistIdResult	콘텐츠 필터링에서 수행하는 사용자 지정 차단 목록에 대한 평가 결과를 나타냅니다.
ContentFilterCitedDetection Result	콘텐츠 필터링에 의해 수행되는 보호된 리소스에 대한 검색 작업의 결과를 나타냅니다.
ContentFilterDetectionResult	콘텐츠 필터링에 의해 수행된 검색 작업의 결과를 나타냅니다.
ContentFilterErrorResults	콘텐츠 필터링 오류 결과에 대한 정보입니다.
ContentFilterResult	필터링된 콘텐츠 심각도 수준 및 필터링되었는지 여부에 대한 정보입니다.
ContentFilterResultsFor Prompt	요청의 단일 프롬프트에 대한 콘텐츠 필터링 결과입니다.
ContentFilterSuccessResult DetailsForPrompt	콘텐츠 필터링 성공 결과에 대한 정보입니다.
ContentFilterSuccessResults ForChoice	생성된 모델 출력에 대해 평가된 콘텐츠 필터링에 대한 정보입니다.
ElasticsearchChatExtension Configuration	Azure OpenAI 채팅 확장으로 사용할 때 Elasticsearch에 대한 구성 가능한 옵션의 특정 표현입니다.
ElasticsearchIndexField MappingOptions	구성된 Elasticsearch® 리소스를 사용할 때 필드가 처리되는 방식을 제어하는 선택적 설정입니다.
EmbeddingItem	단일 포함 관련 비교의 표현입니다.
Embeddings	포함 요청의 응답 데이터 표현입니다. 포함은 텍스트 문자열의 관련성을 측정하며 검색, 클러스터링, 권장 사항 및 기타 유사한 시나리오에 일반적으로 사용됩니다.
EmbeddingsUsage	이 요청 및 응답에 사용되는 토큰의 양을 측정합니다.
EventStream	반복 가능하고 삭제 가능한 읽기 가능한 스트림입니다.

FunctionCall	모델에 의해 생성된 대로 호출되어야 하는 함수의 이름과 인수입니다.
FunctionDefinition	일치하는 사용자 입력에 대한 응답으로 채팅 완료가 호출될 수 있는 호출자 지정 함수의 정의입니다.
FunctionName	채팅 완료 작업을 처리할 때 사용할 특정 요청 제공 함수의 정확한 이름을 지정하는 구조체입니다.
GetAudioTranscriptionOptions	오디오 전사 요청에 대한 옵션
GetAudioTranslationOptions	오디오 번역 요청에 대한 옵션
GetChatCompletionsOptions	이 모듈에는 생성된 해당 모델과 나란히 살려는 모델이 포함되어 있습니다. 이는 생성된 모델과 이름/유형이 다른 고객 지향 모델을 제공하는 데 유용합니다.
GetCompletionsOptions	완료 요청에 대한 구성 정보입니다. 완료는 다양한 작업을 지원하고 제공된 프롬프트 데이터에서 계속되거나 "완료"되는 텍스트를 생성합니다.
GetEmbeddingsOptions	사용자 지정 포함 요청에 대한 옵션
GetImagesOptions	이미지를 생성하는 데 사용되는 요청 데이터를 나타냅니다.
ImageGenerationContentFilterResults	이미지 생성 요청에 대한 콘텐츠 필터링 결과를 설명합니다.
ImageGenerationData	base64로 인코딩된 데이터 또는 이미지를 검색할 수 있는 URL로 제공되는 생성된 단일 이미지의 표현입니다.
ImageGenerationPromptFilterResults	이미지 생성 요청의 프롬프트에 대한 콘텐츠 필터링 결과를 설명합니다.
ImageGenerations	성공적인 이미지 생성 작업의 결과입니다.
MaxTokensFinishDetails	모델이 자연스럽게 완료되기 전에 토큰 제한에 도달했음을 나타내는 중지 이유의 구조화된 표현입니다.
OnYourDataAccessTokenAuthenticationOptions	액세스 토큰을 사용할 때 Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataApiKeyAuthenticationOptions	API 키를 사용할 때 Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataAuthenticationOptions	Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataConnectionStringEncodingAuthenticationOptions	연결 문자열 사용할 때 Azure OpenAI On Your Data에 대한 인증 옵션입니다.

OnYourDataDeploymentNameVectorizationSource	벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보는 동일한 Azure OpenAI 리소스의 내부 포함 모델 배포 이름을 기반으로 합니다.
OnYourDataEncodedApiKeyAuthenticationOptions	Elasticsearch로 인코딩된 API 키를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataEndpointVectorizationSource	벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보는 포함에 대한 공용 Azure OpenAI 엔드 포인트 호출을 기반으로 합니다.
OnYourDataKeyAndKeyIdAuthenticationOptions	Elasticsearch 키 및 키 ID 쌍을 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataModelErrorIdVectorizationSource	검색 서비스 모델 ID를 기반으로 하는 벡터 검색을 적용할 때 Azure OpenAI On Your Data에서 사용하는 벡터화 원본의 세부 정보입니다. 현재 Elasticsearch®에서만 지원됩니다.
OnYourDataSystemAssignedManagedIdentityAuthenticationOptions	시스템 할당 관리 ID를 사용하는 경우 Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataUserAssignedManagedIdentityAuthenticationOptions	사용자가 할당한 관리 ID를 사용할 때 Azure OpenAI On Your Data에 대한 인증 옵션입니다.
OnYourDataVectorizationSource	벡터 검색을 사용하여 Azure OpenAI On Your Data에 대한 벡터화 원본의 추상 표현입니다.
OpenAIClientOptions	
PineconeChatExtensionConfiguration	Azure OpenAI 채팅 확장으로 사용할 때 Pinecone에 대한 구성 가능한 옵션의 특정 표현입니다.
PineconeFieldMappingOptions	구성된 Pinecone 리소스를 사용할 때 필드가 처리되는 방식을 제어하는 선택적 설정입니다.
StopFinishDetails	모델에 의한 자연스러운 종료를 나타내는 중지 이유의 구조화된 표현입니다.

형식 별칭

[+] 테이블 확장

AudioResult	요청된 응답 형식을 기반으로 하는 전사 결과 형식입니다.
AudioResultFormat	오디오 작업의 결과 형식

AudioTranscription Task	"transcribe", "translate"
AzureChatExtension ConfigurationUnion	AzureChatExtensionConfigurationUnion에 대한 별칭
AzureChatExtension Type	"azure_search", "azure_ml_index", "azure_cosmos_db", "elasticsearch", "pinecone"
AzureSearchQuery Type	"simple", "semantic", "vector", "vector_simple_hybrid", "vector_semantic_hybrid"
ChatCompletions NamedToolSelection Union	ChatCompletionsNamedToolSelectionUnion의 별칭
ChatCompletions ResponseFormat Union	ChatCompletionsResponseFormatUnion의 별칭
ChatCompletionsTool CallUnion	ChatCompletionsToolCallUnion에 대한 별칭
ChatCompletionsTool DefinitionUnion	ChatCompletionsToolDefinitionUnion에 대한 별칭
ChatCompletionsTool SelectionPreset	"auto", "none"
ChatFinishDetails Union	ChatFinishDetailsUnion의 별칭
ChatMessageContent ItemUnion	ChatMessageContentItemUnion의 별칭
ChatMessageImage DetailLevel	"auto", "low", "high"
ChatRequestMessage Union	ChatRequestMessageUnion의 별칭
ChatRole	"system", "도우미", "user", "function", "tool"
CompletionsFinish Reason	"stop", "length", "content_filter", "function_call", "tool_calls"
ContentFilterResult DetailsForPrompt	콘텐츠 필터링 범주(검색된 경우)에 대한 정보입니다.
ContentFilterResults ForChoice	콘텐츠 필터링 결과가 검색된 경우의 정보입니다.

ContentFilterSeverity	"safe", "low", "medium", "high"
ElasticsearchQueryType	"simple", "vector"
FunctionCallPreset	"auto", "none"
ImageGenerationQuality	"standard", "hd"
ImageGenerationResponseFormat	"url", "b64_json"
ImageGenerationStyle	"natural", "vivid"
ImageSize	"256x256", "512x512", "1024x1024", "1792x1024", "1024x1792"
OnYourDataAuthenticationOptionsUnion	OnYourDataAuthenticationOptionsUnion에 대한 별칭
OnYourDataAuthenticationType	"api_key", "connection_string", "key_and_key_id", "encoded_api_key", "access_token", "system_assigned_managed_identity", "user_assigned_managed_identity"
OnYourDataVectorizationSourceType	"endpoint", "deployment_name", "model_id"
OnYourDataVectorizationSourceUnion	OnYourDataVectorizationSourceUnion에 대한 별칭

⌚ GitHub에서 Microsoft와 공동 작업

이 콘텐츠의 원본은 GitHub에서 찾을 수 있으며, 여기서 문제와 끌어오기 요청을 만들고 검토할 수도 있습니다. 자세한 내용은 [참여자 가이드](#)를 참조하세요.



Azure SDK for JavaScript 피드백

Azure SDK for JavaScript은(는) 오픈 소스 프로젝트입니다. 다음 링크를 선택하여 피드백을 제공해 주세요.

⚙️ 설명서 문제 열기

☒ 제품 사용자 의견 제공

Azure.AI.OpenAI 네임스페이스

참조

① 중요

일부 정보는 릴리스되기 전에 상당 부분 수정될 수 있는 시험판 제품과 관련이 있습니다. Microsoft는 여기에 제공된 정보에 대해 어떠한 명시적이거나 묵시적인 보증도 하지 않습니다.

클래스

 테이블 확장

ChatChoice	채팅 완료 요청에 대한 단일 완료 결과의 표현입니다.
ChatCompletions	채팅 완료 요청에 대한 전체 응답의 표현입니다.
ChatCompletions Options	채팅 완료 요청에 사용되는 구성 정보입니다.
ChatMessage	채팅 완료 상호 작용 내에서 역할 특성이 지정된 단일 메시지입니다.
Choice	완료 응답 내의 선택 모델입니다.
Completions	완료 요청에 대한 예상 응답 스키마입니다.
CompletionsLog Probability	완료 선택 내의 LogProbs 모델입니다.
Completions Options	본문 스키마를 게시하여 배포에서 프롬프트 완료를 만듭니다.
CompletionsUsage	완료 요청에 대해 처리된 토큰 수의 표현입니다. 개수는 프롬프트, 선택, 선택 대체, best_of 세대 및 기타 소비자의 모든 토큰을 고려합니다.
EmbeddingItem	개체 목록 항목 요청을 포함할 것으로 예상되는 응답 스키마입니다.
Embeddings	요청을 포함할 것으로 예상되는 응답 스키마입니다.
EmbeddingsOptions	배포에서 프롬프트 완료를 만드는 스키마입니다.
EmbeddingsUsage	이 요청 및 응답에 사용된 토큰의 양을 측정합니다.
OpenAIClient	완료 및 검색을 위한 Azure OpenAI API.

OpenAI
Client
Options

OpenAI Client에 대한 클라이언트 옵션입니다.

StreamingChat
Choice

StreamingChat
Completions

Streaming
Choice

Streaming
Completions

구조체

 테이블 확장

ChatRole

채팅 완료 상호 작용 내에서 메시지의 의도된 목적에 대한 설명입니다.

열거형

 테이블 확장

OpenAI
Client
Options.
ServiceVersion

사용할 서비스의 버전입니다.

피드백

이 페이지가 도움이 되었나요?

 Yes

 No

제품 사용자 의견 제공 

Azure OpenAI 서비스 REST API 참조

아티클 • 2024. 03. 12.

이 문서에서는 Azure OpenAI에 대한 유추 REST API 엔드포인트에 대한 세부 정보를 제공합니다.

인증

Azure OpenAI는 두 가지 인증 방법을 제공합니다. API 키 또는 Microsoft Entra ID를 사용할 수 있습니다.

- API 키 인증:** 이 인증 형식의 경우 모든 API 요청은 `api-key` HTTP 헤더에 API 키를 포함해야 합니다. [빠른 시작](#)은 이러한 형식의 인증으로 전화를 거는 방법에 대한 지침을 제공합니다.
- Microsoft Entra ID 인증:** Microsoft Entra 토큰을 사용하여 API 호출을 인증할 수 있습니다. 인증 토큰은 요청에 `Authorization` 헤더로 포함됩니다. 제공된 토큰은 `Bearer` 가 앞에 와야 합니다(예: `Bearer YOUR_AUTH_TOKEN`). [Microsoft Entra ID로 인증](#)하는 방법 가이드를 읽을 수 있습니다.

REST API 버전 관리

서비스 API는 `api-version` 쿼리 매개 변수를 사용하여 버전이 지정됩니다. 모든 버전은 YYYY-MM-DD 날짜 구조를 따릅니다. 예시:

HTTP

```
POST https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/completions?api-version=2023-05-15
```

완료

완료 작업을 통해 모델은 제공된 프롬프트를 기반으로 하나 이상의 예측 완료를 생성합니다. 서비스는 또한 각 위치에서 대체 토큰의 확률을 반환할 수 있습니다.

완료 만들기

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/completions?api-version={{api-version}}
```

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	형식	필수 여부	설명
<code>your-resource-name</code>	string	Required	Azure OpenAI 리소스의 이름입니다.
<code>deployment-id</code>	string	Required	모델을 배포할 때 선택한 배포 이름입니다.
<code>api-version</code>	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- [2022-12-01 Swagger 사양](#)
- [2023-03-15-preview \(2024년 4월 2일 사용 중지\) Swagger 사양](#)
- [2023-05-15 Swagger 사양](#)
- [2023-06-01-preview Swagger 사양](#)
- [2023-07-01-preview \(2024년 4월 2일 사용 중지\) Swagger 사양](#)
- [2023-08-01-preview \(2024년 4월 2일 사용 중지\) Swagger 사양](#)
- [2023-09-01-preview \(2024년 4월 2일 사용 중지\) Swagger 사양](#)
- [2023-10-01-preview Swagger 사양](#)
- [2023-12-01-preview \(2024년 4월 2일 사용 중지\) Swagger 사양](#)

- 2024-02-15-preview Swagger 사양 ↴
- 2024-03-01-preview Swagger 사양 ↴
- 2024-02-01 Swagger 사양 ↴

요청 본문

테이블 확장

매개 변수	형식	필수 여부	기본값	설명
<code>prompt</code>	문자열 또는 배열	선택사항	<code><\\ endoftext\\ ></code>	문자열 또는 문자열 배열로 인코딩된 완료를 생성하라는 프롬프트 또는 프롬프트입니다. <code><\\ endoftext\\ ></code> 는 학습 중에 모델이 볼 수 있는 문서 구분 기호이므로 프롬프트를 지정하지 않으면 새 문서의 시작 부분에서처럼 모델이 생성됩니다.
<code>max_tokens</code>	정수	선택사항	16	완료 시 생성할 최대 토큰 수입니다. 프롬프트의 토큰 수와 <code>max_tokens</code> 는 모델의 컨텍스트 길이를 초과할 수 없습니다. 대부분의 모델에는 컨텍스트 길이가 2048인 토큰이 있습니다 (4096을 지원하는 최신 모델 제외).
<code>temperature</code>	number	선택사항	1	사용할 샘플링 온도(0에서 2 사이)입니다. 값이 높을수록 모델이 더 많은 위험을 감수합니다. 더 창의적인 애플리케이션의 경우 0.9를 시도하고 답변이 잘 정의된 애플리케이션의 경우 0(<code>argmax sampling</code>)을 시도합니다. 일반적으로 이를 변경하거나 <code>top_p</code> 를 변경하는 것이 좋지만 둘 다 변경하는 것은 권장하지 않습니다.
<code>top_p</code>	number	선택사항	1	모델이 <code>top_p</code> 확률 질량을 가진 토큰의 결과를 고려하는 핵 샘플링이라고 하는 온도를 사용한 샘플링의 대안입니다. 따라서 0.1은 상위 10% 확률 질량을 구성하는 토큰만 고려됨을 의미합니다. 일반적으로 이를 변경하거나 온도를 변경하는 것이 좋지만 둘 다 변경하는 것은 권장하지 않습니다.
<code>logit_bias</code>	map	선택사항	null	완료 시 지정된 토큰이 나타날 가능성을 수정합니다. 토큰(GPT 토크나이저에서 토큰 ID로 지정)을 -100에서 100 사이의 관련 바이어스 값에 매핑하는 json 개체를 허용합니다. 이 토크나이저 도구(GPT-2 및 GPT-3 모두에서 작동)를 사용하여 텍스트를 토큰 ID로 변환할 수 있습니다. 수학적으로, 바이어스는 샘플링 전에 모델에 의해 생성된 로짓에 추가됩니다. 정확한 효과는 모델마다 다르지만, -1과 1 사이의 값은 선택 가능성을 낮추거나 높여야 합니다. -100이나 100 같은 값은 관련 토큰을 금지하거나 독점적으로 선택해야 합니다. 예를 들어 {"50256": -100}을 전달하여 <code>< endoftext ></code> 토큰이 생성되지 않도록 할 수 있습니다.
<code>user</code>	string	선택사항		최종 사용자를 나타내는 고유 식별자로, 남용을 모니터링하고 감지하는 데 도움이 됩니다.
<code>n</code>	정수	선택사항	1	각 프롬프트에 대해 생성할 완료 수입니다. 참고: 이 매개 변수는 많은 완료를 생성하므로 토큰 할당량을 빠르게 소모할 수 있습니다. 신중하게 사용하고 <code>max_tokens</code> 및 중지에 대한 적절한 설정이 있는지 확인합니다.
<code>stream</code>	부울 값	선택사항	False	부분 진행률을 다시 스트리밍할지 여부를 나타냅니다. 설정된 경우 사용할 수 있게 되면 토큰은 데이터 전송 이벤트로 전송되고 스트림은 <code>data: [DONE]</code> 메시지로 종료됩니다.
<code>logprobs</code>	정수	선택사항	null	가장 가능성が高い 토큰과 선택한 토큰에 대한 로그 확률을 포함합니다. 예를 들어 <code>logprobs</code> 가 10이면 API는 가장 가능성이 높은 토큰 10개의 목록을 반환합니다. API는 항상 샘플링된 토큰의 <code>logprob</code> 을 반환하므로 응답에 <code>logprobs+1</code> 요소가 있을 수 있습니다. 이 매개 변수는 <code>gpt-35-turbo</code> 와 함께 사용할 수 없습니다.
<code>suffix</code>	string	선택사항	null	삽입된 텍스트가 완료된 뒤에 오는 접미사입니다.
<code>echo</code>	부울 값	선택사항	False	완료와 함께 프롬프트를 다시 에코합니다. 이 매개 변수는 <code>gpt-35-turbo</code> 와 함께 사용할 수 없습니다.

매개 변수	형식	필수 여부	기본값	설명
<code>stop</code>	문자열 또는 배열	선택사항	null	API가 추가 토큰 생성을 중지하는 최대 4개의 시퀀스입니다. 반환된 텍스트에는 중지 시퀀스가 포함되지 않습니다. 비전이 포함된 GPT-4 Turbo의 경우 시퀀스가 최대 2개 지원됩니다.
<code>presence_penalty</code>	number	선택사항	0	-2.0~2.0 사이의 숫자 양수 값은 지금까지 텍스트에 나타나는지 여부에 따라 새 토큰에 페널티를 부여하여 모델이 새 항목에 대해 이야기할 가능성을 높입니다.
<code>frequency_penalty</code>	number	선택사항	0	-2.0~2.0 사이의 숫자 양수 값은 지금까지 텍스트의 기준 빈도를 기반으로 새 토큰에 불이익을 주어 모델이 동일한 줄을 그대로 반복할 가능성을 줄입니다.
<code>best_of</code>	정수	선택사항	1	서버 쪽에서 best_of 완료를 생성하고 "최상"(토큰당 로그 확률이 가장 낮은 것)을 반환합니다. 결과를 스트리밍할 수 없습니다. n과 함께 사용하면 best_of는 후보 완료 횟수를 제어하고 n은 반환할 횟수를 지정합니다. best_of는 n보다 커야 합니다. 참고: 이 매개 변수는 많은 완료를 생성하므로 토큰 할당량을 빠르게 소모할 수 있습니다. 신중하게 사용하고 <code>max_tokens</code> 및 중지에 대한 적절한 설정이 있는지 확인합니다. 이 매개 변수는 <code>gpt-35-turbo</code> 와 함께 사용할 수 없습니다.

예제 요청

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/completions?api-version=2023-05-15 \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d "{
  \"prompt\": \"Once upon a time\",
  \"max_tokens\": 5
}"
```

예제 응답

JSON

```
{
  "id": "cmpl-4kGh7iXtjW4lc9eGhff6Hp8C7btDQ",
  "object": "text_completion",
  "created": 1646932609,
  "model": "ada",
  "choices": [
    {
      "text": ", a dark line crossed",
      "index": 0,
      "logprobs": null,
      "finish_reason": "length"
    }
  ]
}
```

예제 응답에서 `finish_reason`은 `stop`과 같습니다. `finish_reason`이 `content_filter`와 같으면 [콘텐츠 필터링 가이드](#)를 참조하여 이 문제가 발생하는 이유를 이해하세요.

포함

기계 학습 모델 및 기타 알고리즘에서 쉽게 사용할 수 있는 지정된 입력의 벡터 표현을 가져옵니다.

① 참고

OpenAI는 현재 `text-embedding-ada-002` 을 사용하여 더 많은 수의 배열 입력을 허용합니다. Azure OpenAI는 현재 `text-embedding-ada-002 (Version 2)` 에 대해 최대 16개의 입력 배열을 지원합니다. 둘 다 이 모델에 대해 8191 미만으로 유지하려면 API 요청당 최대 입력 토큰 제한이 필요합니다.

포함 만들기

HTTP

POST `https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/embeddings?api-version={{api-version}}`

경로 매개 변수

 테이블 확장

매개 변수	형식	필수 여부	설명
<code>your-resource-name</code>	string	Required	Azure OpenAI 리소스의 이름입니다.
<code>deployment-id</code>	string	Required	모델 배포의 이름입니다. 전화를 걸려면 먼저 모델을 배포해야 합니다.
<code>api-version</code>	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-03-15-preview (2024년 4월 2일 사용 중지) Swagger 사양 
- 2023-05-15 Swagger 사양 
- 2023-06-01-preview Swagger 사양 
- 2023-07-01-preview (2024년 4월 2일 사용 중지) Swagger 사양 
- 2023-08-01-preview (2024년 4월 2일 사용 중지) Swagger 사양 
- 2023-09-01-preview (2024년 4월 2일 사용 중지) Swagger 사양 
- 2023-10-01-preview Swagger 사양 
- 2023-12-01-preview (2024년 4월 2일 사용 중지) Swagger 사양 
- 2024-02-15-preview Swagger 사양 
- 2024-03-01-preview Swagger 사양 
- 2024-02-01 Swagger 사양 

요청 본문

 테이블 확장

매개 변수	형식	필수 여부	기본값	설명
<code>input</code>	문자열 또는 배열	예	해당 없음	배열 또는 문자열로 인코딩된 포함 항목을 가져올 텍스트를 입력합니다. 입력 토큰 수는 사용 중인 모델에 따라 달라집니다. <code>text-embedding-ada-002 (Version 2)</code> 만 배열 입력을 지원합니다.
<code>user</code>	string	아니요	Null	최종 사용자를 나타내는 고유 식별자입니다. 이렇게 하면 Azure OpenAI가 남용을 모니터링하고 검색하는 데 도움이 됩니다. PII 식별자를 전달하지 말고 GUID와 같은 의사 익명 값을 대신 사용
<code>encoding_format</code>	string	아니요	<code>float</code>	포함을 반환할 형식입니다. <code>float</code> 또는 <code>base64</code> 중 하나일 수 있습니다. 기본값은 <code>float</code> 입니다. [추가됨 2024-03-01-preview]
<code>dimensions</code>	정수	아니요		결과 출력 포함에 포함해야 하는 차원의 수입니다. 이후 모델에서 <code>text-embedding-3</code> 만 지원됩니다. [추가된 기능 2024-03-01-preview]

예제 요청

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/embeddings?api-version=2023-05-15 \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d "{\"input\": \"The food was delicious and the waiter...\"}"
```

예제 응답

JSON

```
{  
  "object": "list",  
  "data": [  
    {  
      "object": "embedding",  
      "embedding": [  
        0.018990106880664825,  
        -0.0073809814639389515,  
        .... (1024 floats total for ada)  
        0.021276434883475304,  
      ],  
      "index": 0  
    }  
  ],  
  "model": "text-similarity-babbage:001"  
}
```

채팅 완료

GPT-35-Turbo 및 GPT-4 모델을 사용하여 채팅 메시지 완성을 만듭니다.

채팅 완료 만들기

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/chat/completions?api-version={{api-version}}
```

경로 매개 변수

 테이블 확장

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
deployment-id	string	Required	모델 배포의 이름입니다. 전화를 걸려면 먼저 모델을 배포해야 합니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 또는 YYYY-MM-DD-미리 보기 형식을 따릅니다.

지원되는 버전

- 2023-03-15-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-05-15 [Swagger 사양](#)
- 2023-06-01-preview [Swagger 사양](#)
- 2023-07-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-08-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-09-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-10-01-preview [Swagger 사양](#)
- 2023-12-01-preview (2024년 4월 2일 사용 중지) (이 버전 이상은 Vision 시나리오에 필요) [Swagger 사양](#)
- 2024-02-15-preview [Swagger 사양](#)
- 2024-03-01-preview [Swagger 사양](#)
- 2024-02-01 [Swagger 사양](#)

요청 본문

요청 본문은 일련의 메시지로 구성됩니다. 모델은 이전 메시지를 컨텍스트로 사용하여 마지막 메시지에 대한 응답을 생성합니다.

 테이블 확장

매개 변수	형식	필수 여부	기본 값	설명
messages	array	예	해당 없음	이 채팅 완료 요청과 관련된 일련의 메시지입니다. 대화에 이전 메시지를 포함해야 합니다. 각 메시지에는 <code>role</code> 과 <code>content</code> 가 있습니다.
role	string	예	해당 없음	현재 메시지를 제공하는 사용자를 나타냅니다. <code>system</code> , <code>user</code> , <code>assistant</code> , <code>tool</code> , 또는 <code>function</code> 일 수 있습니다.
content	문자열 또는 배열	예	해당 없음	메시지의 콘텐츠입니다. 비전 지원 시나리오가 아니라면 문자열어야 합니다. 최신 API 버전과 함께 비전 모델이 포함된 GPT-4 Turbo를 사용하는 <code>user</code> 메시지의 일부라면 <code>content</code> 는 각 항목이 텍스트 또는 이미지를 나타내는 구조의 배열이어야 합니다. <ul style="list-style-type: none">• <code>text</code>: 입력 텍스트는 속성이 다음과 같은 구조체로 표시됩니다.<ul style="list-style-type: none">◦ <code>type = "text"</code>◦ <code>text = </code> 입력 텍스트• <code>images</code>: 입력 이미지는 속성이 다음과 같은 구조체로 표시됩니다.<ul style="list-style-type: none">◦ <code>type = "image_url"</code>◦ <code>image_url = </code> 속성이 다음과 같은 구조체<ul style="list-style-type: none">▪ <code>url = </code> 이미지 URL▪ (선택 사항) <code>detail = high</code>, 또는 <code>low auto</code>
contentPart	개체	아니요	해당 없음	사용자의 다중 모달 메시지 중 일부입니다. 텍스트 형식 또는 이미지 형식일 수 있습니다. 텍스트인 경우 텍스트 문자열이 됩니다. 이미지인 경우 <code>contentPartImage</code> 개체가 됩니다.
contentPartImage	개체	아니요	해당 없음	사용자가 업로드한 이미지를 나타냅니다. 여기에는 <code>url</code> 속성이 있고, 이는 이미지의 URL 또는 Base 64로 인코딩된 이미지 데이터의 URL에 해당합니다. 또한 <code>auto</code> , <code>low</code> 또는 <code>high</code> 일 수 있는 <code>detail</code> 속성이 있습니다.
enhancements	개체	아니요	해당 없음	채팅에 요청된 비전 향상 기능을 나타냅니다. 이 속성에는 <code>grounding</code> , <code>ocr</code> 각각 부울 <code>enabled</code> 속성이 있습니다. OCR 서비스 및/또는 개체 검색/접지 서비스 [이 미리 보기 매개 변수는 GA API에서 2024-02-01 사용할 수 없음]을 요청하는 데 사용합니다.
dataSources	개체	아니요	해당 없음	추가 리소스 데이터를 나타냅니다. 비전 기능을 향상시키려면 Computer Vision 리소스 데이터가 필요합니다. 속성과 <code>type</code> 속성 <code>"AzureComputerVision"</code> <code>parameters</code> 이 있어야 하는 속성이 <code>endpoint</code> 과 <code>key</code> 있습니다. 이러한 문자열은 Computer Vision 리소스의 엔드포인트 URL 및 액세스 키로 설정해야 합니다.

예제 요청

텍스트 전용 채팅

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/chat/completions?api-version=2023-05-15 \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d '{"messages":[{"role": "system", "content": "You are a helpful assistant."}, {"role": "user", "content": "Does Azure OpenAI support customer managed keys?"}, {"role": "assistant", "content": "Yes, customer managed keys are supported by Azure OpenAI."}, {"role": "user", "content": "Do other Azure AI services support this too?"}]}'
```

비전을 사용하는 채팅

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/chat/completions?api-version=2023-12-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d '{
  "messages": [
    {
      "role": "system",
      "content": "You are a helpful assistant."
    },
    {
      "role": "user",
      "content": [
        {
          "type": "text",
          "text": "Describe this picture:"
        },
        {
          "type": "image_url",
          "image_url": {
            "url": "https://learn.microsoft.com/azure/ai-services/computer-vision/media/quickstarts/presentation.png"
          }
        }
      ]
    }
  ]
}'
```

비전으로 향상된 채팅

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/extensions/chat/completions?api-version=2023-12-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d '{
  "enhancements": {
    "ocr": {
      "enabled": true
    },
    "grounding": {
      "enabled": true
    }
  },
  "dataSources": [
    {
      "type": "AzureComputerVision",
      "parameters": {
        "endpoint": "<Computer Vision Resource Endpoint>",
        "key": "<Computer Vision Resource Key>"
      }
    }
  ],
  "messages": [
    {
      "role": "system",
      "content": "You are a helpful assistant."
    },
    {
      "role": "user",
      "content": [
        {
          "type": "text",
          "text": "Describe this picture:"
        },
        {
          "type": "image_url",
          "image_url": "https://learn.microsoft.com/azure/ai-services/computer-vision/media/quickstarts/presentation.png"
        }
      ]
    }
  ]
}'
```

예제 응답

콘솔

```
{
  "id": "chatcmpl-6v7mkQj980V1yBec6ETrKPRqFjNw9",
  "object": "chat.completion",
  "created": 1679072649,
  "model": "gpt-35-turbo",
  "usage": {
    "prompt_tokens": 58,
    "completion_tokens": 68,
    "total_tokens": 126
  },
  "choices": [
    {
      "message": {
        "role": "assistant",
        "content": "Yes, other Azure AI services also support customer managed keys. Azure AI services offer multiple options for customers to manage keys, such as using Azure Key Vault, customer-managed keys in Azure Key Vault or customer-managed keys through Azure Storage service. This helps customers ensure that their data is secure and access to their services is controlled."
      },
      "finish_reason": "stop",
      "index": 0
    }
  ]
}
```

읽기 용이성을 위해 조정된 출력 서식입니다. 실제 출력은 줄 바꿈이 없는 단일 텍스트 블록입니다.

예제 응답에서 `finish_reason`은 `stop`과 같습니다. `finish_reason`이 `content_filter`와 같으면 [콘텐츠 필터링 가이드](#)를 참조하여 이 문제가 발생하는 이유를 이해하세요.

ⓘ 중요

`functions` 및 `function_call` 매개 변수는 API의 [2023-12-01-preview](#) 버전이 릴리스되어 사용되지 않습니다. `functions`을 바꾸는 것은 `tools` 매개 변수입니다. `function_call`을 바꾸는 것은 `tool_choice` 매개 변수입니다. [2023-12-01-preview](#)의 일부로 도입된 별별 함수 호출은 `gpt-35-turbo(1106)` 및 GPT-4 Turbo Preview로도 알려진 `gpt-4(1106-preview)`에서만 지원됩니다.

매개 변수	형식	필수 여부	기본값	설명
messages	array	Required		이 채팅 완료 요청과 관련된 컨텍스트 메시지 모음입니다. 일반적인 사용법은 어시스턴트의 동작에 대한 치침을 제공하는 시스템 역할에 대한 채팅 메시지 로 시작하고 사용자와 도우미 역할 간의 메시지가 교대로 이어집니다.
temperature	number	선택 사항	1	사용할 샘플링 온도(0에서 2 사이)입니다. 0.8과 같이 값이 높을수록 출력이 더욱 무작위로 생성되고, 0.2와 같이 값이 낮을수록 출력이 더욱 집중되고 결정적이게 됩니다. 일반적으로 이를 변경하거나 <code>top_p</code> 을(를) 변경하는 것이 좋지만 둘 다 변경하지는 않는 것이 좋습니다.
n	정수	선택 사항	1	각 입력 메시지에 대해 생성할 채팅 완료 선택 항목 수입니다.
stream	부울 값	선택 사항	false	설정되면 ChatGPT의 경우처럼 부분 메시지 델타가 전송됩니다. 토큰은 데이터 전용 서버 전송 이벤트로 전송되며 스트림은 <code>data: [DONE]</code> 메시지로 종료됩니다."
stop	문자열 또는 배열	선택 사항	null	API가 추가 토큰 생성을 중지하는 최대 4개의 시퀀스입니다.
max_tokens	정수	선택 사항	inf	생성된 답변에 허용되는 최대 토큰 수입니다. 기본적으로 모델이 반환할 수 있는 토큰 수는 (4096 - 프롬프트 토큰)입니다.
presence_penalty	number	선택 사항	0	-2.0~2.0 사이의 숫자 양수 값은 지금까지 텍스트에 나타나는지 여부에 따라 새 토큰에 페널티를 부여하여 모델이 새 항목에 대해 이야기할 가능성을 높입니다.
frequency_penalty	number	선택 사항	0	-2.0~2.0 사이의 숫자 양수 값은 지금까지 텍스트의 기준 빈도를 기반으로 새 토큰에 불이익을 주어 모델이 동일한 줄을 그대로 반복할 가능성을 줄입니다.
logit_bias	개체	선택 사항	null	완료 시 지정된 토큰이 나타날 가능성을 수정합니다. 토큰(토크나이저에서 토큰 ID로 지정)을 -100에서 100 사이의 관련 바이어스 값에 매핑하는 json 개체를 허용합니다. 수학적으로, 바이어스는 샘플링 전에 모델에 의해 생성된 로짓에 추가됩니다. 정확한 효과는 모델마다 다르지만, -1과 1 사이의 값은 선택 가능성을 낮추거나 높여야 합니다. -100이나 100 같은 값은 관련 토큰을 금지하거나 독점적으로 선택해야 합니다.
user	string	선택 사항		Azure OpenAI가 남용을 모니터링하고 감지하는 데 도움이 될 수 있는 최종 사용자를 나타내는 고유 식별자입니다.
function_call		선택 사항		[Deprecated in 2023-12-01-preview replacement parameter is <code>tools_choice</code>] 모델이 함수 호출에 응답하는 방식을 제어합니다. "없음"은 모델이 함수를 호출하지 않고 최종 사용자에게 응답함을 의미합니다. <code>auto</code> 는 모델이 최종 사용자 또는 함수 호출 중에서 선택할 수 있습니다. <code>{"name": "my_function"}</code> 을 통해 특정 함수를 지정하면 모델이 해당 함수를 호출하게 됩니다. 기능이 없을 경우 "none"이 기본값입니다. <code>auto</code> 는 함수가 있는 경우 기본값입니다. 이 매개 변수에는 API 버전 2023-07-01-preview 이 필요합니다.
functions	<code>FunctionDefinition[]</code>	선택 사항		[Deprecated in 2023-12-01-preview replacement parameter is <code>tools</code>] 모델이 JSON 입력을 생성할 수 있는 함수 목록입니다. 이 매개 변수에는 API 버전 2023-07-01-preview 이 필요합니다.
tools	string(도구의 형식입니다. <code>function</code> 만 지원됩니다.)	선택 사항		모델이 호출할 수 있는 도구 목록입니다. 현재 함수만 도구로 지원됩니다. 이를 사용하여 모델이 JSON 입력을 생성할 수 있는 함수 목록을 제공합니다. 이 매개 변수에는 API 버전 2023-12-01-preview 이 필요합니다.
tool_choice	문자열 또는 개체	선택 사항	함수가 없는 경우 <code>none</code> 이다. <code>auto</code> 는 함수가 있는 경우 기본값입니다.	모델에서 호출되는 함수(해당하는 경우)를 제어합니다. 없음은 모델이 함수를 호출하지 않고 대신 메시지를 생성한다는 것을 의미합니다. <code>auto</code> 는 모델이 메시지를 생성하거나 함수를 호출하는 중에서 선택할 수 있습니다. <code>{"type: "function", "function": {"name": "my_function"}}</code> 을 통해 특정 함수를 지정하면 모델이 해당 함수를 호출하게 됩니다. 이 매개 변수에는 API 버전 2023-12-01-preview 이상이 필요합니다.

채팅메시지

채팅 완료 상호 작용 내의 단일 역할 속성 메시지입니다.

[\[+\] 테이블 확장](#)

이름	형식	설명
콘텐츠	string	이 메시지 페이로드와 연결된 텍스트입니다.
function_call	FunctionCall	모델에 의해 생성된 대로 호출되어야 하는 함수의 이름과 인수입니다.
name	string	이 메시지 작성자의 name입니다. 역할이 function인 경우 name이 필요하며 응답이 content에 있는 함수의 이름이어야 합니다. a~z, A~Z, 0~9, 밑줄을 포함할 수 있으며 최대 길이는 64자입니다.
역할(role)	채팅 역할	이 메시지 페이로드와 관련된 역할

채팅역할

채팅 완료 상호작용 내 메시지의 의도된 목적에 대한 설명입니다.

[\[+\] 테이블 확장](#)

이름	형식	설명
도우미	string	시스템 지시, 사용자 프롬프트 입력에 대한 응답을 제공하는 역할입니다.
function	string	채팅 완료에 대한 기능 결과를 제공하는 역할입니다.
시스템	string	도우미의 행동을 지시하거나 설정하는 역할.
user	string	채팅 완료를 위한 입력을 제공하는 역할입니다.

함수

2023-12-01-preview API 버전에 추가된 tools 매개 변수와 함께 사용됩니다.

[\[+\] 테이블 확장](#)

이름	형식	설명
description	string	함수를 호출하는 시기과 방법을 선택하기 위해 모델에서 사용하는 함수의 기능에 대한 설명입니다.
name	string	호출할 함수의 이름입니다. a~z, A~Z, 0~9 또는 밑줄과 대시를 포함해야 하며 최대 길이는 64자여야 합니다.
매개 변수	개체	함수가 허용하는 매개 변수로, JSON 스키마 개체로 설명됩니다. 형식에 대한 설명서는 JSON 스키마 참조 를 참조하세요."

FunctionCall-Degraded

모델에 의해 생성된 대로 호출되어야 하는 함수의 이름과 인수입니다. 이를 위해서는 API 버전 2023-07-01-preview 이 필요합니다.

[\[+\] 테이블 확장](#)

이름	형식	설명
arguments	string	JSON 형식의 모델에 의해 생성된 함수 호출에 사용할 인수입니다. 모델이 항상 유효한 JSON을 생성하지는 않고 함수 스키마에서 정의되지 않은 매개 변수를 조작할 수 있습니다. 함수를 호출하기 전에 코드에서 인수의 유효성을 검사하세요.
name	string	호출할 함수의 이름입니다.

FunctionDefinition-Degraded

일치하는 사용자 입력에 대한 응답으로 채팅 완료가 호출될 수 있는 호출자 지정 함수의 정의입니다. 이를 위해서는 API 버전 2023-07-01-preview 이 필요합니다.

[\[+\] 테이블 확장](#)

이름	형식	설명
description	string	함수가 수행하는 작업에 대한 설명입니다. 모델이 기능을 선택하고 해당 매개 변수를 해석할 때 이 설명을 사용합니다.
name	string	호출할 함수의 이름입니다.
매개 변수		함수가 허용하는 매개 변수로, JSON 스키마 개체로 설명됩니다.

완료 확장

채팅 완료를 위한 확장(예: Azure OpenAI On Your Data).

ⓘ 중요

다음 정보는 API 버전 2023-12-01-preview에 대한 것입니다. 현재 **버전의 API가 아닙니다**. 최신 참조 설명서를 찾으려면 [Azure OpenAI On Your Data 참조를 참조하세요](#).

채팅 완료 확장 기능 사용

HTTP

```
POST {your-resource-name}/openai/deployments/{deployment-id}/extensions/chat/completions?api-version={api-version}
```

경로 매개 변수

 테이블 확장

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
deployment-id	string	Required	모델 배포의 이름입니다. 전화를 걸려면 먼저 모델을 배포해야 합니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-06-01-preview [Swagger 사양](#)
- 2023-07-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-08-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-09-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-10-01-preview [Swagger 사양](#)
- 2023-12-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)

예제 요청

Azure AI Search, Azure Cosmos DB for MongoDB vCore, Pinecone 및 Elasticsearch를 사용하여 요청을 수행할 수 있습니다. 자세한 내용은 [Azure OpenAI On Your Data](#)를 참조하세요.

Azure AI 검색

Console

```
curl -i -X POST YOUR_RESOURCE_NAME/openai/deployments/YOUR_DEPLOYMENT_NAME/extensions/chat/completions?api-version=2023-06-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d \
{
  "temperature": 0,
  "max_tokens": 1000,
  "top_p": 1.0,
  "dataSources": [
```

```

    {
      "type": "AzureCognitiveSearch",
      "parameters": {
        "endpoint": "YOUR_AZURE_COGNITIVE_SEARCH_ENDPOINT",
        "key": "YOUR_AZURE_COGNITIVE_SEARCH_KEY",
        "indexName": "YOUR_AZURE_COGNITIVE_SEARCH_INDEX_NAME"
      }
    }
  ],
  "messages": [
    {
      "role": "user",
      "content": "What are the differences between Azure Machine Learning and Azure AI services?"
    }
  ]
}

```

Azure Cosmos DB for MongoDB vCore

JSON

```

curl -i -X POST YOUR_RESOURCE_NAME/openai/deployments/YOUR_DEPLOYMENT_NAME/extensions/chat/completions?api-
version=2023-06-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d \
'

{
  "temperature": 0,
  "top_p": 1.0,
  "max_tokens": 800,
  "stream": false,
  "messages": [
    {
      "role": "user",
      "content": "What is the company insurance plan?"
    }
  ],
  "dataSources": [
    {
      "type": "AzureCosmosDB",
      "parameters": {
        "authentication": {
          "type": "ConnectionString",
          "connectionString": "mongodb+srv://onyourdatatest:{password}#{@{cluster-
name}.mongocluster.cosmos.azure.com/?tls=true&authMechanism=SCRAM-SHA-256&retrywrites=false&maxIdleTimeMS=120000}"
        },
        "databaseName": "vectordb",
        "containerName": "azuredocs",
        "indexName": "azuredocindex",
        "embeddingDependency": {
          "type": "DeploymentName",
          "deploymentName": "{embedding deployment name}"
        },
        "fieldsMapping": {
          "vectorFields": [
            "contentvector"
          ]
        }
      }
    }
  ]
}

```

Elasticsearch

콘솔

```

curl -i -X POST YOUR_RESOURCE_NAME/openai/deployments/YOUR_DEPLOYMENT_NAME/extensions/chat/completions?api-
version=2023-12-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \

```

```
-d \
{
  "messages": [
    {
      "role": "system",
      "content": "you are a helpful assistant that talks like a pirate"
    },
    {
      "role": "user",
      "content": "can you tell me how to care for a parrot?"
    }
  ],
  "dataSources": [
    {
      "type": "Elasticsearch",
      "parameters": {
        "endpoint": "{search endpoint}",
        "indexName": "{index name}",
        "authentication": {
          "type": "KeyAndKeyId",
          "key": "{key}",
          "keyId": "{key id}"
        }
      }
    }
  ]
}
```

Azure Machine Learning

콘솔

```
curl -i -X POST YOUR_RESOURCE_NAME/openai/deployments/YOUR_DEPLOYMENT_NAME/extensions/chat/completions?api-
version=2023-12-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d \
'
{
  "messages": [
    {
      "role": "system",
      "content": "you are a helpful assistant that talks like a pirate"
    },
    {
      "role": "user",
      "content": "can you tell me how to care for a parrot?"
    }
  ],
  "dataSources": [
    {
      "type": "AzureMLIndex",
      "parameters": {
        "projectResourceId": "/subscriptions/{subscription-id}/resourceGroups/{resource-group-
name}/providers/Microsoft.MachineLearningServices/workspaces/{workspace-id}",
        "name": "my-project",
        "version": "5"
      }
    }
  ]
}
```

Pinecone

콘솔

```
curl -i -X POST YOUR_RESOURCE_NAME/openai/deployments/YOUR_DEPLOYMENT_NAME/extensions/chat/completions?api-
version=2023-12-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d \
'
{

```

```

"messages": [
  {
    "role": "system",
    "content": "you are a helpful assistant that talks like a pirate"
  },
  {
    "role": "user",
    "content": "can you tell me how to care for a parrot?"
  }
],
"dataSources": [
  {
    "type": "Pinecone",
    "parameters": {
      "authentication": {
        "type": "APIKey",
        "apiKey": "{api key}"
      },
      "environment": "{environment name}",
      "indexName": "{index name}",
      "embeddingDependency": {
        "type": "DeploymentName",
        "deploymentName": "{embedding deployment name}"
      },
      "fieldsMapping": {
        "titleField": "title",
        "urlField": "url",
        "filepathField": "filepath",
        "contentFields": [
          "content"
        ],
        "contentFieldsSeparator": "\n"
      }
    }
  }
]
}

```

예제 응답

JSON

```
{
  "id": "12345678-1a2b-3c4e5f-a123-12345678abcd",
  "model": "",
  "created": 1684304924,
  "object": "chat.completion",
  "choices": [
    {
      "index": 0,
      "messages": [
        {
          "role": "tool",
          "content": "{\"citations\": [{\"content\": \"\\nAzure AI services are cloud-based artificial intelligence (AI) services...\", \"id\": null, \"title\": \"What is Azure AI services\", \"filepath\": null, \"url\": null, \"metadata\": {\"chunking\": \"original document size=250. Scores=0.4314117431640625 and 1.72564697265625.0ng Highlight count=4.\", \"chunk_id\": \"0\"}, \"intent\": \"[\\\"Learn about Azure AI services.\\\"]\"}], \"end_turn\": false
        },
        {
          "role": "assistant",
          "content": " \nAzure AI services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. [doc1]. Azure Machine Learning is a cloud service for accelerating and managing the machine learning project lifecycle. [doc1].",
          "end_turn": true
        }
      ]
    }
}
```

매개 변수	Type	필수 여부	기본 값	설명
messages	array	Required	null	채팅 형식으로 채팅 완료를 생성할 메시지입니다.
dataSources	array	Required		Azure OpenAI On Your Data 기능에 사용할 데이터 원본입니다.
temperature	number	선택 사항	0	사용할 샘플링 온도(0에서 2 사이)입니다. 0.8과 같이 값이 높을수록 출력이 더욱 무작위로 생성되고, 0.2와 같이 값이 낮을수록 출력이 더욱 집중되고 결정적이게 됩니다. 일반적으로 이를 변경하거나 top_p(를) 변경하는 것이 좋지만 둘 다 변경하지는 않는 것이 좋습니다.
top_p	number	선택 사항	1	핵 샘플링이라고 하는 온도 샘플링의 대안으로, 모델은 확률 질량이 top_p 인 토큰의 결과를 고려합니다. 따라서 0.1은 상위 10% 확률 질량을 구성하는 토큰만 고려됨을 의미합니다. 일반적으로 이를 변경하거나 온도를 변경하는 것이 좋지만 둘 다 변경하는 것은 권장하지 않습니다.
stream	부울 값	선택 사항	false	설정되면 ChatGPT에서처럼 부분 메시지 델타가 전송됩니다. 토큰은 사용할 수 있게 되면 데이터 전용 서버 전송 이벤트로 전송되며 스트림은 "messages": [{"delta": {"content": "[DONE]"}, "index": 2, "end_turn": true}] 메시지에 의해 종료됩니다.
stop	문자열 또는 배열	선택 사항	null	API가 추가 토큰 생성을 중지하는 최대 2개의 시퀀스입니다.
max_tokens	정수	선택 사항	1000	생성된 답변에 허용되는 최대 토큰 수입니다. 기본적으로 모델이 반환할 수 있는 토큰 수는 4096 - prompt_tokens 입니다.

다음 매개 변수는 `dataSources` 내부의 `parameters` 필드 내에서 사용할 수 있습니다.

데이터 확장

매개 변수	Type	필수 여부	기본 값	설명
<code>type</code>	string	Required	null	Azure OpenAI On Your Data 기능에 사용할 데이터 원본입니다. Azure AI Search의 경우 값이 <code>AzureCognitiveSearch</code> 입니다. Azure Cosmos DB for MongoDB vCore의 경우 값이 <code>AzureCosmosDB</code> 입니다. Elasticsearch의 경우 값이 <code>Elasticsearch</code> 입니다. Azure Machine Learning의 경우 값이 <code>AzureMLIndex</code> 입니다. Pinecone의 경우 값이 <code>Pinecone</code> 입니다.
<code>indexName</code>	string	Required	null	사용할 검색 인덱스입니다.
<code>inScope</code>	부울 값	선택 사항	true	설정된 경우 이 값은 접두 데이터 컨텐츠와 관련된 응답을 제한합니다.
<code>topNDocuments</code>	number	선택 사항	5	응답을 생성하는 데 사용되는 데이터 인덱스에서 최고점을 매기는 문서의 수를 지정합니다. 짧은 문서가 있거나 더 많은 컨텍스트를 제공하려는 경우 값을 높여야 할 수 있습니다. Azure OpenAI Studio에서 검색된 문서 매개 변수입니다.
<code>semanticConfiguration</code>	string	선택 사항	null	의미 체계 검색 구성. <code>queryType</code> 이(가) <code>semantic</code> 또는 <code>vectorSemanticHybrid</code> (으)로 설정된 경우에만 필요합니다.
<code>roleInformation</code>	string	선택 사항	null	응답을 생성할 때 모델이 어떻게 작동해야 하는지와 참조해야 하는 컨텍스트에 대한 지침을 모델에 제공합니다. Azure OpenAI Studio의 "시스템 메시지"에 해당합니다. 자세한 내용은 데이터 사용 을 참조하세요. 전체 토큰 한도에 포함되는 100개의 토큰 한도가 있습니다.
<code>filter</code>	string	선택 사항	null	중요한 문서에 대한 액세스를 제한 하는 데 사용되는 필터 패턴
<code>embeddingEndpoint</code>	string	선택 사항	null	일반적으로 <code>https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/embeddings?api-version=2023-05-15</code> 형식인 Ada 포함 모델 배포에 대한 엔드포인트 URL입니다. 프라이빗 네트워크 및 프라이빗 엔드포인트 외부의 벡터 검색 용 <code>embeddingKey</code> 매개 변수와 함께 사용합니다.
<code>embeddingKey</code>	string	선택 사항	null	Ada 포함 모델 배포를 위한 API 키입니다. 프라이빗 네트워크 및 프라이빗 엔드포인트 외부의 벡터 검색 용 <code>embeddingEndpoint</code> 과 함께 사용합니다.
<code>embeddingDeploymentName</code>	string	선택 사항	null	동일한 Azure OpenAI 리소스 내 Ada 포함 모델의 배포 이름입니다. 벡터 검색 용 <code>embeddingEndpoint</code> 및 <code>embeddingKey</code> 대신 사용합니다. <code>embeddingEndpoint</code> 및 <code>embeddingKey</code> 매개 변수가 모두 정의된 경우에만 사용해야 합니다. 이 매개 변수가 제공되면 Azure OpenAI On Your Data는 내부 호출을 사용하여 Azure OpenAI 엔드포인트를 호출하는 대신 Ada 포함 모델을 평가합니다. 이렇게 하면 프라이빗 네트워크와 프라이빗 엔드포인트에서 벡터 검색을 사용할 수 있습니다. 이 매개 변수가 정의되었는지 여부에 상관없이 청구는 동일하게 유지됩니다. API 버전 <code>2023-06-01-preview</code> 이상부터 포함 모델을 사용할 수 있는 지역에서 사용할 수 있습니다.

매개 변수	Type	필수 여부	기본값	설명
strictness	number	선택 사항	3	쿼리와 관련된 문서를 분류하는 임계값을 설정합니다. 값을 높이면 관련성 임계값이 커지고 응답에 대한 관련성이 낮은 문서가 더 많이 필터링됩니다. 이 값을 너무 높게 설정하면 사용 가능한 문서가 제한되기 때문에 모델이 응답을 생성하지 못할 수도 있습니다.

Azure AI Search 매개 변수

다음 매개 변수가 Azure AI Search에 사용됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
endpoint	string	Required	null	Azure AI Search 전용입니다. 데이터 원본 엔드포인트입니다.
key	string	Required	null	Azure AI Search 전용입니다. 서비스의 Azure Cognitive Search 관리 키 중 하나입니다.
queryType	string	선택 사항	simple	Azure AI Search에 사용되는 쿼리 옵션을 나타냅니다. 사용 가능한 형식: <code>simple</code> , <code>semantic</code> , <code>vector</code> , <code>vectorSimpleHybrid</code> , <code>vectorSemanticHybrid</code> .
fieldsMapping	사전	Azure AI Search의 선택 사항입니다.	null	데이터 원본을 추가할 때 매핑할 필드를 정의합니다.

다음 매개 변수는 `authentication` 필드 내에서 사용되며, 이를 통해 [공용 네트워크에 액세스하지 않고도 Azure OpenAI를 사용할 수 있습니다](#).

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
<code>type</code>	string	Required	null	인증 유형입니다.
<code>managedIdentityResourceId</code>	string	Required	null	인증 시 이용할 사용자가 할당한 관리 ID의 리소스 ID입니다.

JSON

```
"authentication": {
    "type": "UserAssignedManagedIdentity",
    "managedIdentityResourceId": "/subscriptions/{subscription-id}/resourceGroups/{resource-group}/providers/Microsoft.ManagedIdentity/userAssignedIdentities/{resource-name}"
},
```

다음 매개 변수는 `fieldsMapping` 필드 내에서 사용됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
<code>titleField</code>	string	선택 사항	null	각 문서의 원래 제목을 포함하는 인덱스의 필드입니다.
<code>urlField</code>	string	선택 사항	null	각 문서의 원래 URL을 포함하는 인덱스의 필드입니다.
<code>filepathField</code>	string	선택 사항	null	각 문서의 원래 파일 이름을 포함하는 인덱스의 필드입니다.
<code>contentFields</code>	사전	선택 사항	null	각 문서의 기본 텍스트 콘텐츠를 포함하는 인덱스의 필드입니다.
<code>contentFieldsSeparator</code>	string	선택 사항	null	콘텐츠 필드의 구분 기호입니다. 기본적으로 <code>\n</code> 을 사용합니다.

JSON

```
"fieldsMapping": {
    "titleField": "myTitleField",
    "urlField": "myUrlField",
    "filepathField": "myFilePathField",
```

```

"contentFields": [
    "myContentField"
],
"contentFieldsSeparator": "\n"
}

```

다음 매개 변수는 선택적 `embeddingDependency` 매개 변수 내부에서 사용되며, 여기에는 동일한 Azure OpenAI 리소스에서 내부 포함 모델의 배포 이름을 기반으로 하는 벡터화 원본의 세부 정보가 포함됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본 값	설명
<code>deploymentName</code>	string	선택 사항	null	사용할 벡터화 원본의 형식입니다.
<code>type</code>	string	선택 사항	null	동일한 Azure OpenAI 리소스 내에 있는 포함 모델의 배포 이름입니다. 이렇게 하면 Azure OpenAI API 키 없이 Azure OpenAI 공용 네트워크에 액세스하지 않고도 벡터 검색을 사용할 수 있습니다.

JSON

```

"embeddingDependency": {
    "type": "DeploymentName",
    "deploymentName": "{embedding deployment name}"
},

```

Azure Cosmos DB for MongoDB vCore 매개 변수

다음 매개 변수는 Azure Cosmos DB for MongoDB vCore에서 사용됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본 값	설명
<code>type</code> (<code>authentication</code> 내부에 있음)	string	Required	null	Azure Cosmos DB for MongoDB vCore 전용입니다. 사용할 인증입니다. Azure Cosmos Mongo vCore, 값은 <code>ConnectionString</code> 입니다.
<code>connectionString</code>	string	Required	null	Azure Cosmos DB for MongoDB vCore 전용입니다. Azure Cosmos Mongo vCore 계정을 인증하는 데 사용할 연결 문자열입니다.
<code>databaseName</code>	string	Required	null	Azure Cosmos DB for MongoDB vCore 전용입니다. Azure Cosmos Mongo vCore 데이터베이스 이름입니다.
<code>containerName</code>	string	Required	null	Azure Cosmos DB for MongoDB vCore 전용입니다. 데이터베이스의 Azure Cosmos Mongo vCore 컨테이너 이름입니다.
<code>type</code> (<code>embeddingDependencyType</code> 내부에 있음)	string	Required	null	포함하는 모델 종속성을 나타냅니다.
<code>deploymentName</code> (<code>embeddingDependencyType</code> 내부에 있음)	string	Required	null	포함하는 모델 배포 이름입니다.
<code>fieldsMapping</code>	사전	Azure Cosmos DB for MongoDB vCore에서 필 요합니다.	null	인덱스 데이터 열 매핑. Azure Cosmos DB for MongoDB vCore를 사용하는 경우 벡터를 저장하는 필드를 나타내는 <code>vectorFields</code> 값이 필요합니다.

다음 매개 변수는 선택적 `embeddingDependency` 매개 변수 내부에서 사용되며, 여기에는 동일한 Azure OpenAI 리소스에서 내부 포함 모델의 배포 이름을 기반으로 하는 벡터화 원본의 세부 정보가 포함됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본 값	설명
deploymentName	string	선택 사항	null	사용할 벡터화 원본의 형식입니다.
type	string	선택 사항	null	동일한 Azure OpenAI 리소스 내에 있는 포함 모델의 배포 이름입니다. 이렇게 하면 Azure OpenAI API 키 없이 Azure OpenAI 공용 네트워크에 액세스하지 않고도 벡터 검색을 사용할 수 있습니다.

JSON

```
"embeddingDependency": {
  "type": "DeploymentName",
  "deploymentName": "{embedding deployment name}"
},
```

Elasticsearch 매개 변수

Elasticsearch에는 다음 매개 변수가 사용됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
endpoint	string	Required	null	Elasticsearch에 연결하기 위한 엔드포인트입니다.
indexName	string	Required	null	Elasticsearch 인덱스의 이름입니다.
type(authentication) 내부에 있음)	string	Required	null	사용할 인증입니다. Elasticsearch의 경우 값이 <code>keyAndKeyId</code> 입니다.
key(authentication) 내부에 있음)	string	Required	null	Elasticsearch에 연결하는 데 사용되는 키입니다.
keyId(authentication) 내부에 있음)	string	Required	null	사용할 키 ID입니다. ElasticSearch용입니다.

다음 매개 변수는 `fieldsMapping` 필드 내에서 사용됩니다.

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
titleField	string	선택 사항	null	각 문서의 원래 제목을 포함하는 인덱스의 필드입니다.
urlField	string	선택 사항	null	각 문서의 원래 URL을 포함하는 인덱스의 필드입니다.
filepathField	string	선택 사항	null	각 문서의 원래 파일 이름을 포함하는 인덱스의 필드입니다.
contentFields	사전	선택 사항	null	각 문서의 기본 텍스트 콘텐츠를 포함하는 인덱스의 필드입니다.
contentFieldsSeparator	string	선택 사항	null	콘텐츠 필드의 구분 기호입니다. 기본적으로 <code>\n</code> 을 사용합니다.
vectorFields	사전	선택 사항	null	벡터 데이터를 나타내는 필드의 이름입니다.

JSON

```
"fieldsMapping": {
  "titleField": "myTitleField",
  "urlField": "myUrlField",
  "filepathField": "myFilePathField",
  "contentFields": [
    "myContentField"
  ],
  "contentFieldsSeparator": "\n",
  "vectorFields": [
    "myVectorField"
  ]
}
```

다음 매개 변수는 선택적 `embeddingDependency` 매개 변수 내부에서 사용되며, 여기에는 동일한 Azure OpenAI 리소스에서 내부 포함 모델의 배포 이름을 기반으로 하는 벡터화 원본의 세부 정보가 포함됩니다.

매개 변수	Type	필수 여부	기본 값	설명
deploymentName	string	선택 사항	null	사용할 벡터화 원본의 형식입니다.
type	string	선택 사항	null	동일한 Azure OpenAI 리소스 내에 있는 포함 모델의 배포 이름입니다. 이렇게 하면 Azure OpenAI API 키 없이 Azure OpenAI 공용 네트워크에 액세스하지 않고도 벡터 검색을 사용할 수 있습니다.

JSON

```
"embeddingDependency": {
    "type": "DeploymentName",
    "deploymentName": "{embedding deployment name}"
},
```

Azure Machine Learning 매개 변수

다음 매개 변수는 Azure Machine Learning에서 사용됩니다.

매개 변수	Type	필수 여부	기본값	설명
projectResourceId	string	Required	null	프로젝트 리소스 ID입니다.
name	string	Required	null	Azure Machine Learning 프로젝트의 이름입니다.
version(authentication 내부에 있음)	string	Required	null	Azure Machine Learning 벡터 인덱스의 버전입니다.

다음 매개 변수는 선택적 `embeddingDependency` 매개 변수 내부에서 사용되며, 여기에는 동일한 Azure OpenAI 리소스에서 내부 포함 모델의 배포 이름을 기반으로 하는 벡터화 원본의 세부 정보가 포함됩니다.

매개 변수	Type	필수 여부	기본 값	설명
deploymentName	string	선택 사항	null	사용할 벡터화 원본의 형식입니다.
type	string	선택 사항	null	동일한 Azure OpenAI 리소스 내에 있는 포함 모델의 배포 이름입니다. 이렇게 하면 Azure OpenAI API 키 없이 Azure OpenAI 공용 네트워크에 액세스하지 않고도 벡터 검색을 사용할 수 있습니다.

JSON

```
"embeddingDependency": {
    "type": "DeploymentName",
    "deploymentName": "{embedding deployment name}"
},
```

Pinecone 매개 변수

다음 매개 변수는 Pinecone에 사용됩니다.

매개 변수	Type	필수 여부	기본 값	설명
type(authentication 내부에 있음)	string	Required	null	사용할 인증입니다. Pinecone의 경우 값이 <code>APIKey</code> 입니다.
apiKey(authentication 내부에 있음)	string	Required	null	Pinecone의 API 키입니다.
environment	string	Required	null	Pinecone 환경의 이름입니다.

매개 변수	Type	필수 여부	기본 값	설명
indexName	string	Required	null	Pinecone 인덱스의 이름입니다.
embeddingDependency	string	Required	null	벡터 검색에 포함되는 종속성입니다.
type (embeddingDependency 내부에 있음)	string	Required	null	종속성의 형식입니다. Pinecone의 경우 값이 DeploymentName입니다.
deploymentName (embeddingDependency 내부에 있음)	string	Required	null	배포의 이름입니다.
titleField (fieldsMapping 내부에 있음)	string	Required	null	제목으로 사용할 인덱스 필드의 이름입니다.
urlField (fieldsMapping 내부에 있음)	string	Required	null	URL로 사용할 인덱스 필드의 이름입니다.
filepathField (fieldsMapping 내부에 있음)	string	Required	null	파일 경로로 사용할 인덱스 필드의 이름입니다.
contentFields (fieldsMapping 내부에 있음)	string	Required	null	콘텐츠로 처리해야 하는 인덱스 필드의 이름입니다.
vectorFields	사전 선택 사항	선택 사항	null	벡터 데이터를 나타내는 필드의 이름입니다.
contentFieldsSeparator (fieldsMapping 내부에 있음)	string	Required	null	콘텐츠 필드의 구분 기호입니다. 기본적으로 \n을 사용합니다.

다음 매개 변수는 선택적 embeddingDependency 매개 변수 내부에서 사용되며, 여기에는 동일한 Azure OpenAI 리소스에서 내부 포함 모델의 배포 이름을 기반으로 하는 벡터화 원본의 세부 정보가 포함됩니다.

테이블 확장

매개 변수	Type	필수 여부	기본 값	설명
deploymentName	string	선택 사항	null	사용할 벡터화 원본의 형식입니다.
type	string	선택 사항	null	동일한 Azure OpenAI 리소스 내에 있는 포함 모델의 배포 이름입니다. 이렇게 하면 Azure OpenAI API 키 없이 Azure OpenAI 공용 네트워크에 액세스하지 않고도 벡터 검색을 사용할 수 있습니다.

JSON

```
"embeddingDependency": {
    "type": "DeploymentName",
    "deploymentName": "{embedding deployment name}"
},
```

수집 작업 시작(미리 보기)

팁

선택한 JOB_NAME이 인덱스 이름으로 사용됩니다. 인덱스 이름의 [제약 조건](#)에 유의하세요.

콘솔

```
curl -i -X PUT https://YOUR_RESOURCE_NAME.openai.azure.com/openai/extensions/on-your-data/ingestion-jobs/JOB_NAME?
api-version=2023-10-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-H "searchServiceEndpoint: https://YOUR_AZURE_COGNITIVE_SEARCH_NAME.search.windows.net" \
-H "searchServiceAdminKey: YOUR_SEARCH_SERVICE_ADMIN_KEY" \
-H "storageConnectionString: YOUR_STORAGE_CONNECTION_STRING" \
-H "storageContainer: YOUR_INPUT_CONTAINER" \
-d '{ "dataRefreshIntervalInMinutes": 10 }'
```

예제 응답

JSON

```
{
  "id": "test-1",
  "dataRefreshIntervalInMinutes": 10,
  "completionAction": "cleanUpAssets",
  "status": "running",
  "warnings": [],
  "progress": {
    "stageProgress": [
      {
        "name": "Preprocessing",
        "totalItems": 100,
        "processedItems": 100
      },
      {
        "name": "Indexing",
        "totalItems": 350,
        "processedItems": 40
      }
    ]
  }
}
```

헤더 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
<code>searchServiceEndpoint</code>	string	Required	null	데이터를 수집할 검색 리소스의 엔드포인트입니다.
<code>searchServiceAdminKey</code>	string	선택 사항	null	제공된 경우 키가 <code>searchServiceEndpoint</code> 를 인증하는 데 사용됩니다. 제공되지 않은 경우 Azure OpenAI 리소스의 시스템 할당 ID가 사용됩니다. 이 경우 시스템 할당 ID에는 검색 리소스에 대한 "Search Service 기여자" 역할 할당이 있어야 합니다.
<code>storageConnectionString</code>	string	Required	null	입력 데이터가 있는 스토리지 계정의 연결 문자열입니다. 연결 문자열에 계정 키를 제공해야 합니다. 이는 <code>DefaultEndpointsProtocol=https;AccountName=<your storage account>;AccountKey=<your account key></code> 와 같이 표시됩니다.
<code>storageContainer</code>	string	Required	null	입력 데이터가 있는 컨테이너의 이름입니다.
<code>embeddingEndpoint</code>	string	선택 사항	null	의미 체계 또는 키워드 검색만 사용하는 경우에는 필요하지 않습니다. 벡터, 하이브리드 또는 하이브리드 + 의미 체계 검색을 사용하는 경우에 필요합니다.
<code>embeddingKey</code>	string	선택 사항	null	포함 엔드포인트의 키입니다. 포함 엔드포인트가 비어 있지 않은 경우 필요합니다.
<code>url</code>	string	선택 사항	null	URL이 null이 아닌 경우 제공된 URL이 제공된 스토리지 컨테이너로 크롤링된 후 그에 따라 수집됩니다.

본문 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	Type	필수 여부	기본값	설명
<code>dataRefreshIntervalInMinutes</code>	string	Required	0	데이터 새로 고침 간격(단위: 분)입니다. 예약하지 않고 단일 수집 작업을 실행하려면 이 매개 변수를 <code>0</code> 으로 설정합니다.
<code>completionAction</code>	string	선택 사항	<code>cleanUpAssets</code>	작업 완료 시 수집 프로세스가 진행되는 동안 만들어진 자산에 대해 수행해야 하는 작업입니다. 유효한 값은 <code>cleanUpAssets</code> 또는 <code>keepAllAssets</code> 입니다. <code>keepAllAssets</code> 는 중간 결과를 검토하는 데 관심 있는 사용자를 위해 중간 자산을 모두 남겨 두는데, 이는 자산 디버깅 시 유용할 수 있습니다. <code>cleanUpAssets</code> 는 작업 완료 후 자산을 제거합니다.
<code>chunkSize</code>	int	선택 사항	1024	이 숫자는 수집 흐름에서 생성되는 각 청크의 최대 토큰 수를 정의합니다.

수집 작업 나열(미리 보기)

콘솔

```
curl -i -X GET https://YOUR_RESOURCE_NAME.openai.azure.com/openai/extensions/on-your-data/ingestion-jobs?api-version=2023-10-01-preview \
-H "api-key: YOUR_API_KEY"
```

예제 응답

JSON

```
{
  "value": [
    {
      "id": "test-1",
      "dataRefreshIntervalInMinutes": 10,
      "completionAction": "cleanUpAssets",
      "status": "succeeded",
      "warnings": []
    },
    {
      "id": "test-2",
      "dataRefreshIntervalInMinutes": 10,
      "completionAction": "cleanUpAssets",
      "status": "failed",
      "error": {
        "code": "BadRequest",
        "message": "Could not execute skill because the Web Api request failed."
      },
      "warnings": []
    }
  ]
}
```

수집 작업의 상태 가져오기(미리 보기)

콘솔

```
curl -i -X GET https://YOUR_RESOURCE_NAME.openai.azure.com/openai/extensions/on-your-data/ingestion-jobs/YOUR_JOB_NAME?api-version=2023-10-01-preview \
-H "api-key: YOUR_API_KEY"
```

예제 응답 본문

JSON

```
{
  "id": "test-1",
  "dataRefreshIntervalInMinutes": 10,
  "completionAction": "cleanUpAssets",
  "status": "succeeded",
  "warnings": []
}
```

이미지 생성

생성된 이미지 요청(DALL-E 3)

텍스트 캡션에서 이미지 배치를 생성 및 검색합니다.

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/images/generations?api-version={{api-version}}
```

경로 매개 변수

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
deployment-id	string	Required	DALL-E 3 모델 배포의 이름(예: <i>MyDalle3</i>)입니다. 호출하기 전에 먼저 DALL-E 3 모델을 배포해야 합니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-12-01-preview (retiring April 2, 2024) [Swagger 사양 ↗](#)
 - 2024-02-15-preview [Swagger 사양 ↗](#)
 - 2024-02-01 [Swagger 사양 ↗](#)

요청 본문

매개 변수	형식	필수 여부	기본값	설명
<code>prompt</code>	string	Required		원하는 이미지에 대한 텍스트 설명입니다. 최대 길이는 4,000자입니다.
<code>n</code>	정수	선택 사항	1	생성할 이미지 수입니다. <code>n=1</code> 은 DALL-E 3에 한해 지원됩니다.
<code>size</code>	string	선택 사항	<code>1024x1024</code>	생성된 이미지의 크기입니다. <code>1792x1024</code> , <code>1024x1024</code> 또는 <code>1024x1792</code> 중 하나여야 합니다.
<code>quality</code>	string	선택 사항	<code>standard</code>	생성된 이미지의 품질입니다. <code>hd</code> 또는 <code>standard</code> 이어야 합니다.
<code>response_format</code>	string	선택 사항	<code>url</code>	생성된 이미지가 반환되는 형식은 (이미지를 가리키는 URL) 또는 <code>b64_json</code> (JSON 형식의 기본 64 바이트 코드)이어야 <code>url</code> 합니다.
<code>style</code>	string	선택 사항	<code>vivid</code>	생성된 이미지의 스타일입니다. <code>natural</code> 또는 <code>vivid</code> (초현실적/극적 이미지)여야 합니다.
<code>user</code>	string	선택 사항		남용을 모니터링하고 감지하는 데 도움이 될 수 있는 최종 사용자를 나타내는 고유 식별자입니다.

예제 요청

콘솔

```
curl -X POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/images/generations?api-version=2023-12-01-preview \
    -H "Content-Type: application/json" \
    -H "api-key: YOUR_API_KEY" \
    -d '{
        "prompt": "An avocado chair",
        "size": "1024x1024",
        "n": 3,
        "quality": "hd",
        "style": "vivid"
    }'
```

예제 응답

작업은 작업의 ID와 상태를 포함하는 202 상태 코드와 `GenerateImagesResponse` JSON 개체를 반환합니다.

JSON

```
{  
  "created": 1698116662,  
  "data": [  
    {
```

```
        "url": "url to the image",
        "revised_prompt": "the actual prompt that was used"
    },
    {
        "url": "url to the image"
    },
    ...
]
```

생성된 이미지 요청(DALL-E 2 미리 보기)

텍스트 캡션에서 이미지 배치를 생성합니다.

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/images/generations:submit?api-version={{api-version}}
```

경로 매개 변수

 테이블 확장

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-06-01-preview Swagger 사양 ↗

요청 본문

 테이블 확장

매개 변수	형식	필수 여부	기본값	설명
prompt	string	Required		원하는 이미지에 대한 텍스트 설명입니다. 최대 길이는 1000자입니다.
n	정수	선택 사항	1	생성할 이미지 수입니다. 1~5 사이여야 합니다.
size	string	선택 사항	1024 x 1024	생성된 이미지의 크기입니다. 256x256, 512x512 또는 1024x1024 중 하나여야 합니다.

예제 요청

콘솔

```
curl -X POST https://YOUR_RESOURCE_NAME.openai.azure.com/openai/images/generations:submit?api-version=2023-06-01-preview \
-H "Content-Type: application/json" \
-H "api-key: YOUR_API_KEY" \
-d '{
  "prompt": "An avocado chair",
  "size": "512x512",
  "n": 3
}'
```

예제 응답

작업은 작업의 ID와 상태를 포함하는 202 상태 코드와 GenerateImagesResponse JSON 개체를 반환합니다.

JSON

```
{
    "id": "f508bcf2-e651-4b4b-85a7-58ad77981ffa",
```

```
        "status": "notRunning"
    }
```

생성된 이미지 결과 가져오기(DALL-E 2 미리 보기)

이 API를 사용하여 이미지 생성 작업의 결과를 검색합니다. 이미지 생성은 현재 `api-version=2023-06-01-preview`에서만 가능합니다.

HTTP

```
GET https://{{your-resource-name}}.openai.azure.com/openai/operations/images/{operation-id}?api-version={api-version}
```

경로 매개 변수

 테이블 확장

매개 변수	형식	필수 여부	설명
<code>your-resource-name</code>	string	Required	Azure OpenAI 리소스의 이름입니다.
<code>operation-id</code>	string	Required	원본 이미지 생성 요청을 식별하는 GUID입니다.

지원되는 버전

- 2023-06-01-preview Swagger 사양 ↗

예제 요청

콘솔

```
curl -X GET "https://{{your-resource-name}}.openai.azure.com/openai/operations/images/{operation-id}?api-version=2023-06-01-preview"
-H "Content-Type: application/json"
-H "Api-Key: {api key}"
```

예제 응답

성공하면 작업이 `200` 상태 코드와 `OperationResponse` JSON 개체를 반환합니다. `status` 필드는 `"notRunning"`(작업이 큐에 있지만 아직 시작되지 않음), `"running"`, `"succeeded"`, `"canceled"`(작업 시간이 초과됨), `"failed"` 또는 `"deleted"` 일 수 있습니다. `succeeded` 상태는 생성된 이미지를 해당 URL에서 다운로드할 수 있음을 나타냅니다. 여러 이미지가 생성된 경우 해당 URL은 모두 `result.data` 필드에 반환됩니다.

JSON

```
{
  "created": 1685064331,
  "expires": 1685150737,
  "id": "4b755937-3173-4b49-bf3f-da6702a3971a",
  "result": {
    "data": [
      {
        "url": "<URL_TO_IMAGE>"
      },
      {
        "url": "<URL_TO_NEXT_IMAGE>"
      },
      ...
    ],
    "status": "succeeded"
  }
}
```

서버에서 생성된 이미지 삭제(DALL-E 2 미리 보기)

요청에서 반환된 작업 ID를 사용하여 Azure 서버에서 해당 이미지를 삭제할 수 있습니다. 생성된 이미지는 기본적으로 24시간 후에 자동으로 삭제되지만 원하는 경우 더 일찍 삭제를 실행할 수 있습니다.

HTTP

```
DELETE https://{{your-resource-name}}.openai.azure.com/openai/operations/images/{operation-id}?api-version={{api-version}}
```

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
operation-id	string	Required	원본 이미지 생성 요청을 식별하는 GUID입니다.

지원되는 버전

- 2023-06-01-preview Swagger 사양

예제 요청

콘솔

```
curl -X DELETE "https://{{your-resource-name}}.openai.azure.com/openai/operations/images/{operation-id}?api-version=2023-06-01-preview"
-H "Content-Type: application/json"
-H "Api-Key: {{api key}}"
```

응답

작업이 성공하면 204 상태 코드를 반환합니다. 이 API는 작업이 종료 상태(`running` 아님)에 있는 경우에만 성공합니다.

음성 텍스트 변환

음성 텍스트 변환 전사 요청

오디오 파일을 기록합니다.

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/audio/transcriptions?api-version={{api-version}}
```

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
deployment-id	string	Required	MyWhisperDeployment와 같은 Whisper 모델 배포의 이름입니다. 전화를 걸려면 먼저 Whisper 모델을 배포해야 합니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. 이 값은 YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-09-01-preview (2024년 4월 2일 사용 중지) Swagger 사양

- 2023-10-01-preview Swagger 사양 ↴
- 2023-12-01-preview (2024년 4월 2일 사용 중지) Swagger 사양 ↴
- 2024-02-15-preview Swagger 사양 ↴
- 2024-03-01-preview Swagger 사양 ↴
- 2024-02-01 Swagger 사양 ↴

요청 본문

 테이블 확장

매개 변수	형식	필수 여부	기본 값	설명
file	파일	예	해당 없음	<p><code>flac, mp3, mp4, mpeg, mpga, m4a, ogg, wav, webm</code> 형식 중 하나로 기록할 오디오 파일 개체(파일 이름 아님)입니다.</p> <p>Azure OpenAI Whisper 모델의 파일 크기 제한은 25MB입니다. 25MB보다 큰 파일을 전사해야 하는 경우 청크로 분할합니다. 또는 Azure AI Speech 일괄 처리 전사 API를 사용할 수 있습니다.</p> <p>GitHub의 Azure AI Speech SDK 리포지토리에서 샘플 오디오 파일을 가져올 수 있습니다.</p>
language	string	아니요	Null	<p><code>en</code> 와 같은 입력 오디오의 언어입니다. ISO-639-1 형식으로 입력 언어를 제공하면 정확도와 대기 시간이 향상됩니다.</p> <p>지원되는 언어 목록은 OpenAI 설명서를 참조하세요.</p>
prompt	string	아니요	Null	<p>모델 스타일을 안내하거나 이전 오디오 세그먼트를 계속 진행하기 위한 선택적 텍스트입니다. 프롬프트는 오디오 언어와 일치해야 합니다.</p> <p>사용 사례 예제를 비롯한 프롬프트에 대한 자세한 내용은 OpenAI 설명서를 참조하세요.</p>
response_format	string	아니요	json	<p>json, text, srt, verbose_json 또는 vtt 옵션 중 하나에 있는 전사 출력의 형식입니다.</p> <p>기본값은 <code>json</code>입니다.</p>
temperature	number	아니요	0	<p>샘플링 온도(0에서 1 사이)입니다.</p> <p>0.8과 같이 값이 높을수록 출력이 더 무작위로 생성되고, 0.2와 같이 값이 낮을수록 출력이 더욱 집중되고 결정적이게 됩니다. 0으로 설정하면 모델은 로그 확률을 사용하여 특정 임계값에 도달할 때까지 온도를 자동으로 높입니다.</p> <p>기본값은 0입니다.</p>

예제 요청

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/audio/transcriptions?api-version=2023-09-01-preview \
-H "Content-Type: multipart/form-data" \
-H "api-key: $YOUR_API_KEY" \
-F file="@./YOUR_AUDIO_FILE_NAME.wav" \
-F "language=en" \
-F "prompt=The transcript contains zoology terms and geographical locations." \
-F "temperature=0" \
-F "response_format=srt"
```

예제 응답

srt

```
1
00:00:00,960 --> 00:00:07,680
The ocelot, Lepardus pardalis, is a small wild cat native to the southwestern United States,
2
00:00:07,680 --> 00:00:13,520
Mexico, and Central and South America. This medium-sized cat is characterized by
```

```
3  
00:00:13,520 --> 00:00:18,960  
solid black spots and streaks on its coat, round ears, and white neck and undersides.  
  
4  
00:00:19,760 --> 00:00:27,840  
It weighs between 8 and 15.5 kilograms, 18 and 34 pounds, and reaches 40 to 50 centimeters  
  
5  
00:00:27,840 --> 00:00:34,560  
16 to 20 inches at the shoulders. It was first described by Carl Linnaeus in 1758.  
  
6  
00:00:35,360 --> 00:00:42,880  
Two subspecies are recognized, L. p. paradalis and L. p. mitis. Typically active during twilight  
  
7  
00:00:42,880 --> 00:00:48,480  
and at night, the ocelot tends to be solitary and territorial. It is efficient at climbing,  
  
8  
00:00:48,480 --> 00:00:54,480  
leaping, and swimming. It preys on small terrestrial mammals such as armadillo, opossum,  
  
9  
00:00:54,480 --> 00:00:56,480  
and lagomorphs.
```

음성 텍스트 변환 요청

다른 언어의 오디오 파일을 영어로 변환합니다. 지원되는 언어 목록은 [OpenAI 설명서](#)를 참조하세요.

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/audio/translations?api-version={{api-version}}
```

경로 매개 변수

[\[+\] 테이블 확장](#)

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.
deployment-id	string	Required	MyWhisperDeployment와 같은 Whisper 모델 배포의 이름입니다. 전화를 걸려면 먼저 Whisper 모델을 배포해야 합니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. 이 값은 YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2023-09-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2023-10-01-preview [Swagger 사양](#)
- 2023-12-01-preview (2024년 4월 2일 사용 중지) [Swagger 사양](#)
- 2024-02-15-preview [Swagger 사양](#)
- 2024-03-01-preview [Swagger 사양](#)
- 2024-02-01 [Swagger 사양](#)

요청 본문

[\[+\] 테이블 확장](#)

매개 변수	형식	필수 여부	기본 값	설명
file	파일	예	해당 없음	<p>flac, mp3, mp4, mpeg, mpga, m4a, ogg, wav 또는 webm 형식 중 하나로 기록할 오디오 파일(파일 이름 아님)입니다.</p> <p>Azure OpenAI Whisper 모델의 파일 크기 제한은 25MB입니다. 25MB보다 큰 파일을 전사해야 하는 경우 청크로 분할합니다.</p> <p>GitHub의 Azure AI Speech SDK 리포지토리에서 샘플 오디오 파일을 다운로드할 수 있습니다.</p>
prompt	string	아니요	Null	<p>모델 스타일을 안내하거나 이전 오디오 세그먼트를 계속 진행하기 위한 선택적 텍스트입니다. 프롬프트는 오디오 언어와 일치해야 합니다.</p> <p>사용 사례 예제를 비롯한 프롬프트에 대한 자세한 내용은 OpenAI 설명서를 참조하세요.</p>
response_format	string	아니요	json	<p>json, text, srt, verbose_json 또는 vtt 옵션 중 하나에 있는 전사 출력의 형식입니다.</p> <p>기본값은 json입니다.</p>
temperature	number	아니요	0	<p>샘플링 온도(0에서 1 사이)입니다.</p> <p>0.8과 같이 값이 높을수록 출력이 더 무작위로 생성되고, 0.2와 같이 값이 낮을수록 출력이 더욱 집중되고 결정적이게 됩니다. 0으로 설정하면 모델은 로그 확률을 사용하여 특정 임계값에 도달할 때까지 온도를 자동으로 높입니다.</p> <p>기본값은 0입니다.</p>

예제 요청

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/audio/translations?api-version=2023-09-01-preview \
-H "Content-Type: multipart/form-data" \
-H "api-key: $YOUR_API_KEY" \
-F file="@./YOUR_AUDIO_FILE_NAME.wav" \
-F "temperature=0" \
-F "response_format=json"
```

예제 응답

JSON

```
{
  "text": "Hello, my name is Wolfgang and I come from Germany. Where are you heading today?"}
```

텍스트 음성 변환

텍스트 음성 변환 합성

HTTP

```
POST https://{{your-resource-name}}.openai.azure.com/openai/deployments/{{deployment-id}}/audio/speech?api-version={{api-version}}
```

경로 매개 변수

 테이블 확장

매개 변수	형식	필수 여부	설명
your-resource-name	string	Required	Azure OpenAI 리소스의 이름입니다.

매개 변수	형식	필수 여부	설명
deployment-id	string	Required	텍스트 음성 변환 모델 배포의 이름(예: <code>MyTextToSpeechDeployment</code>)입니다. 호출하려면 먼저 텍스트 음성 변환 모델(예: <code>tts-1</code> 또는 <code>tts-1-hd</code>)을 배포해야 합니다.
api-version	string	Required	이 작업에 사용할 API 버전입니다. 이 값은 YYYY-MM-DD 형식을 따릅니다.

지원되는 버전

- 2024-02-15-preview Swagger 사양 ↗

요청 본문

☞ 테이블 확장

매개 변수	형식	필수 여부	기본값	설명
model	string	예	해당 없음	사용 가능한 TTS 모델(<code>tts-1</code> 또는 <code>tts-1-hd</code>) 중 하나입니다.
input	string	예	해당 없음	오디오를 생성할 텍스트입니다. 최대 길이는 4,096자입니다. 선택한 언어로 입력 텍스트를 지정합니다. ¹
voice	string	예	해당 없음	오디오를 생성할 때 사용할 음성입니다. 지원되는 음성은 <code>alloy</code> , <code>echo</code> , <code>fable</code> , <code>onyx nova</code> , <code>shimmer</code> 입니다. 음성 미리 보기는 OpenAI 텍스트 음성 변환 가이드 에서 확인할 수 있습니다.

¹ 텍스트 음성 변환 모델은 일반적으로 위스퍼 모델과 동일한 언어를 지원합니다. 지원되는 언어 목록은 [OpenAI 설명서](#)를 참조하세요.

예제 요청

콘솔

```
curl https://YOUR_RESOURCE_NAME.openai.azure.com/openai/deployments/YOUR_DEPLOYMENT_NAME/audio/speech?api-version=2024-02-15-preview \
-H "api-key: $YOUR_API_KEY" \
-H "Content-Type: application/json" \
-d '{
    "model": "tts-hd",
    "input": "I'm excited to try text to speech.",
    "voice": "alloy"
}' --output speech.mp3
```

예제 응답

음성은 이전 요청의 오디오 파일로 반환됩니다.

관리 API

Azure OpenAI는 Azure AI 서비스의 일부로 배포됩니다. 모든 Azure AI 서비스는 만들기, 업데이트 및 삭제 작업을 위해 동일한 관리 API 집합을 사용합니다. 관리 API는 OpenAI 리소스 내에서 모델을 배포하는 데에도 사용됩니다.

[관리 API 참조 설명서](#)

다음 단계

모델 및 REST API를 사용한 미세 조정에 대해 알아보세요. [Azure OpenAI를 지원하는 기본 모델](#)에 대해 자세히 알아봅니다.

Fine Tuning

참조

Service: Azure AI Services

API Version: 2023-10-01-preview

Operations

[+] 테이블 확장

Cancel	지정된 fine-tune-id로 지정된 미세 조정 작업의 처리를 취소합니다.
Create	지정된 학습 파일에서 지정된 모델을 미세 조정하는 작업을 만듭니다. 응답에는 작업 상태 및 하이퍼 매개 변수를 포함하여 큐에 추가된 작업의 세부 정보가 포함됩니다.
Delete	지정된 fine-tune-id에 지정된 미세 조정 작업을 삭제합니다.
Get	지정된 fine-tune-id로 지정된 단일 미세 조정 작업에 대한 세부 정보를 가져옵니다. 세부 정보에는 기본 모델, 학습 및 유효성 검사 파일, 하이퍼 매개 변수가 포함됩니다....
Get Events	지정된 fine-tune-id에 지정된 미세 조정 작업에 대한 이벤트를 가져옵니다. 이벤트는 작업 상태 변경되는 경우(예: 실행 중 또는 완료) 및 res...
List	Azure OpenAI 리소스가 소유한 모든 미세 조정 작업 목록을 가져옵니다. 각 미세 조정 작업에 대해 반환되는 세부 정보에는 기본 식별자 외에...

Deployments - Create Or Update

참조

Service: Azure AI Services

API Version: 2023-05-01

Cognitive Services 계정과 연결된 배포의 상태를 업데이트합니다.

HTTP

PUT

<https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.CognitiveServices/accounts/{accountName}/deployments/{deploymentName}?api-version=2023-05-01>

URI 매개 변수

 테이블 확장

Name	In(다음 안에)	필수	형식	Description
accountName	path	True	string	Cognitive Services 계정의 이름입니다. Regex pattern: ^[a-zA-Z0-9][a-zA-Z0-9_.-]*\$
deploymentName	path	True	string	Cognitive Services 계정과 연결된 배포의 이름
resourceGroupName	path	True	string	리소스 그룹의 이름. 이름은 대소문자를 구분하지 않습니다.
subscriptionId	path	True	string	대상 구독의 ID입니다.
api-version	query	True	string	이 작업에 사용할 API 버전입니다.

요청 본문

 테이블 확장

Name	형식	Description
properties	DeploymentProperties	Cognitive Services 계정 배포의 속성입니다.
sku	Sku	SKU를 나타내는 리소스 모델 정의

응답

 테이블 확장

Name	형식	Description
200 OK	Deployment	배포를 성공적으로 만들거나 업데이트합니다.
201 Created	Deployment	배포를 성공적으로 만듭니다.
Other Status Codes	ErrorResponse	작업이 실패한 이유를 설명하는 오류 응답입니다.

예제

PutDeployment

Sample Request

HTTP

HTTP

```
PUT  
https://management.azure.com/subscriptions/subscriptionId/resourceGroups/resourceGroupName/providers/Microsoft.CognitiveServices/accounts/accountName/deployments/deploymentName?api-version=2023-05-01
```

```
{  
  "sku": {  
    "name": "Standard",  
    "capacity": 1  
  },  
  "properties": {  
    "model": {  
      "format": "OpenAI",  
      "name": "ada",  
      "version": "1"  
    }  
  }  
}
```

Sample Response

Status code: 200

JSON

```
{  
  "id":  
    "/subscriptions/subscriptionId/resourceGroups/resourceGroupName/providers/Microsoft.CognitiveServices/accounts/accountName/  
deployments/deploymentName",  
  "name": "deploymentName",  
  "type": "Microsoft.CognitiveServices/accounts/deployments",  
  "sku": {  
    "name": "Standard",  
    "capacity": 1  
  },  
  "properties": {  
    "model": {  
      "format": "OpenAI",  
      "name": "ada",  
      "version": "1"  
    },  
    "provisioningState": "Succeeded"  
  }  
}
```

Status code: 201

JSON

```
{  
  "id":  
    "/subscriptions/subscriptionId/resourceGroups/resourceGroupName/providers/Microsoft.CognitiveServices/accounts/accountName/  
deployments/deploymentName",  
  "name": "deploymentName",  
  "type": "Microsoft.CognitiveServices/accounts/deployments",  
  "sku": {  
    "name": "Standard",  
    "capacity": 1  
  },  
  "properties": {  
    "model": {  
      "format": "OpenAI",  
      "name": "ada",  
      "version": "1"  
    },  
    "provisioningState": "Accepted"  
  }  
}
```

정의

데이터 확장

Name	Description
------	-------------

CallRateLimit	호출 속도 제한 Cognitive Services 계정입니다.
createdByType	리소스를 만든 ID의 형식입니다.
Deployment	Cognitive Services 계정 배포.
DeploymentModel	Cognitive Services 계정 배포 모델의 속성입니다.
DeploymentModelVersionUpgradeOption	배포 모델 버전 업그레이드 옵션.
DeploymentProperties	Cognitive Services 계정 배포의 속성입니다.
DeploymentProvisioningState	작업이 호출되었을 때 리소스의 상태 가져옵니다.
DeploymentScaleSettings	Cognitive Services 계정 배포 모델의 속성입니다.
DeploymentScaleType	배포 크기 조정 유형입니다.
ErrorAdditionalInfo	리소스 관리 오류 추가 정보입니다.
ErrorDetail	오류 세부 정보입니다.
ErrorResponse	오류 응답
RequestMatchPattern	
Sku	SKU를 나타내는 리소스 모델 정의
SkuTier	이 필드는 서비스에 둘 이상의 계층이 있지만 PUT에 필요하지 않은 경우 리소스 공급자가 구현해야 합니다.
systemData	리소스 만들기 및 마지막 수정과 관련된 메타데이터입니다.
ThrottlingRule	

CallRateLimit

호출 속도 제한 Cognitive Services 계정입니다.

 테이블 확장

Name	형식	Description
count	number	호출 속도 제한의 개수 값입니다.
renewalPeriod	number	통화 속도 제한의 갱신 기간(초)입니다.
rules	ThrottlingRule[]	

createdByType

리소스를 만든 ID의 형식입니다.

 테이블 확장

Name	형식	Description
Application	string	
Key	string	
ManagedIdentity	string	
User	string	

Deployment

Cognitive Services 계정 배포.

 테이블 확장

Name	형식	Description

etag	string	리소스 Etag.
id	string	리소스에 대한 정규화된 리소스 ID입니다. 예 - /subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/{resourceProviderNamespace}/{resourceType}/{resourceName}
name	string	리소스의 이름입니다.
properties	Deployment Properties	Cognitive Services 계정 배포의 속성입니다.
sku	Sku	SKU를 나타내는 리소스 모델 정의
systemData	systemData	리소스 만들기 및 마지막 수정과 관련된 메타데이터입니다.
type	string	리소스 형식입니다. 예: "Microsoft.Compute/virtualMachines" 또는 "Microsoft.Storage/storageAccounts"

DeploymentModel

Cognitive Services 계정 배포 모델의 속성입니다.

[\[+\] 테이블 확장](#)

Name	형식	Description
callRateLimit	CallRateLimit	호출 속도 제한 Cognitive Services 계정입니다.
format	string	배포 모델 형식입니다.
name	string	배포 모델 이름입니다.
source	string	선택 사항입니다. 배포 모델 원본 ARM 리소스 ID입니다.
version	string	선택 사항입니다. 배포 모델 버전입니다. 버전을 지정하지 않으면 기본 버전이 할당됩니다. 기본 버전은 모델에 따라 다르며 모델에 사용할 수 있는 새 버전이 있는 경우 변경될 수 있습니다. 모델의 기본 버전은 목록 모델 API에서 찾을 수 있습니다.

DeploymentModelVersionUpgradeOption

배포 모델 버전 업그레이드 옵션.

[\[+\] 테이블 확장](#)

Name	형식	Description
NoAutoUpgrade	string	
OnceCurrentVersionExpired	string	
OnceNewDefaultVersionAvailable	string	

DeploymentProperties

Cognitive Services 계정 배포의 속성입니다.

[\[+\] 테이블 확장](#)

Name	형식	Description
callRateLimit	CallRateLimit	호출 속도 제한 Cognitive Services 계정입니다.
capabilities	object	기능입니다.
model	DeploymentModel	Cognitive Services 계정 배포 모델의 속성입니다.
provisioningState	Deployment ProvisioningState	작업이 호출되었을 때 리소스의 상태 가져옵니다.
raiPolicyName	string	RAI 정책의 이름입니다.
rateLimits	ThrottlingRule[]	

scaleSettings	DeploymentScale Settings	Cognitive Services 계정 배포 모델의 속성입니다.
versionUpgradeOption	DeploymentModel VersionUpgrade Option	배포 모델 버전 업그레이드 옵션.

DeploymentProvisioningState

작업이 호출되었을 때 리소스의 상태 가져옵니다.

[\[+\] 테이블 확장](#)

Name	형식	Description
Accepted	string	
Canceled	string	
Creating	string	
Deleting	string	
Disabled	string	
Failed	string	
Moving	string	
Succeeded	string	

DeploymentScaleSettings

Cognitive Services 계정 배포 모델의 속성입니다.

[\[+\] 테이블 확장](#)

Name	형식	Description
activeCapacity	integer	배포 활성 용량. 이 값은 고객이 최근에 를 업데이트 <code>capacity</code> 한 경우와 <code>capacity</code> 다를 수 있습니다.
capacity	integer	배포 용량.
scaleType	DeploymentScale Type	배포 크기 조정 유형입니다.

DeploymentScaleType

배포 크기 조정 유형입니다.

[\[+\] 테이블 확장](#)

Name	형식	Description
Manual	string	
Standard	string	

ErrorAdditionalInfo

리소스 관리 오류 추가 정보입니다.

[\[+\] 테이블 확장](#)

Name	형식	Description
info	object	추가 정보입니다.
type	string	추가 정보 유형입니다.

ErrorDetail

오류 세부 정보입니다.

 테이블 확장

Name	형식	Description
additionalInfo	ErrorAdditionalInfo[]	오류 추가 정보입니다.
code	string	오류 코드입니다.
details	ErrorDetail[]	오류 세부 정보입니다.
message	string	오류 메시지입니다.
target	string	오류 대상입니다.

ErrorResponse

오류 응답

 테이블 확장

Name	형식	Description
error	ErrorDetail	Error 개체.

RequestMatchPattern

 테이블 확장

Name	형식	Description
method	string	
path	string	

Sku

SKU를 나타내는 리소스 모델 정의

 테이블 확장

Name	형식	Description
capacity	integer	SKU가 스케일 아웃/인을 지원하는 경우 용량 정수도 포함되어야 합니다. 리소스에 대해 규모 확장/감축이 불 가능한 경우 생략할 수 있습니다.
family	string	서비스에 동일한 SKU에 대해 서로 다른 세대의 하드웨어가 있는 경우 여기에서 캡처할 수 있습니다.
name	string	SKU의 이름입니다. 예 - P3. 일반적으로 letter+number 코드입니다.
size	string	SKU 크기입니다. 이름 필드가 계층과 다른 값의 조합인 경우 독립 실행형 코드가 됩니다.
tier	SkuTier	이 필드는 서비스에 둘 이상의 계층이 있지만 PUT에 필요하지 않은 경우 리소스 공급자가 구현해야 합니다.

SkuTier

이 필드는 서비스에 둘 이상의 계층이 있지만 PUT에 필요하지 않은 경우 리소스 공급자가 구현해야 합니다.

 테이블 확장

Name	형식	Description
Basic	string	
Enterprise	string	

Free	string
Premium	string
Standard	string

systemData

리소스 만들기 및 마지막 수정과 관련된 메타데이터입니다.

 테이블 확장

Name	형식	Description
createdAt	string	UTC(리소스 만들기)의 타임스탬프입니다.
createdBy	string	리소스를 만든 ID입니다.
createdByType	createdByType	리소스를 만든 ID의 형식입니다.
lastModifiedAt	string	리소스 마지막 수정의 타임스탬프(UTC)
lastModifiedBy	string	리소스를 마지막으로 수정한 ID입니다.
lastModifiedByType	createdByType	리소스를 마지막으로 수정한 ID 유형입니다.

ThrottlingRule

 테이블 확장

Name	형식	Description
count	number	
dynamicThrottlingEnabled	boolean	
key	string	
matchPatterns	RequestMatchPattern[]	
minCount	number	
renewalPeriod	number	

RAG를 사용하여 .NET 엔터프라이즈 채팅 샘플 시작

아티클 • 2024. 03. 20.

이 문서에서는 [.NET용 엔터프라이즈 채팅 앱 샘플](#)을 배포하고 실행하는 방법을 보여줍니다. 이 샘플은 가상 회사의 직원 복리후생에 대한 답변을 가져오기 위해 C#, Azure OpenAI Service 및 Azure AI 검색의 RAG(검색 증강 생성)를 사용하여 채팅 앱을 구현합니다. 직원 복리후생 채팅 앱에는 직원 핸드북, 복리후생 문서, 회사 역할 및 예상 결과치 목록을 포함한 PDF 파일이 포함되어 있습니다.

- [데모 비디오](#)

지금 시작

이 문서의 지침을 따르면 다음을 수행할 수 있습니다.

- Azure에 채팅 앱을 배포합니다.
- 직원 혜택에 대한 답변을 얻습니다.
- 응답 동작을 변경하려면 설정을 변경합니다.

이 절차를 완료하면 사용자 지정 코드를 사용하여 새 프로젝트 수정을 시작할 수 있습니다.

이 문서는 Azure Open AI Service 및 Azure AI 검색을 사용하여 채팅 앱을 빌드하는 방법을 보여 주는 문서 컬렉션의 일부입니다.

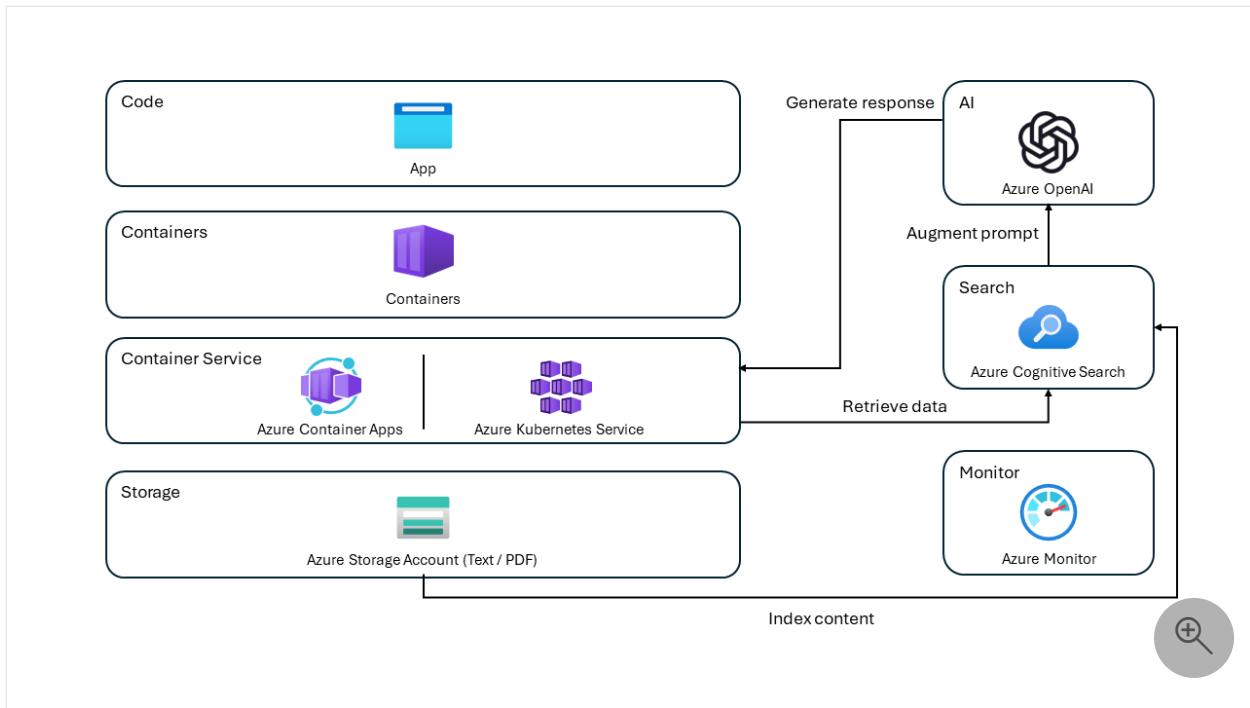
컬렉션의 다른 문서는 다음과 같습니다.

- [Python](#)
- [JavaScript](#)
- [Java](#)

아키텍처 개요

이 샘플 애플리케이션에서 Contoso Electronics라는 가상의 회사는 직원에게 복리후생, 내부 정책 및 직무 설명 및 역할에 대한 질문을 할 수 있는 채팅 앱 환경을 제공합니다.

채팅 앱의 아키텍처는 다음 다이어그램에 나와 있습니다.



- **사용자 인터페이스** - 애플리케이션의 채팅 인터페이스는 Blazor WebAssembly [애플리케이션입니다](#). 이 인터페이스는 사용자 쿼리를 허용하고, 요청을 애플리케이션 백 엔드로 라우팅하고, 생성된 응답을 표시합니다.
- **백 엔드** - 애플리케이션 백 엔드는 ASP.NET Core 최소 API[입니다](#). 백 엔드는 Blazor 정적 웹 애플리케이션을 호스트하며 서로 다른 서비스 간의 상호 작용을 오케스트레이션합니다. 이 애플리케이션에서 사용되는 서비스는 다음과 같습니다.
 - **Azure Cognitive Search** – Azure Storage 계정에 저장된 데이터의 문서를 인덱싱 합니다. 이렇게 하면 벡터 검색 기능을 사용하여 [문서를 검색](#) 할 수 있습니다.
 - **Azure OpenAI 서비스** – 응답을 생성하는 LLM(대규모 언어 모델)을 제공합니다. [의미 체계 커널](#)은 Azure OpenAI 서비스와 함께 사용하여 보다 복잡한 AI 워크플로를 오케스트레이션합니다.

비용

이 아키텍처의 대부분의 리소스는 기본 또는 사용량 가격 책정 계층을 사용합니다. 사용량 가격 책정은 사용량을 기준으로 책정됩니다. 즉, 사용한 만큼만 비용을 지불하면 됩니다. 이 문서를 완료하려면 요금이 발생하지만 요금은 최소화됩니다. 문서가 완료되면 리소스를 삭제하여 요금 발생을 중지할 수 있습니다.

자세한 내용은 [Azure 샘플: 샘플 리포지토리의 비용](#)을 참조하세요.

필수 조건

이 문서를 완료하는 데 필요한 모든 종속성을 갖춘 [개발 컨테이너](#) 환경을 사용할 수 있습니다. GitHub Codespaces(브라우저)에서 개발 컨테이너를 실행하거나 Visual Studio Code를 사용하여 로컬로 실행할 수 있습니다.

이 문서를 진행하려면 다음 필수 조건이 필요합니다.

Codespaces(권장)

1. Azure 구독 - [체험 구독 만들기](#)
2. Azure 계정 권한 - Azure 계정에는 [사용자 액세스 관리자](#) 또는 [소유자](#)와 같은 Microsoft.Authorization/roleAssignments/write 권한이 있어야 합니다.
3. 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한. 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 엽니다.
4. GitHub 계정

개방형 개발 환경

이 문서를 완료하려면 모든 종속성이 설치된 개발 환경으로 지금 시작합니다.

GitHub Codespaces(권장)

[GitHub Codespaces](#)는 사용자 인터페이스로 [웹용 Visual Studio Code](#)를 사용하여 GitHub에서 관리하는 개발 컨테이너를 실행합니다. 가장 간단한 개발 환경을 위해서는 GitHub Codespaces를 사용하여 이 문서를 완료하는 데 필요한 올바른 개발자 도구와 종속성을 미리 설치합니다.

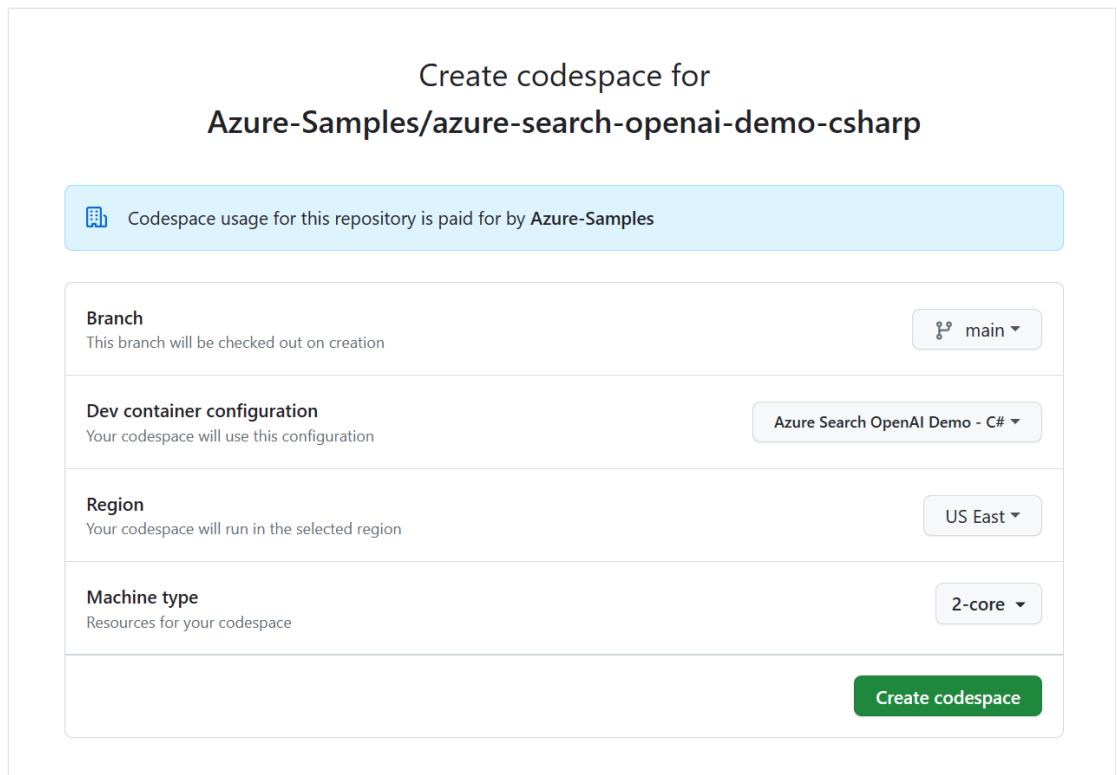
① 중요

모든 GitHub 계정은 2개의 코어 인스턴스를 사용하여 매월 최대 60시간 동안 Codespaces를 무료로 사용할 수 있습니다. 자세한 내용은 [GitHub Codespaces 월별 포함 스토리지 및 코어 시간](#)을 참조하세요.

1. [Azure-Samples/azure-search-openai-demo-csharp](#) GitHub 리포지토리의 `main` 분기에 새 GitHub Codespace를 만드는 프로세스를 시작합니다.
2. 개발 환경과 설명서를 동시에 사용하려면 다음 단추를 마우스 오른쪽 단추로 클릭하고 새 창에서 링크 열기를 선택합니다.

[GitHub Codespaces에서 이 프로젝트 열기](#)

3. [codespace 만들기](#) 페이지에서 codespace 구성 설정을 검토한 후 새 codespace 만들기를 선택합니다.



4. codespace가 생성될 때까지 기다립니다. 이 프로세스에는 몇 분 정도 걸릴 수 있습니다.

5. 화면 하단의 터미널에서 Azure 개발자 CLI를 사용하여 Azure에 로그인합니다.

```
Bash
azd auth login
```

6. 터미널에서 코드를 복사한 다음 브라우저에 붙여넣습니다. 지침에 따라 Azure 계정으로 인증합니다.

7. 이 문서의 나머지 작업은 이 개발 컨테이너의 컨텍스트에서 수행됩니다.

배포 및 실행

샘플 리포지토리에는 Azure에 채팅 앱을 배포하는 데 필요한 모든 코드와 구성 파일이 포함되어 있습니다. 다음 단계에서는 샘플을 Azure에 배포하는 과정을 안내합니다.

Azure에 채팅 앱 배포

ⓘ 중요

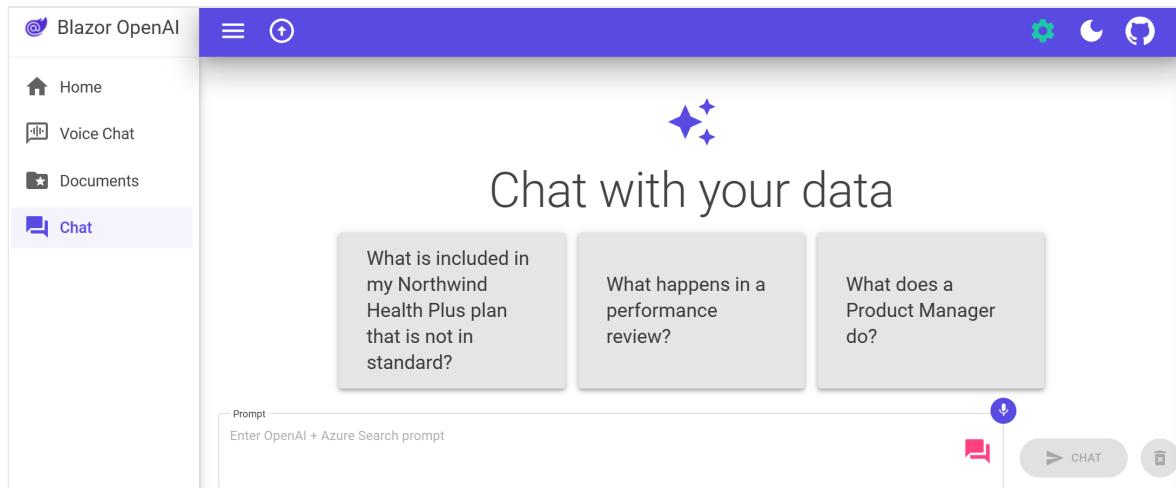
이 섹션에서 만들어진 Azure 리소스는 주로 Azure AI 검색 리소스에서 즉각적인 비용이 발생합니다. 이러한 리소스는 명령이 완전히 실행되기 전에 중단하더라도 비용이 발생할 수 있습니다.

1. 다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 프로비전하고 소스 코드를 배포합니다.

```
Bash
```

```
azd up
```

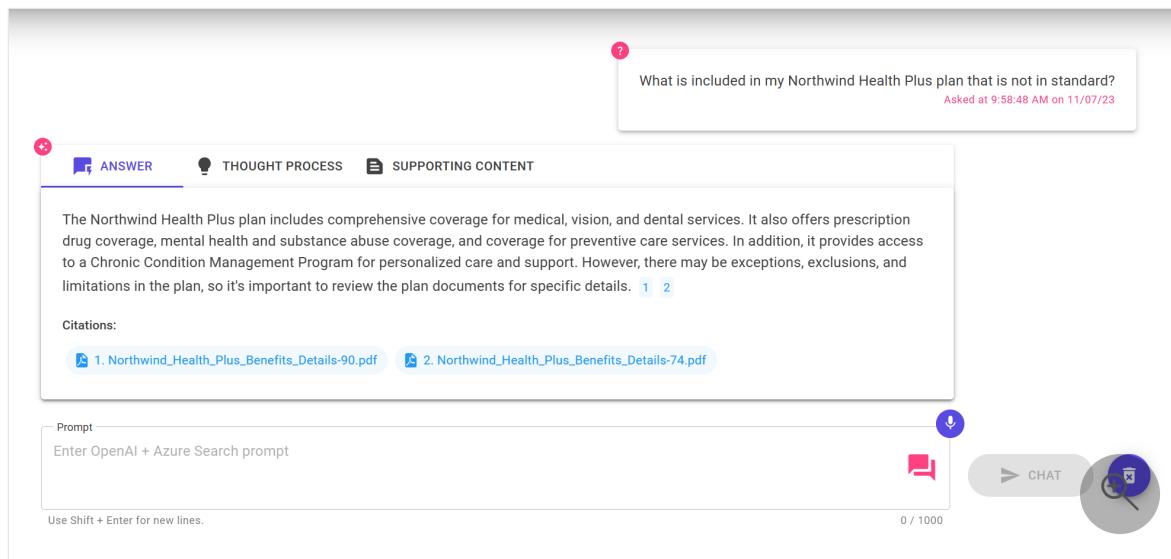
2. 환경 이름을 입력하라는 메시지가 표시되면 짧고 소문자로 유지합니다. 예: `myenv`. 리소스 그룹 이름의 일부로 사용됩니다.
3. 메시지가 표시되면 리소스를 만들 구독을 선택합니다.
4. 처음 위치를 선택하라는 메시지가 표시되면 가까운 위치를 선택합니다. 이 위치는 호스팅을 포함한 대부분의 리소스에 사용됩니다.
5. OpenAI 모델의 위치를 묻는 메시지가 표시되면 가까운 위치를 선택합니다. 첫 번째 위치와 동일한 위치를 사용할 수 있는 경우 해당 위치를 선택합니다.
6. 앱이 배포될 때까지 기다립니다. 배포가 완료되는 데 최대 20분이 걸릴 수 있습니다.
7. 애플리케이션이 성공적으로 배포되면 터미널에 URL이 표시됩니다.
8. 브라우저에서 채팅 애플리케이션을 열려면 `Deploying service web`이라고 표시된 URL을 선택합니다.



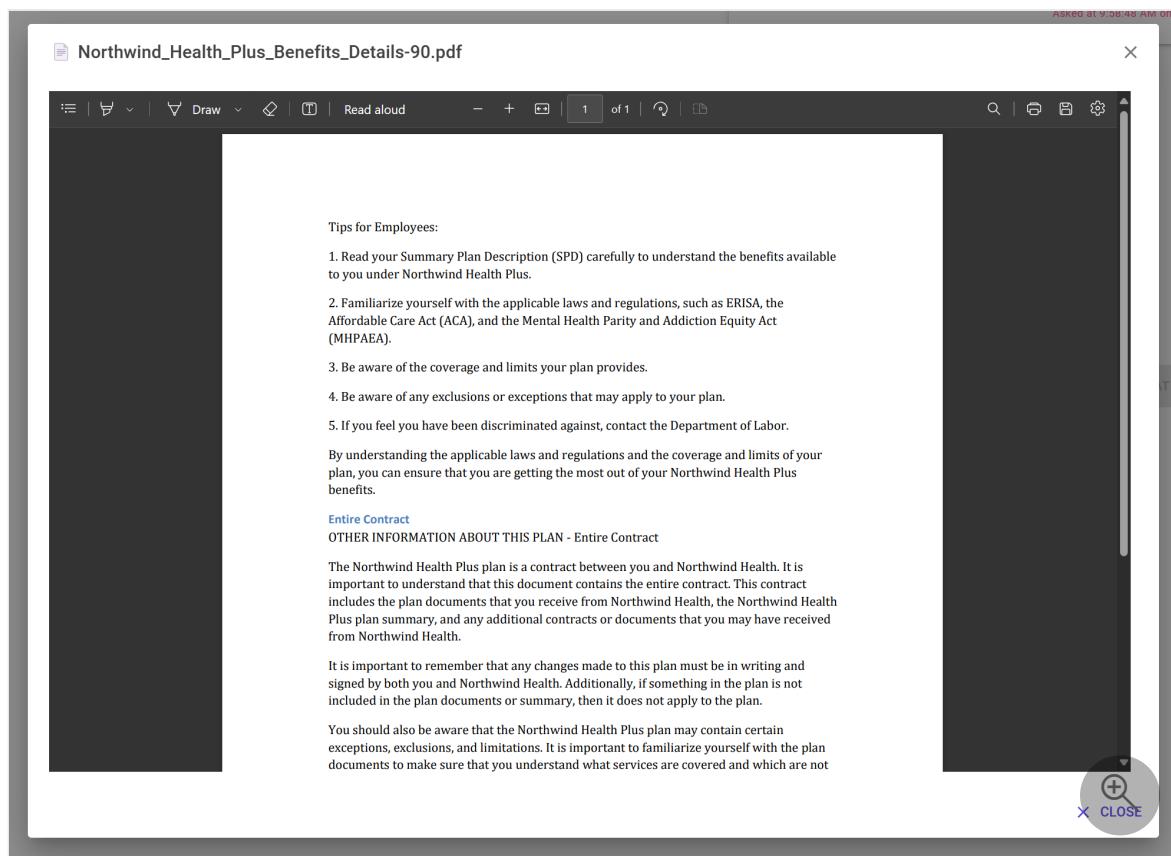
채팅 앱을 사용하여 PDF 파일에서 답변 가져오기

채팅 앱에는 PDF 파일의 직원 복리후생 정보가 미리 로드되어 있습니다. 채팅 앱을 이용하여 혜택에 대해 질문할 수 있습니다. 다음 단계에서는 채팅 앱을 사용하는 과정을 안내합니다.

1. 브라우저에서 왼쪽 탐색 메뉴를 사용하여 채팅 페이지로 이동합니다.
2. 채팅 텍스트 상자에 "표준에 포함되지 않은 Northwind Health Plus 플랜에는 무엇이 있나요"을 선택하거나 입력합니다.



3. 답변에서 인용을 선택합니다. 정보의 원본을 표시하는 팝업 창이 열립니다.



4. 답변이 어떻게 생성되었는지 이해하려면 답변 상자 상단에 있는 탭 사이를 이동합니다.

탭	설명
사고 과정	이는 채팅의 상호 작용에 대한 스크립트입니다. 시스템 프롬프트(content)와 사용자 질문(content)을 볼 수 있습니다.
지원 내용	여기에는 사용자의 질문에 답변하기 위한 정보와 원본 재질이 포함됩니다. 원본 자료 인용 수는 개발자 설정나와 있습니다. 기본값은 3입니다.
인용	인용이 포함된 원본 페이지가 표시됩니다.

5. 완료되면 답변 탭으로 다시 이동합니다.

채팅 앱 설정을 사용하여 응답 동작 변경

채팅의 지능은 OpenAI 모델과 해당 모델과 상호 작용하는 데 사용되는 설정에 의해 결정됩니다.

Configure Answer Generation

Override prompt template

Override prompt template

Retrieve this many documents from search

3

Exclude category

Exclude category

Use semantic ranker for retrieval

Retrieval Mode

Text Hybrid Vector

Use query-contextual summaries
instead of whole documents

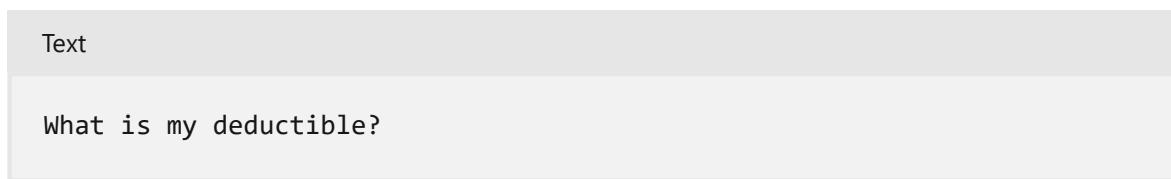
Suggest follow-up questions

X CLOSE

설정	설명
프롬프트 템플릿 재정의	이는 답변을 생성하는 데 사용되는 프롬프트입니다.
이 많은 검색 결과를 검색합니다.	답변을 생성하는 데 사용되는 검색 결과의 수입니다. 인용의 사고 과정 및 지원 콘텐츠 탭에서 반환된 이러한 원본을 확인할 수 있습니다.
범주 제외	검색 결과에서 제외되는 문서 범주입니다.
검색을 위해 의미 순위매기기 사용	이는 기계 학습을 사용하여 검색 결과의 관련성을 높이는 Azure AI 검색 의 기능입니다.
검색 모드	벡터 + 텍스트 는 검색 결과가 문서의 텍스트와 문서의 포함을 기반으로 한다는 의미입니다. 벡터 는 검색 결과가 문서의 포함을 기반으로 한다는 의미입니다. 텍스트 는 검색 결과가 문서의 텍스트를 기반으로 한다는 의미입니다.
전체 문서 대신 쿼리 컨텍스트 요약을 사용합니다.	<code>Use semantic ranker</code> 와 <code>Use query-contextual summaries</code> 를 모두 선택하면 LLM은 순위가 가장 높은 문서의 모든 구절 대신 주요 구절에서 추출된 캡션을 사용합니다.
후속 질문 제안	채팅 앱에서 답변에 따라 후속 질문을 제안하도록 합니다.

다음 단계에서는 설정을 변경하는 과정을 안내합니다.

- 브라우저에서 페이지 오른쪽 상단에 있는 기어 아이콘을 선택합니다.
- 후속 질문 제안** 확인란을 선택하고 동일한 질문을 다시 질문합니다.



채팅은 다음과 같은 후속 질문 제안을 반환합니다.

- "네트워크 외부 서비스에 대한 비용 분담금은 얼마인가요?"
- "예방 진료 서비스에도 공제액이 적용되나요?"
- "처방약 공제액은 어떻게 적용되나요?"

- 설정 탭에서 **검색에 의미 순위매기기 사용**을 선택 취소합니다.
- 같은 질문을 다시 물어보세요.



- 답변의 차이점은 무엇인가요?

의미 순위매기기를 사용한 답변은 단일 답변(The deductible for the Northwind Health Plus plan is \$2,000 per year)을 제공했습니다.

의미 순위매기기가 없는 답변은 덜 직접적인 답변인 Based on the information provided, it is unclear what your specific deductible is. The Northwind Health Plus plan has different deductible amounts for in-network and out-of-network services, and there is also a separate prescription drug deductible. I would recommend checking with your provider or referring to the specific benefits details for your plan to determine your deductible amount를 반환했습니다.

리소스 정리

Azure 리소스 정리

이 문서에서 만들어진 Azure 리소스는 Azure 구독에 요금이 청구됩니다. 앞으로 이러한 리소스가 필요하지 않을 것으로 예상되는 경우 추가 요금이 발생하지 않도록 삭제합니다.

다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 삭제하고 소스 코드를 제거합니다.

Bash

```
azd down --purge
```

GitHub Codespaces 정리

GitHub Codespaces

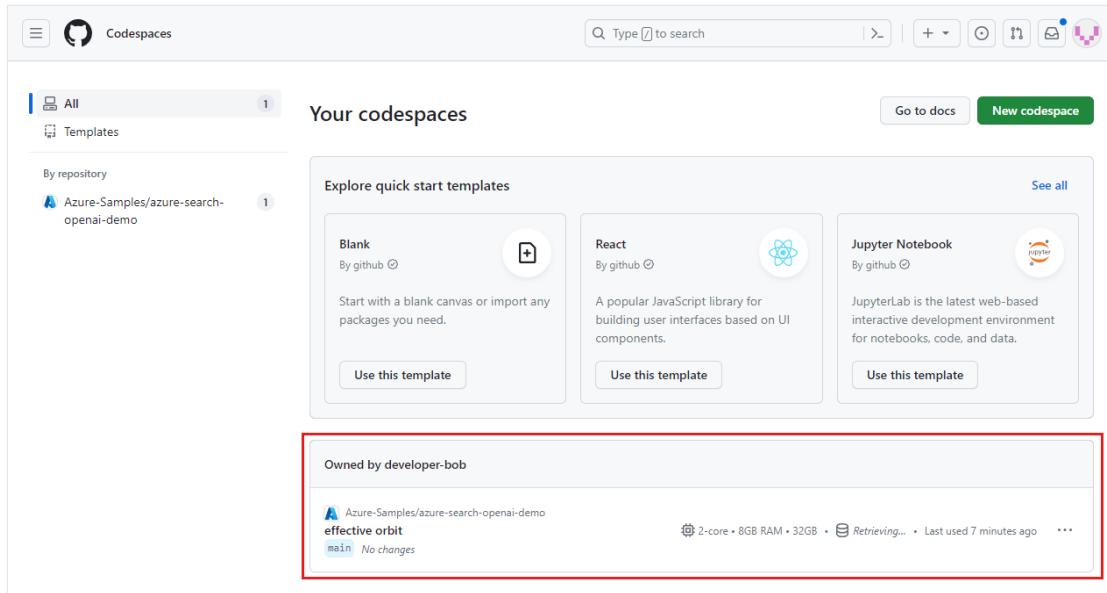
GitHub Codespaces 환경을 삭제하면 계정에 대해 얻을 수 있는 코어당 무료 사용 권한을 최대화할 수 있습니다.

ⓘ 중요

GitHub 계정의 자격에 대한 자세한 내용은 [GitHub Codespaces 월별 포함된 스토리지 및 코어 시간](#)을 참조하세요.

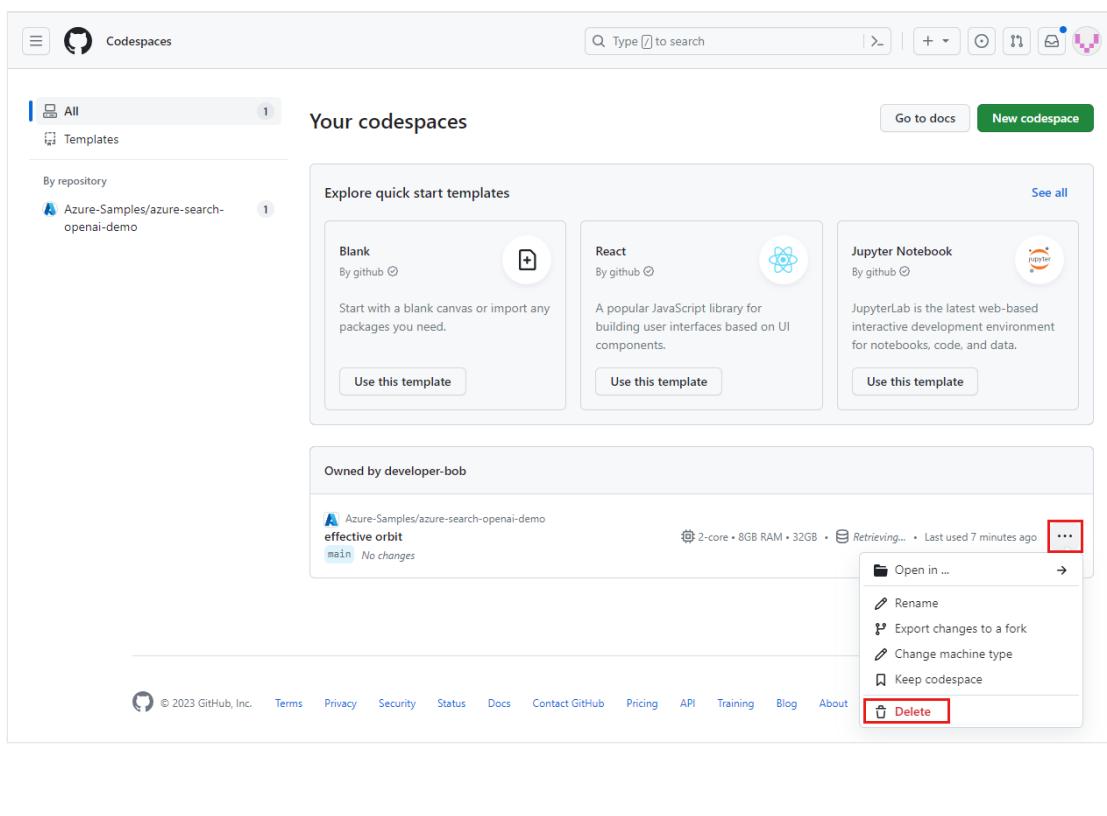
1. GitHub Codespaces 대시보드(<https://github.com/codespaces>)에 로그인합니다.

2. Azure-Samples/azure-search-openai-demo-csharp GitHub 리포지토리에서 제공된 현재 실행 중인 codespace를 찾습니다.



The screenshot shows the GitHub Codespaces interface. In the center, there's a section titled "Your codespaces" with a heading "Explore quick start templates". Below this, a list of codespaces is shown, with one specifically highlighted by a red box. This highlighted codespace is owned by "developer-bob" and is associated with the repository "Azure-Samples/azure-search-openai-demo" and branch "main". The status bar indicates "No changes". To the right of the status bar, it says "Retrieving..." and "Last used 7 minutes ago". At the bottom right of the highlighted box, there are three dots (...).

3. codespace에 대한 상황에 맞는 메뉴를 열고 삭제를 선택합니다.



This screenshot is similar to the previous one, showing the "Your codespaces" page. The same codespace is highlighted with a red box. A context menu has been opened at the bottom right of the highlighted box. The menu items visible are: "Open in ...", "Rename", "Export changes to a fork", "Change machine type", "Keep codespace", and "Delete". The "Delete" option is highlighted with a red box.

도움말 보기

이 샘플 리포지토리는 문제 해결 정보 [문제 해결 정보](#)를 제공합니다.

문제가 해결되지 않으면 리포지토리의 문제 [문제](#)에 문제를 기록합니다.

다음 단계

- 엔터프라이즈 채팅 앱 GitHub 리포지토리 ↗
- Azure OpenAI를 사용하여 채팅 앱 빌드 ↗ 모범 사례 솔루션 아키텍처
- Azure AI 검색을 사용한 생성 AI 앱의 액세스 제어 ↗
- Azure API Management를 사용하여 엔터프라이즈급 OpenAI 솔루션 빌드 ↗
- 하이브리드 검색 및 순위 지정 기능으로 탁월한 벡터 검색 성능 제공 ↗

GitHub에서 Microsoft와 공동 작업

이 콘텐츠의 원본은 GitHub에서 찾을 수 있으며, 여기서 문제와 끌어오기 요청을 만들고 검토할 수도 있습니다. 자세한 내용은 [참여자 가이드](#)를 참조하세요.

.NET

.NET 피드백

.NET은(는) 오픈 소스 프로젝트입니다. 다음 링크를 선택하여 피드백을 제공해 주세요.

☞ 설명서 문제 열기

☞ 제품 사용자 의견 제공

RAG를 사용하여 Java 엔터프라이즈 채팅 샘플 시작

아티클 • 2024. 04. 13.

이 문서에서는 Java용 엔터프라이즈 채팅 앱 샘플을 배포하고 실행하는 방법을 보여 줍니다. 이 샘플에서는 가상 회사의 직원 혜택에 대한 답변을 얻기 위해 Azure AI Search에서 Java, Azure OpenAI Service 및 RAG(검색 증강 세대)를 사용하여 채팅 앱을 구현합니다. 앱은 직원 핸드북, 혜택 문서 및 회사 역할 및 기대치 목록을 포함한 PDF 파일로 시드됩니다.

- [데모 비디오](#)

[지금 시작](#)

이 문서의 지침을 따르면 다음을 수행할 수 있습니다.

- Azure에 채팅 앱을 배포합니다.
- 직원 혜택에 대한 답변을 얻습니다.
- 응답 동작을 변경하려면 설정을 변경합니다.

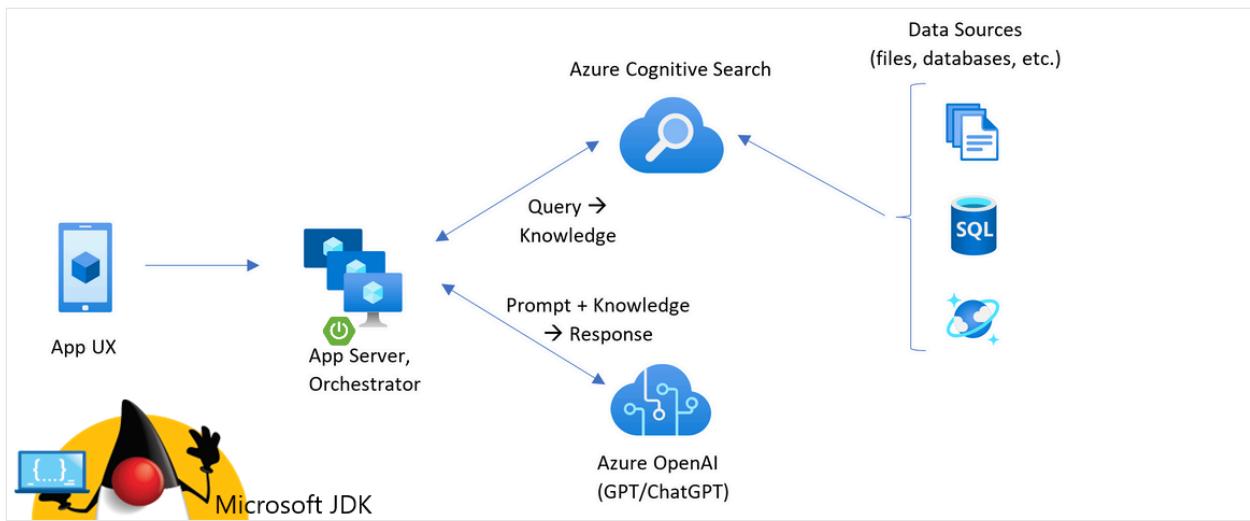
이 문서를 완료하면 사용자 지정 코드를 사용하여 새 프로젝트 수정을 시작할 수 있습니다.

이 문서는 Azure OpenAI Service 및 Azure AI Search를 사용하여 채팅 앱을 빌드하는 방법을 보여 주는 문서 모음의 일부입니다. 컬렉션의 다른 문서는 다음과 같습니다.

- [.Net](#)
- [JavaScript](#)
- [Python](#)

아키텍처 개요

다음 다이어그램은 채팅 앱의 간단한 아키텍처를 보여 줍니다.



아키텍처의 주요 구성 요소는 다음과 같습니다.

- 대화형 채팅 환경을 호스팅하는 웹 애플리케이션입니다.
- 사용자 고유의 데이터에서 답변을 가져오는 Azure AI Search 리소스입니다.
- 다음을 제공하는 Azure OpenAI Service:
 - 고유의 데이터에 대한 검색을 향상시키는 키워드입니다.
 - OpenAI 모델의 답변.
 - ada 모델의 포함

비용

이 아키텍처의 대부분의 리소스는 기본 또는 사용량 가격 책정 계층을 사용합니다. 사용량 가격 책정은 사용량을 기준으로 책정됩니다. 즉, 사용한 만큼만 비용을 지불하면 됩니다. 이 문서를 완료하려면 요금이 발생하지만 요금은 최소화됩니다. 문서가 완료되면 리소스를 삭제하여 요금 발생을 중지할 수 있습니다.

샘플 리포지토리의 [비용에 대해 자세히](#) 알아봅니다.

필수 조건

이 문서를 완료하는 데 필요한 모든 종속성을 갖춘 [개발 컨테이너](#) 환경을 사용할 수 있습니다. GitHub Codespaces(브라우저)에서 개발 컨테이너를 실행하거나 Visual Studio Code를 사용하여 로컬로 실행할 수 있습니다.

이 문서를 사용하려면 다음 필수 구성 요소가 필요합니다.

Codespaces(권장)

1. Azure 구독 - [체험 구독 만들기](#)

2. Azure 계정 권한 - Azure 계정에는 [사용자 액세스 관리자](#) 또는 [소유자](#)와 같은 Microsoft.Authorization/roleAssignments/write 권한이 있어야 합니다.
3. 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한. 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 엽니다.
4. GitHub 계정

개방형 개발 환경

이 문서를 완료하려면 모든 종속성이 설치된 개발 환경으로 지금 시작합니다.

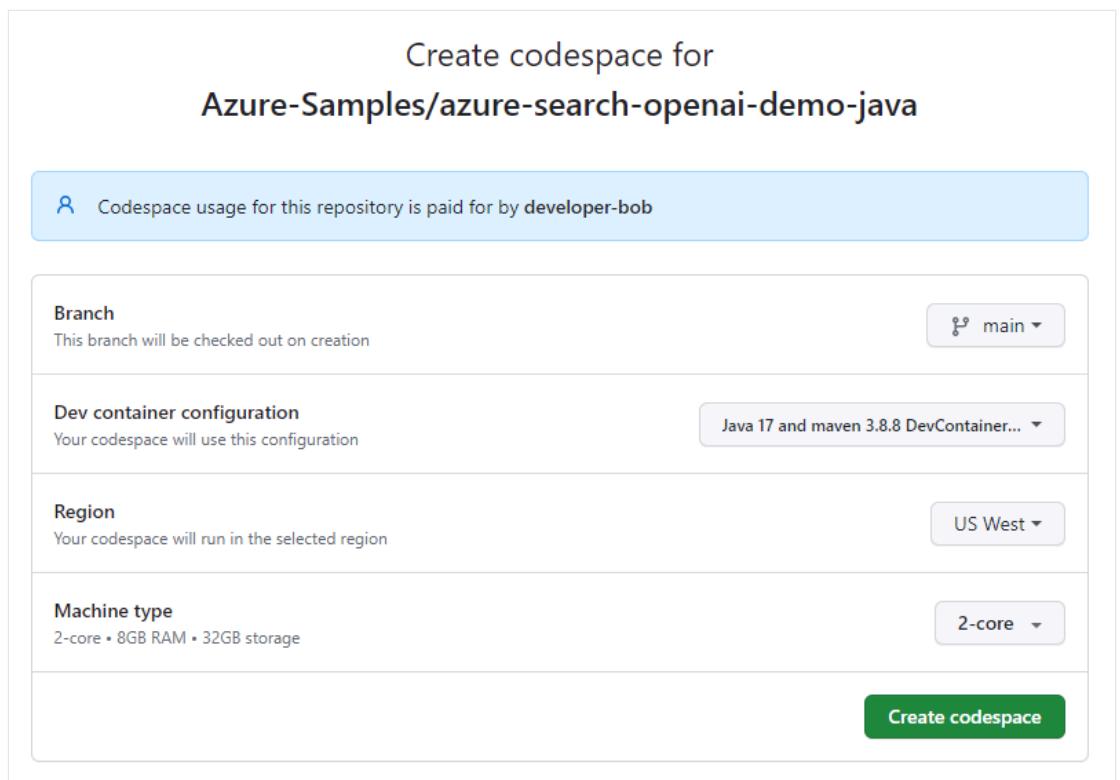
GitHub Codespaces(권장)

[GitHub Codespaces](#) 는 사용자 인터페이스로 [웹용 Visual Studio Code](#) 를 사용하여 GitHub에서 관리하는 개발 컨테이너를 실행합니다. 가장 간단한 개발 환경을 위해서는 GitHub Codespaces를 사용하여 이 문서를 완료하는 데 필요한 올바른 개발자 도구와 종속성을 미리 설치합니다.

ⓘ 중요

모든 GitHub 계정은 2개의 코어 인스턴스를 사용하여 매월 최대 60시간 동안 Codespaces를 무료로 사용할 수 있습니다. 자세한 내용은 [GitHub Codespaces 월별 포함 스토리지 및 코어 시간](#) 을 참조하세요.

1. [Azure-Samples/azure-search-openai-demo-java](#) GitHub 리포지토리의 `main` 분기에 새 GitHub Codespace를 만드는 프로세스를 시작합니다.
 2. 개발 환경과 설명서를 동시에 사용하려면 다음 단추를 마우스 오른쪽 단추로 클릭하고 새 창에서 링크 열기를 선택합니다.
- [GitHub Codespaces에서 이 프로젝트 열기](#)
3. [codespace 만들기](#) 페이지에서 codespace 구성 설정을 검토한 다음, 새 codespace 만들기를 선택합니다.



4. codespace가 생성될 때까지 기다립니다. 이 프로세스에는 몇 분 정도 걸릴 수 있습니다.

5. 화면 하단의 터미널에서 Azure 개발자 CLI를 사용하여 Azure에 로그인합니다.

```
Bash
azd auth login
```

6. 터미널에서 코드를 복사한 다음 브라우저에 붙여넣습니다. 지침에 따라 Azure 계정으로 인증합니다.

7. 이 문서의 나머지 작업은 이 개발 컨테이너의 컨텍스트에서 수행됩니다.

배포 및 실행

샘플 리포지토리에는 Azure에 채팅 앱을 배포하는 데 필요한 모든 코드와 구성 파일이 포함되어 있습니다. 다음 단계에서는 샘플을 Azure에 배포하는 과정을 안내합니다.

Azure에 채팅 앱 배포

ⓘ 중요

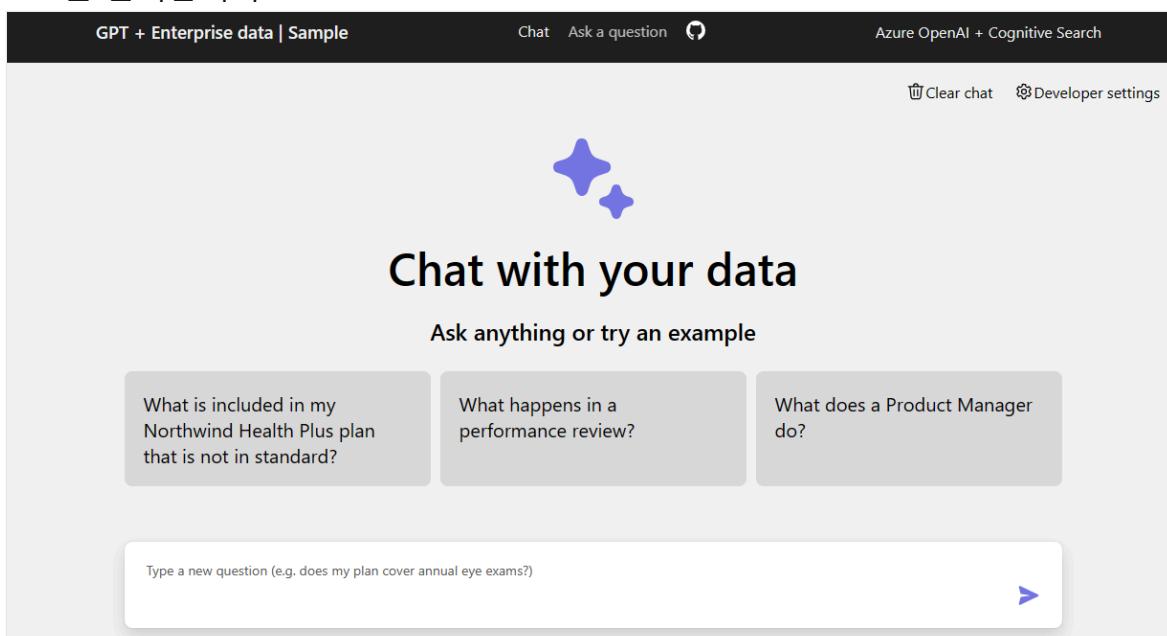
이 섹션에서 만들어진 Azure 리소스는 주로 Azure AI 검색 리소스에서 즉각적인 비용이 발생합니다. 이러한 리소스는 명령이 완전히 실행되기 전에 중단하더라도 비용이 발생할 수 있습니다.

1. 다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 프로비전하고 소스 코드를 배포합니다.

```
Bash
```

```
azd up
```

2. 환경 이름을 입력하라는 메시지가 표시되면 짧고 소문자로 유지합니다. 예: `myenv`. 리소스 그룹 이름의 일부로 사용됩니다.
3. 메시지가 표시되면 리소스를 만들 구독을 선택합니다.
4. 처음 위치를 선택하라는 메시지가 표시되면 가까운 위치를 선택합니다. 이 위치는 호스팅을 포함한 대부분의 리소스에 사용됩니다.
5. OpenAI 모델의 위치를 묻는 메시지가 표시되면 가까운 위치를 선택합니다. 첫 번째 위치와 동일한 위치를 사용할 수 있는 경우 해당 위치를 선택합니다.
6. 앱이 배포될 때까지 기다립니다. 배포를 완료하는 데 5~10분이 걸릴 수 있습니다.
7. 애플리케이션이 성공적으로 배포되면 터미널에 URL이 표시됩니다.
8. 브라우저에서 채팅 애플리케이션을 열려면 `Deploying service web`이라고 표시된 URL을 선택합니다.



채팅 앱을 사용하여 PDF 파일에서 답변 가져오기

채팅 앱에는 PDF 파일의 직원 복리후생 정보가 미리 로드되어 있습니다. 채팅 앱을 이용하여 혜택에 대해 질문할 수 있습니다. 다음 단계에서는 채팅 앱을 사용하는 과정을 안내합니다.

1. 브라우저에서 채팅 텍스트 상자에 "표준이 아닌 Northwind Health Plus 플랜에 포함된 항목"을 선택하거나 입력합니다.

The screenshot shows a web-based AI chat interface. At the top, it says "GPT + Enterprise data | Sample". Below that are buttons for "Chat" and "Ask a question" with a microphone icon. To the right, it says "Azure OpenAI + Cognitive Search". In the main area, there's a text input box containing the question: "What is included in my Northwind Health Plus plan that is not in standard?". A response card follows, enclosed in a red box:

The Northwind Health Plus plan includes coverage for emergency services, mental health and substance abuse coverage, and out-of-network services, which are not included in the standard plan. The Health Plus plan also offers a wider range of prescription drug coverage, including specialty drugs, compared to the standard plan. [1. Benefit_Options-2.pdf](#)

Below the response card is a text input field with placeholder text "Type a new question (e.g. does my plan cover annual eye exams?)". To the right of this field is a search icon consisting of a magnifying glass and a plus sign.

2. 답변에서 인용 중 하나를 선택합니다.

This screenshot is similar to the previous one, showing the same AI interface. The question "What is included in my Northwind Health Plus plan that is not in standard?" has been asked again. The response card contains the same text as before, with the citation link "[1. Benefit_Options-2.pdf](#)" highlighted by a red box.

Below the response card is a text input field with placeholder text "Type a new question (e.g. does my plan cover annual eye exams?)". To the right of this field is a search icon consisting of a magnifying glass and a plus sign.

3. 오른쪽 창에서 탭을 사용하여 답변이 생성된 방법을 이해합니다.

[+] 테이블 확장

탭	설명
사고 과정	이는 채팅의 상호 작용에 대한 스크립트입니다.
지원 내용	여기에는 사용자의 질문에 답변하기 위한 정보와 원본 재질이 포함됩니다.
인용	여기에는 인용이 포함된 PDF 페이지가 표시됩니다.

4. 완료되면 선택한 탭을 다시 선택하여 창을 닫습니다.

채팅 앱 설정을 사용하여 응답 동작 변경

채팅 앱의 인텔리전스는 OpenAI 모델 및 모델과 상호 작용하는 데 사용되는 설정에 따라 결정됩니다.

Configure answer generation ×

Override prompt template

Retrieve this many search results:

3

Exclude category

Use semantic ranker for retrieval
 Use query-contextual summaries instead of whole documents
 Suggest follow-up questions

Retrieval mode *

Vectors + Text (Hybrid) ▼

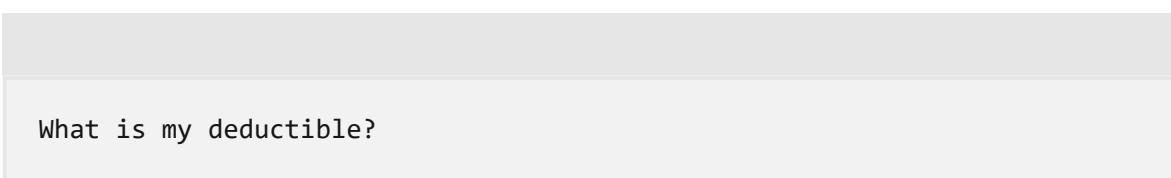
Stream chat completion responses

Close

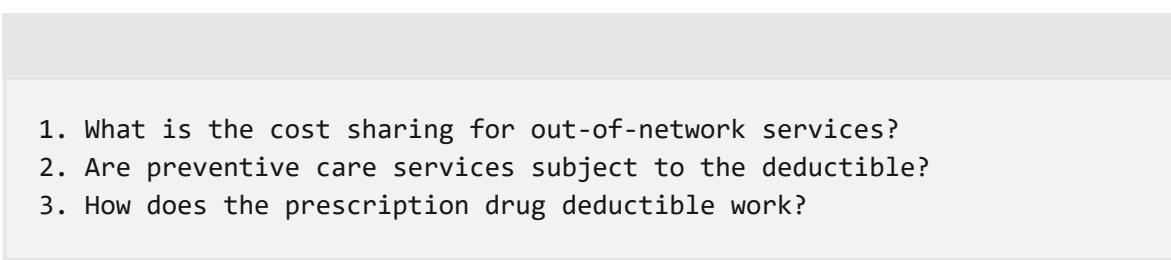
설정	설명
프롬프트 템플릿 재정의	이는 답변을 생성하는 데 사용되는 프롬프트입니다.
이 많은 검색 결과를 검색합니다.	답변을 생성하는 데 사용되는 검색 결과의 수입니다. 인용의 사고 과정 및 지원 콘텐츠 탭에서 반환된 이러한 원본을 확인할 수 있습니다.
범주 제외	검색 결과에서 제외되는 문서 범주입니다.
검색을 위해 의미 순위매기기 사용	이는 기계 학습을 사용하여 검색 결과의 관련성을 높이는 Azure AI 검색 의 기능입니다.
전체 문서 대신 쿼리 컨텍스트 요약을 사용합니다.	<code>Use semantic ranker</code> 와 <code>Use query-contextual summaries</code> 를 모두 선택하면 LLM은 순위가 가장 높은 문서의 모든 구절 대신 주요 구절에서 추출된 캡션을 사용합니다.
후속 질문 제안	채팅 앱에서 답변에 따라 후속 질문을 제안하도록 합니다.
검색 모드	벡터 + 텍스트 는 검색 결과가 문서의 텍스트와 문서의 포함을 기반으로 한다는 의미입니다. 벡터 는 검색 결과가 문서의 포함을 기반으로 한다는 의미입니다. 텍스트 는 검색 결과가 문서의 텍스트를 기반으로 한다는 의미입니다.
채팅 완료 응답 스트리밍	응답에 대한 전체 답변을 사용할 수 있게 될 때까지 기다리는 대신 응답을 스트리밍합니다.

다음 단계에서는 설정을 변경하는 과정을 안내합니다.

- 브라우저에서 개발자 설정 **탭**을 선택합니다.
- 후속 질문 제안** 확인란을 선택하고 동일한 질문을 다시 질문합니다.



채팅은 다음과 같은 제안된 후속 질문을 반환했습니다.



- 설정 탭에서 **검색에 의미 순위매기기 사용**을 선택 취소합니다.
- 같은 질문을 다시 하시겠습니까?

What is my deductible?

5. 답변의 차이점은 무엇인가요?

예를 들어 의미 체계 순위자를 사용한 응답은 단일 대답 The deductible for the Northwind Health Plus plan is \$2,000 per year을 제공했습니다.

의미 체계 순위가 없는 리포지토리는 대답을 반환했으며, 답변을 Based on the information provided, it is unclear what your specific deductible is. The Northwind Health Plus plan has different deductible amounts for in-network and out-of-network services, and there is also a separate prescription drug deductible. I would recommend checking with your provider or referring to the specific benefits details for your plan to determine your deductible amount 얻기 위해 더 많은 작업이 필요했습니다.

리소스 정리

Azure 리소스 정리

이 문서에서 만들어진 Azure 리소스는 Azure 구독에 요금이 청구됩니다. 앞으로 이러한 리소스가 필요하지 않을 것으로 예상되는 경우 추가 요금이 발생하지 않도록 삭제합니다.

다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 삭제하고 소스 코드를 제거합니다.

Bash

```
azd down --purge
```

GitHub Codespaces 정리

GitHub Codespaces

GitHub Codespaces 환경을 삭제하면 계정에 대해 얻을 수 있는 코어당 무료 사용 권한을 최대화할 수 있습니다.

ⓘ 중요

GitHub 계정의 자격에 대한 자세한 내용은 [GitHub Codespaces 월별 포함된 스토리지 및 코어 시간](#)을 참조하세요.

1. GitHub Codespaces 대시보드(<https://github.com/codespaces>)에 로그인합니다.
2. [Azure-Samples/azure-search-openai-demo-java](#) GitHub 리포지토리에서 제공된 현재 실행 중인 codespace를 찾습니다.

The screenshot shows the GitHub Codespaces dashboard. On the left, there's a sidebar with 'All' selected under 'Codespaces' and a 'Templates' section. Below that is a 'By repository' section with a single item: 'Azure-Samples/azure-search-openai-demo-java'. The main area is titled 'Your codespaces' and contains several cards for quick start templates: 'Blank', 'React', 'Jupyter Notebook', and '.NET'. Below these is a section titled 'Owned by developer-bob' which lists the user's own codespaces. One codespace, 'potential train', is shown with a red box around it. It has a GitHub icon, the repository name, a branch 'main', and a status 'No changes'. To the right of the codespace details are resource information (2-core, 8GB RAM, 32GB storage, 3.37 GB used) and a timestamp ('Last used about 19 hours ago'). There are also three dots (...).

3. codespace에 대한 상황에 맞는 메뉴를 열고 삭제를 선택합니다.

This screenshot is similar to the previous one but shows the context menu for the 'potential train' codespace being opened. The menu items are: 'Open in ...', 'Rename', 'Export changes to a fork', 'Change machine type', 'Keep codespace', and 'Delete'. The 'Delete' option is highlighted with a red box.

질문에 대한 답변은 어떻게 합니까?

앱은 2개의 앱으로 구분됩니다.

- Vite 빌드 도구와 함께 React 프레임워크를 사용하는 프런트 엔드 JavaScript 애플리케이션입니다.
- 백 엔드 Java 애플리케이션이 질문에 답변합니다.

백 엔드 /chat API는 답변을 가져오는 프로세스를 단계:

- RAG 옵션 빌드: 답변을 생성하는 데 사용할 옵션 집합을 만듭니다.
- RAG 옵션을 사용하여 접근 방식 만들기: 검색 기반 모델과 생성 기반 모델의 조합을 사용하여 정확하고 자연스러운 응답을 생성하는 방법을 만듭니다.
- RAG 옵션 및 이전 대화를 사용하여 접근 방식을 실행합니다. 접근 방식 및 RAG 옵션을 사용하여 이전 대화에 따라 답변을 생성합니다. 답변에는 응답을 생성하는 데 사용된 문서에 대한 정보가 포함됩니다.

도움말 보기

이 샘플 리포지토리는 문제 해결 정보 [문제 해결 정보](#)를 제공합니다.

발급된 문제가 해결되지 않으면 리포지토리의 문제에 [문제에](#) 문제를 기록합니다.

다음 단계

- 엔터프라이즈 채팅 앱 GitHub 리포지토리 [문제 해결 정보](#)
- Azure OpenAI를 사용하여 채팅 앱 빌드 [모범 사례 솔루션 아키텍처](#)
- Azure AI 검색을 사용한 생성 AI 앱의 액세스 제어 [문제 해결 정보](#)
- Azure API Management를 사용하여 엔터프라이즈급 OpenAI 솔루션 빌드 [문제 해결 정보](#)
- 하이브리드 검색 및 순위 지정 기능으로 탁월한 벡터 검색 성능 제공 [문제 해결 정보](#)

피드백

이 페이지가 도움이 되었나요?

 Yes

 No

Microsoft Q&A에서 도움말 보기

RAG를 사용하여 JavaScript 엔터프라이즈 채팅 샘플 시작

아티클 • 2024. 03. 19.

이 문서에서는 JavaScript용 엔터프라이즈 채팅 앱 샘플을 배포하고 실행하는 방법을 보여 줍니다. 이 샘플에서는 Azure AI Search에서 JavaScript, Azure OpenAI Service 및 RAG(검색 증강 세대)를 사용하여 채팅 앱을 구현하여 임대 속성에 대한 답변을 얻습니다. 임대 속성 채팅 앱은 개인 정보 보호 정책, 서비스 약관 및 지원을 포함하여 markdown 파일(*.md)의 데이터로 시드됩니다.

- [데모 JavaScript](#) - 전체 스택 비디오
- [데모 JavaScript](#) - Python 백 엔드 비디오가 있는 프런트 엔드

지금 시작

이 문서의 지침을 따르면 다음을 수행할 수 있습니다.

- Azure에 채팅 앱을 배포합니다.
- 임대 속성 웹 사이트 정보에 대한 답변을 가져옵니다.
- 응답 동작을 변경하려면 설정을 변경합니다.

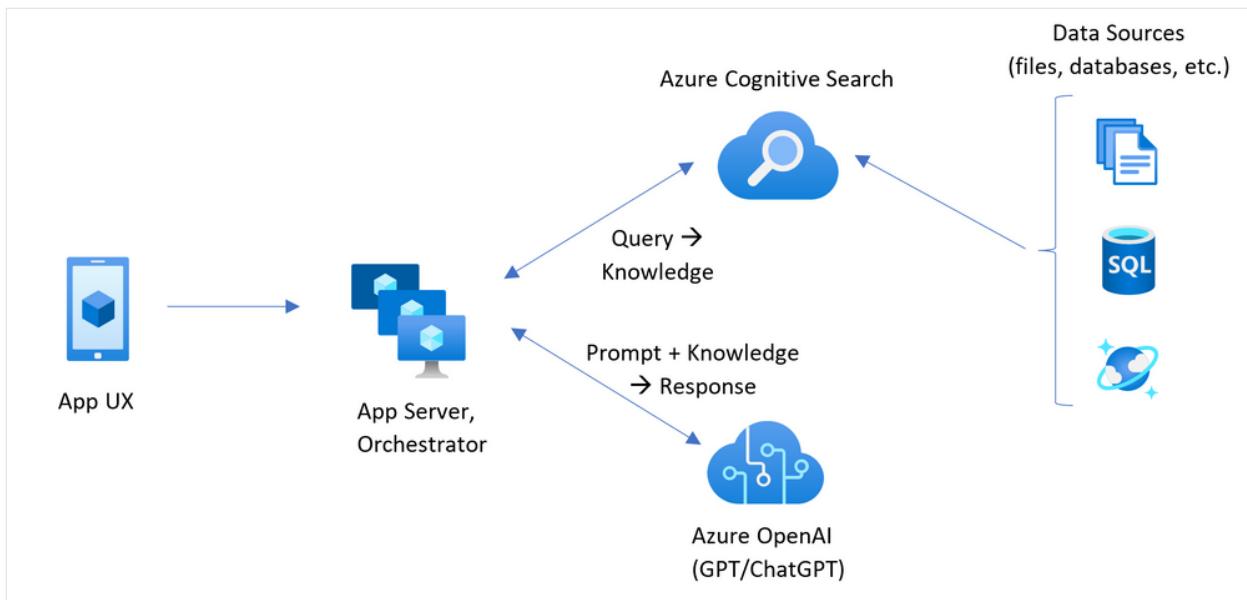
이 문서를 완료하면 사용자 지정 코드 및 데이터로 새 프로젝트 수정을 시작할 수 있습니다.

이 문서는 Azure OpenAI Service 및 Azure AI Search를 사용하여 채팅 앱을 빌드하는 방법을 보여 주는 문서 모음의 일부입니다. 컬렉션의 다른 문서는 다음과 같습니다.

- [.NET](#)
- [Java](#)
- [Python](#)

아키텍처 개요

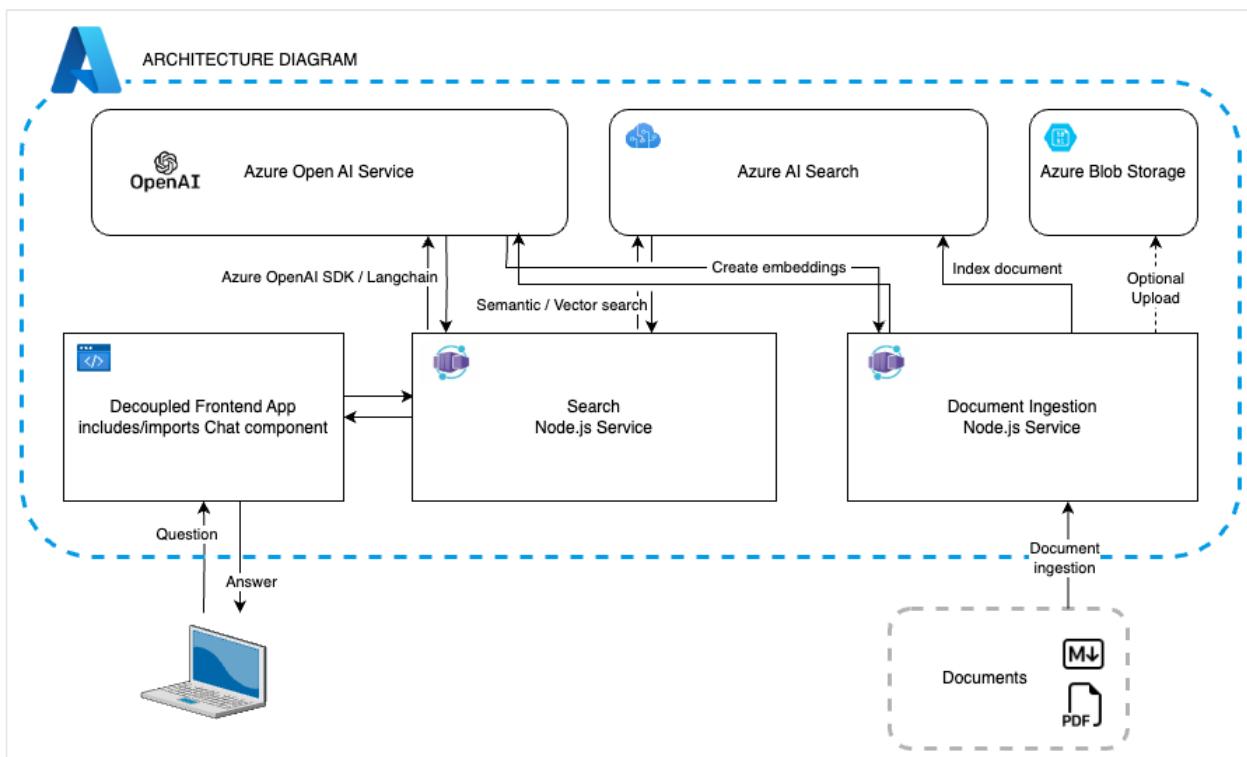
다음 다이어그램은 채팅 앱의 간단한 아키텍처를 보여 줍니다.



채팅 샘플 애플리케이션은 Contoso Real Estate라는 가상의 회사를 위해 빌드되었으며, 지능형 채팅 환경을 통해 고객은 제품 사용에 대한 지원 질문을 할 수 있습니다. 샘플 데이터에는 서비스 약관, 개인정보 보호 정책 및 지원 가이드를 설명하는 문서 집합이 포함되어 있습니다. 문서는 배포 중에 아키텍처로 수집됩니다.

애플리케이션은 다음을 비롯한 여러 구성 요소에서 만들어집니다.

- **검색 서비스:** 검색 및 검색 기능을 제공하는 백 엔드 서비스입니다.
- **인덱서 서비스:** 데이터를 인덱싱하고 검색 인덱스를 만드는 서비스입니다.
- **웹앱:** 사용자 인터페이스를 제공하고 사용자와 백 엔드 서비스 간의 상호 작용을 오케스트레이션하는 프런트 엔드 웹 애플리케이션입니다.



비용

이 아키텍처의 대부분의 리소스는 기본 또는 사용량 가격 책정 계층을 사용합니다. 사용량 가격 책정은 사용량을 기준으로 책정됩니다. 즉, 사용한 만큼만 비용을 지불하면 됩니다. 이 문서를 완료하려면 요금이 발생하지만 요금은 최소화됩니다. 문서를 완료하면 리소스를 삭제하여 요금 발생을 중지할 수 있습니다.

샘플 리포지토리의 [비용에 대해](#) 자세히 알아봅니다.

필수 조건

이 문서를 완료하는 데 필요한 모든 종속성을 갖춘 [개발 컨테이너](#) 환경을 사용할 수 있습니다. GitHub Codespaces(브라우저)에서 개발 컨테이너를 실행하거나 Visual Studio Code를 사용하여 로컬로 실행할 수 있습니다.

이 문서를 사용하려면 다음 필수 구성 요소가 필요합니다.

Codespaces(권장)

1. Azure 구독 - [체험 구독 만들기](#)
2. Azure 계정 권한 - Azure 계정에는 [사용자 액세스 관리자](#) 또는 [소유자](#)와 같은 Microsoft.Authorization/roleAssignments/write 권한이 있어야 합니다.
3. 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한. 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 업니다.
4. GitHub 계정

개방형 개발 환경

이 문서를 완료하려면 모든 종속성이 설치된 개발 환경으로 지금 시작합니다.

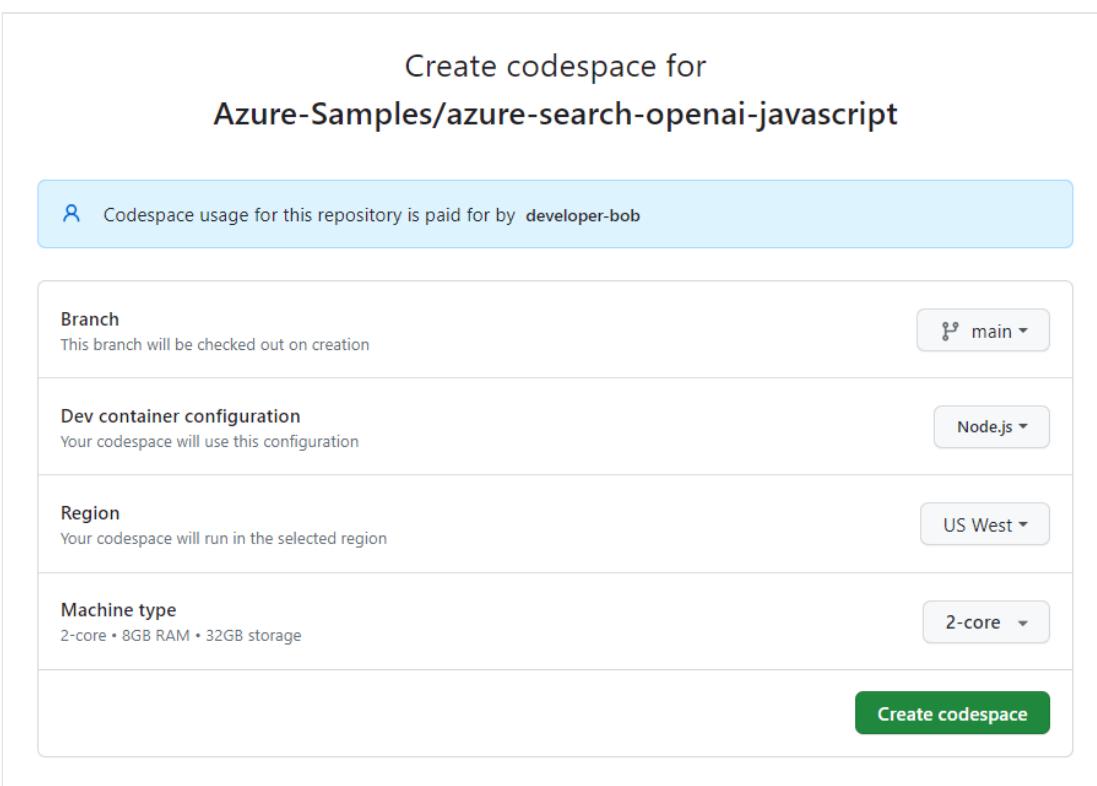
GitHub Codespaces(권장)

[GitHub Codespaces](#)는 사용자 인터페이스로 [웹용 Visual Studio Code](#)를 사용하여 GitHub에서 관리하는 개발 컨테이너를 실행합니다. 가장 간단한 개발 환경을 위해서는 GitHub Codespaces를 사용하여 이 문서를 완료하는 데 필요한 올바른 개발자 도구와 종속성을 미리 설치합니다.

① 중요

모든 GitHub 계정은 2개의 코어 인스턴스를 사용하여 매월 최대 60시간 동안 Codespaces를 무료로 사용할 수 있습니다. 자세한 내용은 [GitHub Codespaces 월별 포함 스토리지 및 코어 시간](#)을 참조하세요.

1. [Azure-Samples/azure-search-openai-javascript](#) GitHub 리포지토리의 `main` 분기에 새 GitHub Codespace를 만드는 프로세스를 시작합니다.
2. 개발 환경과 설명서를 동시에 사용하려면 다음 단추를 마우스 오른쪽 단추로 클릭하고 새 창에서 링크 열기를 선택합니다.
3. [codespace 만들기](#) 페이지에서 codespace 구성 설정을 검토한 다음, 새 codespace 만들기를 선택합니다.



4. codespace가 생성될 때까지 기다립니다. 이 프로세스에는 몇 분 정도 걸릴 수 있습니다.
5. 화면 하단의 터미널에서 Azure 개발자 CLI를 사용하여 Azure에 로그인합니다.

```
Bash
azd auth login
```

- 터미널에서 코드를 복사한 다음 브라우저에 붙여넣습니다. 지침에 따라 Azure 계정으로 인증합니다.
- 이 문서의 나머지 작업은 이 개발 컨테이너의 컨텍스트에서 수행됩니다.

배포 및 실행

샘플 리포지토리에는 Azure에 채팅 앱을 배포하는 데 필요한 모든 코드와 구성 파일이 포함되어 있습니다. 다음 단계에서는 샘플을 Azure에 배포하는 과정을 안내합니다.

Azure에 채팅 앱 배포

ⓘ 중요

이 섹션에서 만들어진 Azure 리소스는 주로 Azure AI 검색 리소스에서 즉각적인 비용이 발생합니다. 이러한 리소스는 명령이 완전히 실행되기 전에 중단하더라도 비용이 발생할 수 있습니다.

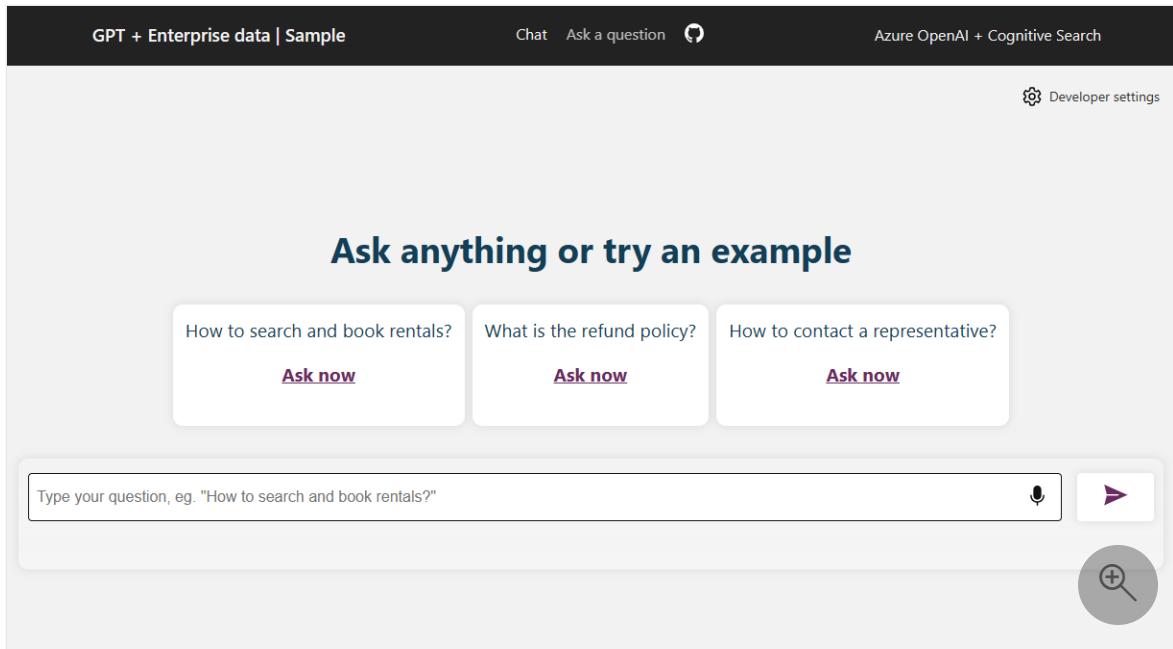
- 다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 프로비전하고 소스 코드를 배포합니다.

Bash

```
azd up
```

- 환경 이름을 입력하라는 메시지가 표시되면 짧고 소문자로 유지합니다. 예를 들어 `myenv`입니다. 리소스 그룹 이름의 일부로 사용됩니다.
- 메시지가 표시되면 리소스를 만들 구독을 선택합니다.
- 처음 위치를 선택하라는 메시지가 표시되면 가까운 위치를 선택합니다. 이 위치는 호스팅을 포함한 대부분의 리소스에 사용됩니다.
- OpenAI 모델의 위치를 묻는 메시지가 표시되면 가까운 위치를 선택합니다. 첫 번째 위치와 동일한 위치를 사용할 수 있는 경우 해당 위치를 선택합니다.
- 앱이 배포될 때까지 기다립니다. 배포를 완료하는 데 5~10분이 걸릴 수 있습니다.
- 애플리케이션이 성공적으로 배포되면 터미널에 URL이 표시됩니다.

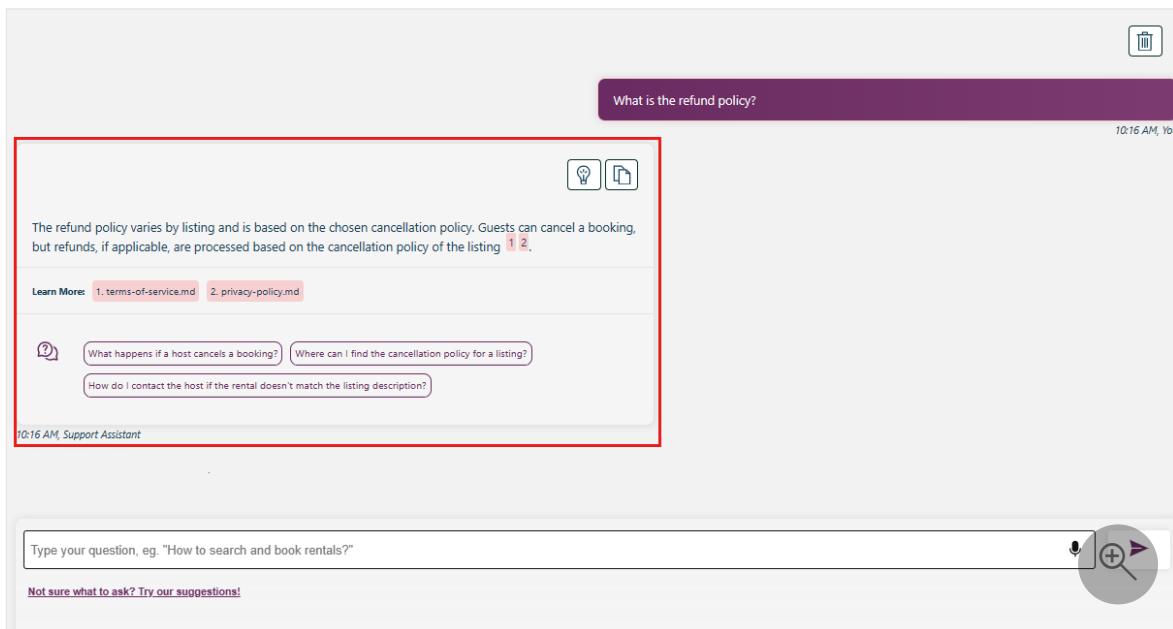
8. 브라우저에서 채팅 애플리케이션을 열려면 Deploying service web이라고 표시된 URL을 선택합니다.



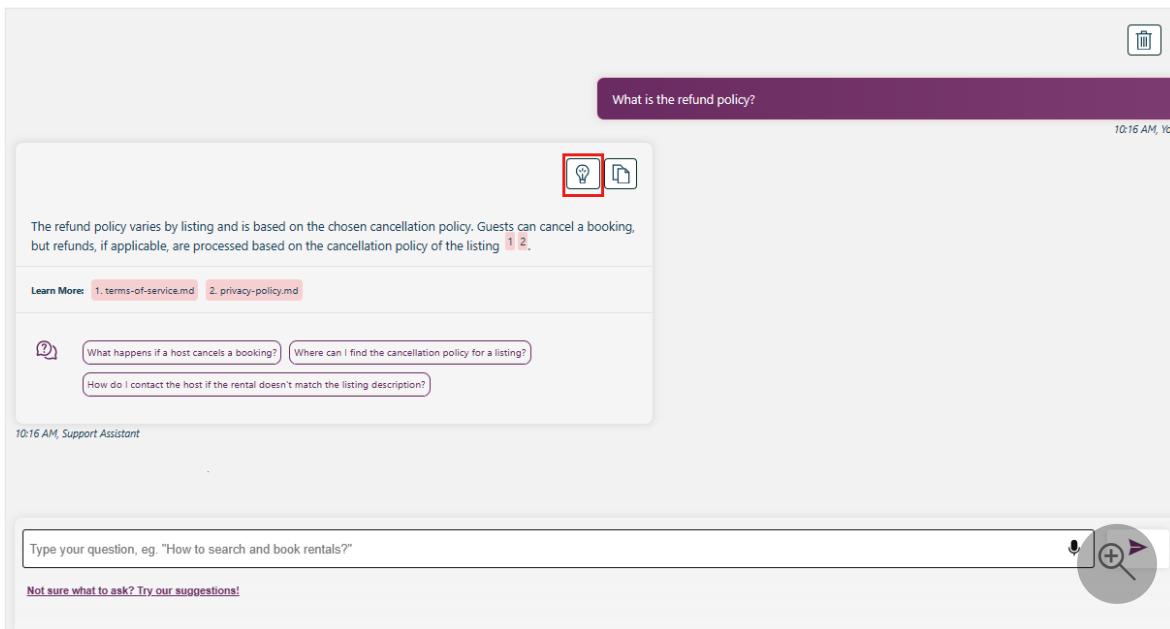
채팅 앱을 사용하여 markdown 파일에서 답변 가져오기

채팅 앱은 markdown 파일 카탈로그의 [임대 정보로 미리 로드됩니다](#). 채팅 앱을 사용하여 임대 프로세스에 대한 질문을 할 수 있습니다. 다음 단계에서는 채팅 앱을 사용하는 과정을 안내합니다.

1. 브라우저에서 페이지 아래쪽의 텍스트 상자에 환불 정책이란?을 선택하거나 입력합니다.



2. 답변에서 생각 프로세스 표시를 선택합니다.



3. 오른쪽 창에서 템을 사용하여 답변이 생성된 방법을 이해합니다.

▣ 테이블 확장

탭	설명
사고 과정	이는 채팅의 상호 작용에 대한 스크립트입니다. 시스템 프롬프트(content)와 사용자 질문(content)을 볼 수 있습니다.
지원 내용	여기에는 사용자의 질문에 답변하기 위한 정보와 원본 재질이 포함됩니다. 원본 자료 인용 수는 개발자 설정나와 있습니다. 기본값은 3입니다.
인용	인용이 포함된 원래 페이지가 표시됩니다.

4. 완료되면 템 위에 X가 표시된 숨기기 단추를 선택합니다.

채팅 앱 설정을 사용하여 응답 동작 변경

채팅 앱의 인텔리전스는 OpenAI 모델 및 모델과 상호 작용하는 데 사용되는 설정에 따라 결정됩니다.

Configure answer generation

Override prompt template

Retrieve this many search results:

3

Exclude category

Use semantic ranker for retrieval

Use query-contextual summaries instead of whole documents

Suggest follow-up questions

Retrieval mode *

Vectors + Text (Hybrid)

Stream chat completion responses

Close

테이블 확장

설정	설명
프롬프트 템플릿 재정의	이는 답변을 생성하는 데 사용되는 프롬프트입니다.
이 많은 검색 결과를 검색합니다.	답변을 생성하는 데 사용되는 검색 결과의 수입니다. 인용의 사고 과정 및 지원 과정에서 검색합니다. 콘텐츠 탭에서 반환된 이러한 원본을 확인할 수 있습니다.
범주 제외	검색 결과에서 제외되는 문서 범주입니다.
검색을 위해 의미 순위 매기기 사용	이는 기계 학습을 사용하여 검색 결과의 관련성을 높이는 Azure AI 검색 의 기능입니다.
전체 문서 대신 쿼리 컨텍스트 요	Use semantic ranker 와 Use query-contextual summaries 를 모두 선택하면 LLM은 순위가 가장 높은 문서의 모든 구절 대신 주요 구절에서 추출된 캡션을

설정	설명
약을 사용합니다.	사용합니다.
후속 질문 제안	채팅 앱에서 답변에 따라 후속 질문을 제안하도록 합니다.
검색 모드	벡터 + 텍스트 는 검색 결과가 문서의 텍스트와 문서의 포함을 기반으로 한다는 의미입니다. 벡터 는 검색 결과가 문서의 포함을 기반으로 한다는 의미입니다. 텍스트 는 검색 결과가 문서의 텍스트를 기반으로 한다는 의미입니다.
채팅 완료 응답 스트리밍	응답에 대한 전체 답변을 사용할 수 있게 될 때까지 기다리는 대신 응답을 스트리밍합니다.

다음 단계에서는 설정을 변경하는 과정을 안내합니다.

- 브라우저에서 개발자 설정 **탭**을 선택합니다.
- 검색 **box** 대신 쿼리 컨텍스트 요약 사용을 선택하고 동일한 질문을 다시 요청합니다.

```
What happens if the rental doesn't fit the description?
```

채팅은 다음과 같은 보다 간결한 답변으로 반환되었습니다.

리소스 정리

Azure 리소스 정리

이 문서에서 만들어진 Azure 리소스는 Azure 구독에 요금이 청구됩니다. 앞으로 이러한 리소스가 필요하지 않을 것으로 예상되는 경우 추가 요금이 발생하지 않도록 삭제합니다.

다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 삭제하고 소스 코드를 제거합니다.

```
Bash
```

```
azd down --purge
```

GitHub Codespaces 정리

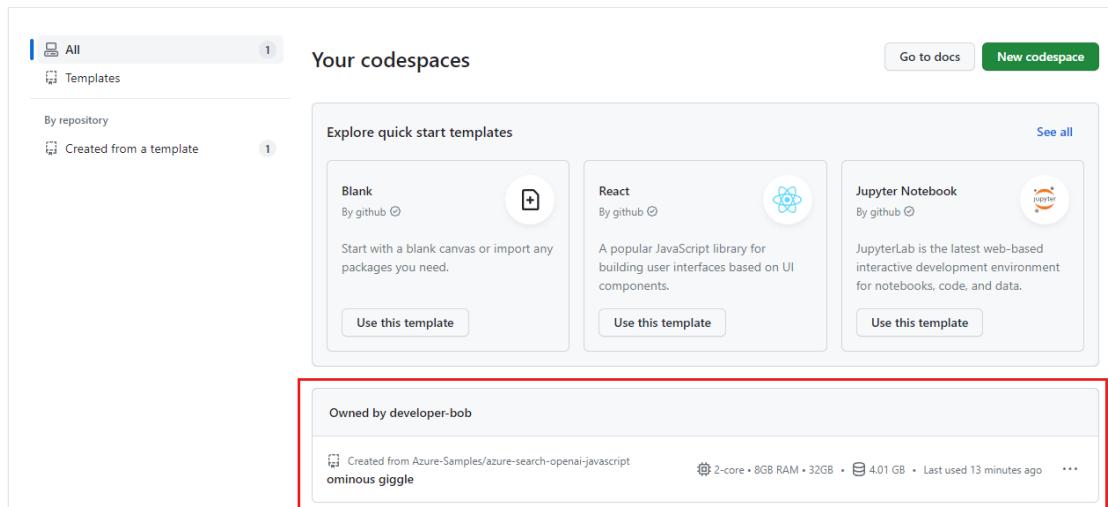
GitHub Codespaces

GitHub Codespaces 환경을 삭제하면 계정에 대해 얻을 수 있는 코어당 무료 사용 권한을 최대화할 수 있습니다.

① 중요

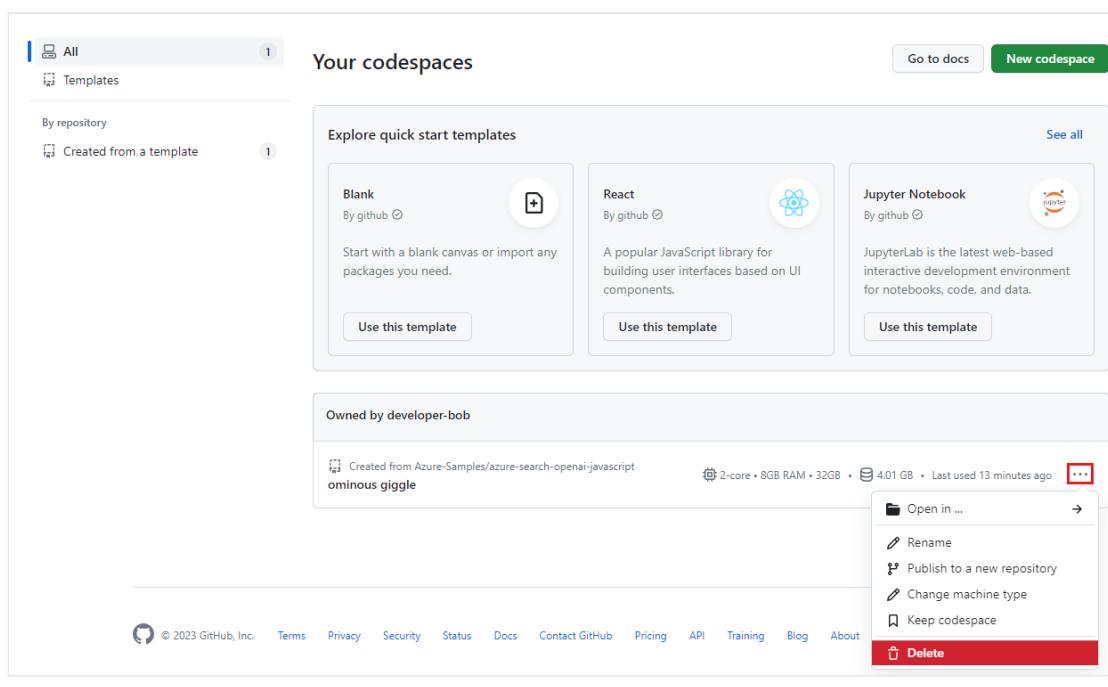
GitHub 계정의 자격에 대한 자세한 내용은 [GitHub Codespaces 월별 포함된 스토리지 및 코어 시간](#)을 참조하세요.

1. GitHub Codespaces 대시보드(<https://github.com/codespaces>)에 로그인합니다.
2. [Azure-Samples/azure-search-openai-javascript](#) GitHub 리포지토리에서 제공된 현재 실행 중인 codespace를 찾습니다.



The screenshot shows the 'Your codespaces' page. At the top, there are filters for 'All' (selected), 'Templates', and repository filters ('By repository' and 'Created from a template'). Below this is a section titled 'Explore quick start templates' with three options: 'Blank', 'React', and 'Jupyter Notebook'. The main list starts with a codespace named 'ominous giggle', which is highlighted with a red box. It shows the owner 'developer-bob', the creation source 'Created from Azure-Samples/azure-search-openai-javascript', and resource details: 2-core, 8GB RAM, 32GB storage, 4.01 GB used, and last used 13 minutes ago. To the right of the list is a 'See all' link.

3. codespace에 대한 상황에 맞는 메뉴를 열고 **삭제**를 선택합니다.



The screenshot shows the same 'Your codespaces' page as the previous one, but the 'ominous giggle' codespace is now selected. A context menu is open next to its card, listing several options: 'Open in ...', 'Rename', 'Publish to a new repository', 'Change machine type', 'Keep codespace', and 'Delete'. The 'Delete' option is highlighted with a red box at the bottom of the menu.

도움말 보기

이 샘플 리포지토리는 문제 해결 정보 [☞](#)를 제공합니다.

발급된 문제가 해결되지 않으면 리포지토리의 문제에 [☞](#) 문제를 기록합니다.

다음 단계

- 엔터프라이즈 채팅 앱 GitHub 리포지토리 [☞](#)
- Azure OpenAI를 사용하여 채팅 앱 빌드 [☞](#) 모범 사례 솔루션 아키텍처
- Azure AI 검색을 사용한 생성 AI 앱의 액세스 제어 [☞](#)
- Azure API Management를 사용하여 엔터프라이즈급 OpenAI 솔루션 빌드 [☞](#)
- 하이브리드 검색 및 순위 지정 기능으로 탁월한 벡터 검색 성능 제공 [☞](#)

RAG를 사용하여 Python 엔터프라이즈 채팅 샘플 시작

아티클 • 2024. 03. 05.

이 문서에서는 Python [용 엔터프라이즈 채팅 앱 샘플을 배포하고 실행하는](#) 방법을 보여줍니다. 이 샘플에서는 가상 회사의 직원 혜택에 대한 답변을 얻기 위해 Azure AI Search에서 Python, Azure OpenAI Service 및 [RAG\(검색 증강 세대\)](#)를 사용하여 채팅 앱을 구현합니다. 앱은 직원 핸드북, 혜택 문서 및 회사 역할 및 기대치 목록을 포함한 PDF 파일로 시드됩니다.

- [데모 비디오](#)

지금 시작

이 문서의 지침을 따르면 다음을 수행할 수 있습니다.

- Azure에 채팅 앱을 배포합니다.
- 직원 혜택에 대한 답변을 얻습니다.
- 응답 동작을 변경하려면 설정을 변경합니다.

이 절차를 완료하면 사용자 지정 코드를 사용하여 새 프로젝트 수정을 시작할 수 있습니다.

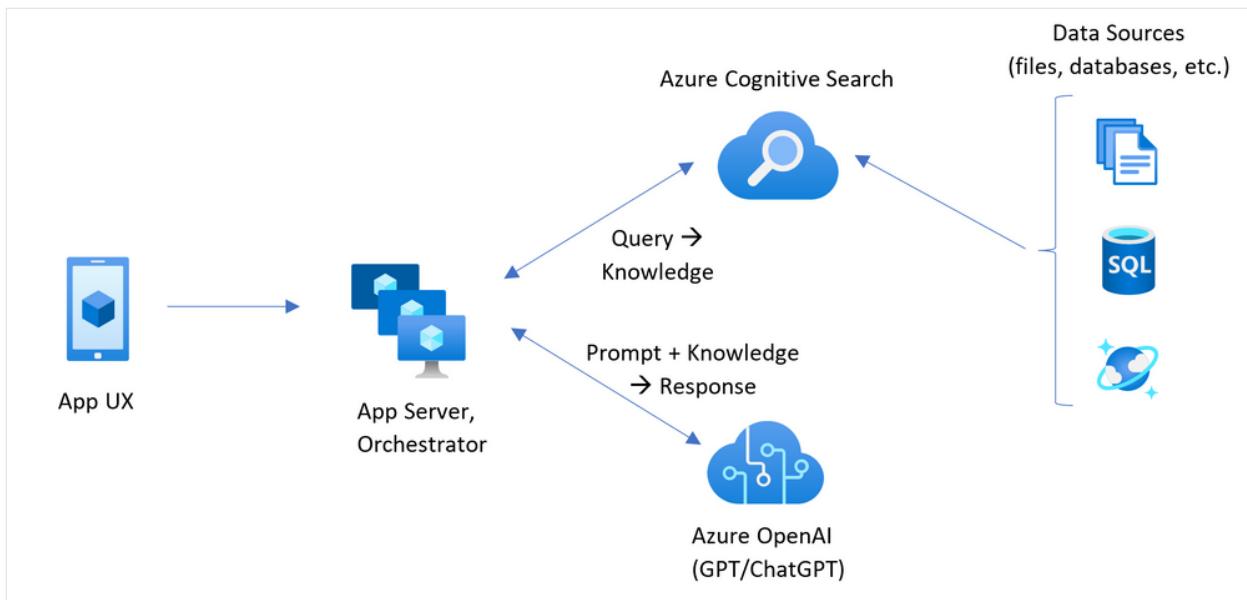
이 문서는 Azure OpenAI Service 및 Azure AI Search를 사용하여 채팅 앱을 빌드하는 방법을 보여 주는 문서 모음의 일부입니다.

컬렉션의 다른 문서는 다음과 같습니다.

- [.NET](#)
- [Java](#)
- [JavaScript](#)
- [JavaScript 프런트 엔드 + Python 백 엔드](#)

아키텍처 개요

다음 다이어그램은 채팅 앱의 간단한 아키텍처를 보여 줍니다.



아키텍처의 주요 구성 요소는 다음과 같습니다.

- 대화형 채팅 환경을 호스팅하는 웹 애플리케이션입니다.
- 사용자 고유의 데이터에서 답변을 가져오는 Azure AI Search 리소스입니다.
- 다음을 제공하는 Azure OpenAI Service:
 - 고유의 데이터에 대한 검색을 향상시키는 키워드입니다.
 - OpenAI 모델의 답변.
 - ada 모델의 포함

비용

이 아키텍처의 대부분의 리소스는 기본 또는 사용량 가격 책정 계층을 사용합니다. 사용량 가격 책정은 사용량을 기준으로 책정됩니다. 즉, 사용한 만큼만 비용을 지불하면 됩니다. 이 문서를 완료하려면 요금이 발생하지만 요금은 최소화됩니다. 문서를 완료하면 리소스를 삭제하여 요금 발생을 중지할 수 있습니다.

샘플 리포지토리의 [비용에 대해](#) 자세히 알아봅니다.

필수 조건

이 문서를 완료하는 데 필요한 모든 종속성을 갖춘 [개발 컨테이너](#) 환경을 사용할 수 있습니다. GitHub Codespaces(브라우저)에서 개발 컨테이너를 실행하거나 Visual Studio Code를 사용하여 로컬로 실행할 수 있습니다.

이 문서를 사용하려면 다음 필수 구성 요소가 필요합니다.

Codespaces(권장)

2. Azure 계정 권한 - Azure 계정에는 [사용자 액세스 관리자](#) 또는 [소유자](#)와 같은 Microsoft.Authorization/roleAssignments/write 권한이 있어야 합니다.
3. 원하는 Azure 구독의 Azure OpenAI에 대한 액세스 권한. 현재 이 서비스에 대한 액세스 권한은 애플리케이션에서만 부여됩니다. <https://aka.ms/oai/access>에서 양식을 작성하여 Azure OpenAI에 대한 액세스를 신청할 수 있습니다. 문제가 있는 경우 이 리포지토리에서 문제를 엽니다.
4. GitHub 계정

개방형 개발 환경

이 문서를 완료하려면 모든 종속성이 설치된 개발 환경으로 지금 시작합니다.

GitHub Codespaces(권장)

[GitHub Codespaces](#) 는 사용자 인터페이스로 [웹용 Visual Studio Code](#) 를 사용하여 GitHub에서 관리하는 개발 컨테이너를 실행합니다. 가장 간단한 개발 환경을 위해서는 GitHub Codespaces를 사용하여 이 문서를 완료하는 데 필요한 올바른 개발자 도구와 종속성을 미리 설치합니다.

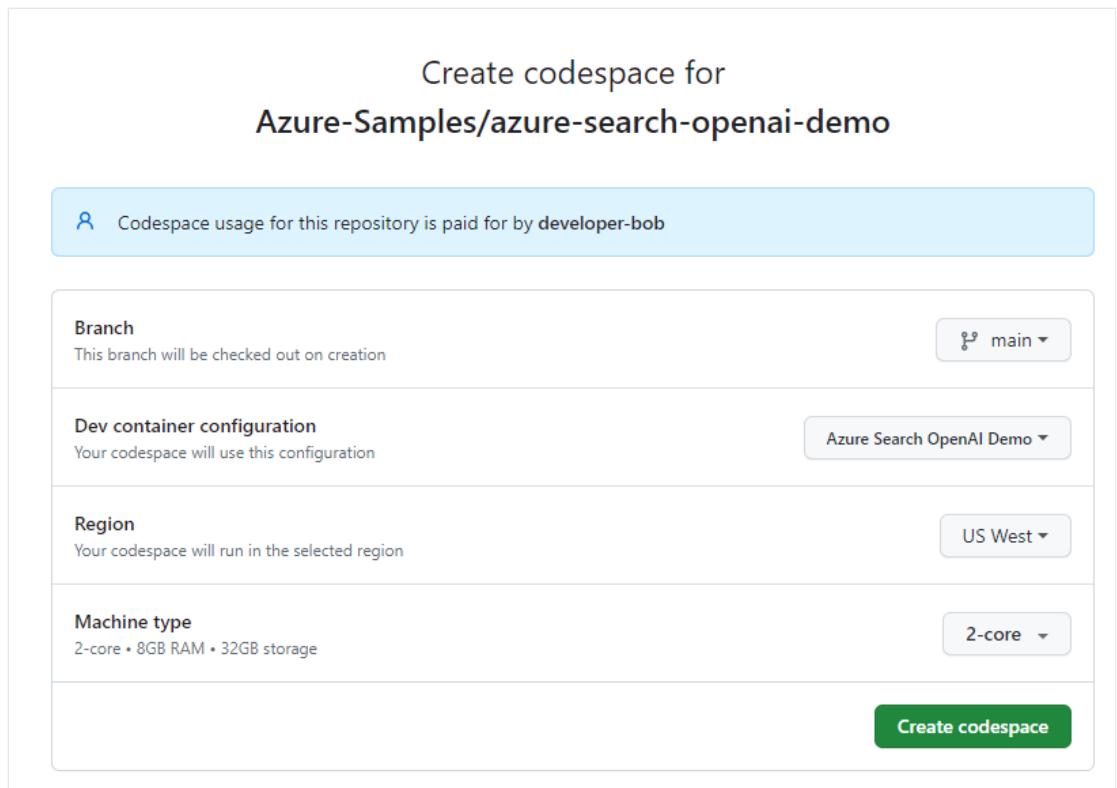
ⓘ 중요

모든 GitHub 계정은 2개의 코어 인스턴스를 사용하여 매월 최대 60시간 동안 Codespaces를 무료로 사용할 수 있습니다. 자세한 내용은 [GitHub Codespaces 월별 포함 스토리지 및 코어 시간](#) 을 참조하세요.

1. [Azure-Samples/azure-search-openai-demo](#) GitHub 리포지토리의 `main` 분기 에 새 GitHub Codespace를 만드는 프로세스를 시작합니다.
2. 개발 환경과 설명서를 동시에 사용하려면 다음 단추를 마우스 오른쪽 단추로 클릭하고 새 창에서 링크 열기를 선택합니다.

[GitHub Codespaces에서 이 프로젝트 열기](#)

3. [codespace 만들기](#) 페이지에서 codespace 구성 설정을 검토한 다음, 새 codespace 만들기를 선택합니다.



4. codespace가 생성될 때까지 기다립니다. 이 프로세스에는 몇 분 정도 걸릴 수 있습니다.

5. 화면 하단의 터미널에서 Azure 개발자 CLI를 사용하여 Azure에 로그인합니다.

```
Bash  
azd auth login
```

6. 터미널에서 코드를 복사한 다음 브라우저에 붙여넣습니다. 지침에 따라 Azure 계정으로 인증합니다.

7. 이 문서의 나머지 작업은 이 개발 컨테이너의 컨텍스트에서 수행됩니다.

배포 및 실행

샘플 리포지토리에는 Azure에 채팅 앱을 배포하는 데 필요한 모든 코드와 구성 파일이 포함되어 있습니다. 다음 단계에서는 샘플을 Azure에 배포하는 과정을 안내합니다.

Azure에 채팅 앱 배포

ⓘ 중요

이 섹션에서 만들어진 Azure 리소스는 주로 Azure AI 검색 리소스에서 즉각적인 비용이 발생합니다. 이러한 리소스는 명령이 완전히 실행되기 전에 중단하더라도 비용이 발생할 수 있습니다.

1. 다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 프로비전하고 소스 코드를 배포합니다.

```
Bash
```

```
azd up
```

2. 환경 이름을 입력하라는 메시지가 표시되면 짧고 소문자로 유지합니다. 예를 들어 `myenv`입니다. 리소스 그룹 이름의 일부로 사용됩니다.
3. 메시지가 표시되면 리소스를 만들 구독을 선택합니다.
4. 처음 위치를 선택하라는 메시지가 표시되면 가까운 위치를 선택합니다. 이 위치는 호스팅을 포함한 대부분의 리소스에 사용됩니다.
5. OpenAI 모델의 위치를 묻는 메시지가 표시되면 가까운 위치를 선택합니다. 첫 번째 위치와 동일한 위치를 사용할 수 있는 경우 해당 위치를 선택합니다.
6. 앱이 배포될 때까지 기다립니다. 배포를 완료하는 데 5~10분이 걸릴 수 있습니다.
7. 애플리케이션이 성공적으로 배포되면 터미널에 URL이 표시됩니다.
8. 브라우저에서 채팅 애플리케이션을 열려면 `(✓) Done: Deploying service webapp 0`라고 표시된 URL을 선택합니다.

GPT + Enterprise data | Sample Chat Ask a question Azure OpenAI + Cognitive Search

Clear chat Developer settings

Chat with your data

Ask anything or try an example

What is included in my Northwind Health Plus plan that is not in standard?

What happens in a performance review?

What does a Product Manager do?

Type a new question (e.g. does my plan cover annual eye exams?)

채팅 앱을 사용하여 PDF 파일에서 답변 가져오기

채팅 앱에는 [PDF 파일](#)의 직원 복리후생 정보가 미리 로드되어 있습니다. 채팅 앱을 이용하여 혜택에 대해 질문할 수 있습니다. 다음 단계에서는 채팅 앱을 사용하는 과정을 안내합니다.

1. 브라우저의 채팅 텍스트 상자에서 성능 검토에서 수행되는 작업을 선택하거나 입력합니다.

GPT + Enterprise data | Sample Chat Ask a question Azure OpenAI + Cognitive Search

Clear chat Developer settings

What happens in a performance review?

During a performance review, employees will have an opportunity to discuss their successes and challenges in the workplace ¹. The review will include constructive feedback and a written summary that includes a rating of the employee's performance, feedback, and goals and objectives for the upcoming year ¹. The review is a two-way dialogue between managers and employees, and employees are encouraged to be honest and open during the process ¹.

Citations: [1. employee_handbook-3.pdf](#)

Type a new question (e.g. does my plan cover annual eye exams?)

2. 답변에서 인용을 선택합니다.

The screenshot shows the Microsoft Q&A interface. At the top, it says "GPT + Enterprise data | Sample". In the center, there's a "Chat" button and an "Ask a question" input field. On the right, it says "Azure OpenAI + Cognitive Search" and has "Clear chat" and "Developer settings" buttons. Below the input field, a message box contains text about performance reviews and a citation: "Citations: 1. employee_handbook-3.pdf". A red box highlights the citation link. At the bottom, there's a search bar with "Type a new question (e.g. does my plan cover annual eye exams?)" and a search icon.

3. 오른쪽 창에서 템을 사용하여 답변이 생성된 방법을 이해합니다.

테이블 확장

탭	설명
사고 과정	이는 채팅의 상호 작용에 대한 스크립트입니다. 시스템 프롬프트(content)와 사용자 질문(content)을 볼 수 있습니다.
지원 내용	여기에는 사용자의 질문에 답변하기 위한 정보와 원본 재질이 포함됩니다. 원본 자료 인용 수는 개발자 설정나와 있습니다. 기본값은 3입니다.
인용	인용이 포함된 원래 페이지가 표시됩니다.

4. 완료되면 선택한 템을 다시 선택하여 창을 닫습니다.

채팅 앱 설정을 사용하여 응답 동작 변경

채팅의 지능은 OpenAI 모델과 해당 모델과 상호 작용하는 데 사용되는 설정에 의해 결정됩니다.

Configure answer generation

Override prompt template

Retrieve this many search results:

3

Exclude category

Use semantic ranker for retrieval

Use query-contextual summaries instead of whole documents

Suggest follow-up questions

Retrieval mode *

Vectors + Text (Hybrid)

Stream chat completion responses

Close

[] 테이블 확장

설정	설명
프롬프트 템플릿 재정의	이는 답변을 생성하는 데 사용되는 프롬프트입니다.
이 많은 검색 결과를 검색합니다.	답변을 생성하는 데 사용되는 검색 결과의 수입니다. 인용의 사고 과정 및 지원을 확인할 수 있습니다.
범주 제외	검색 결과에서 제외되는 문서 범주입니다.
검색을 위해 의미 순위매기기 사용	이는 기계 학습을 사용하여 검색 결과의 관련성을 높이는 Azure AI 검색 의 기능입니다.
전체 문서 대신 쿼리 컨텍스트 요	Use semantic ranker 와 Use query-contextual summaries 를 모두 선택하면 LLM은 순위가 가장 높은 문서의 모든 구절 대신 주요 구절에서 추출된 캡션을

설정	설명
약을 사용합니다.	사용합니다.
후속 질문 제안	채팅 앱에서 답변에 따라 후속 질문을 제안하도록 합니다.
검색 모드	벡터 + 텍스트 는 검색 결과가 문서의 텍스트와 문서의 포함을 기반으로 한다는 의미입니다. 벡터 는 검색 결과가 문서의 포함을 기반으로 한다는 의미입니다. 텍스트 는 검색 결과가 문서의 텍스트를 기반으로 한다는 의미입니다.
채팅 완료 응답 스트리밍	응답에 대한 전체 답변을 사용할 수 있게 될 때까지 기다리는 대신 응답을 스트리밍합니다.

다음 단계에서는 설정을 변경하는 과정을 안내합니다.

- 브라우저에서 개발자 설정 **탭**을 선택합니다.
- 후속 질문 제안** 확인란을 선택하고 동일한 질문을 다시 질문합니다.

What happens in a performance review?

채팅은 다음과 같은 제안된 후속 질문을 반환했습니다.

- What is the frequency of performance reviews?
- How can employees prepare for a performance review?
- Can employees dispute the feedback received during the performance review?

- 설정 탭에서 **검색에 의미 순위 매기기 사용**을 선택 취소합니다.
- 같은 질문을 다시 하시겠습니까?

What happens in a performance review?

- 답변의 차이점은 무엇인가요?

의미 체계 순위: Contoso Electronics의 성과 검토 중에 직원들은 직장에서의 성공과 과제에 대해 논의할 기회를 갖게 됩니다(1). 검토는 직원들이 자신의 역할을 개발하고 성장시키는 데 도움이 되는 긍정적이고 건설적인 피드백을 제공합니다(1). 직원은 성과 검토에 대한 서면 요약을 받게 되며, 여기에는 향후 연도(1)에 대한 성과, 피

드백 및 목표 및 목표에 대한 평가가 포함됩니다. 성과 검토는 관리자와 직원 간의 양방향 대화입니다(1).

의미 체계 순위가 없는 경우: Contoso Electronics에서 성과 검토를 수행하는 동안 직원들은 직장에서의 성공과 과제에 대해 논의할 수 있습니다. 긍정적이고 건설적인 피드백은 직원들이 자신의 역할을 개발하고 성장시키는 데 도움이 됩니다. 향후 연도의 성과, 피드백 및 목표 등급을 포함하여 성과 검토에 대한 서면 요약이 제공됩니다. 검토는 관리자와 직원 간의 양방향 대화입니다(1).

리소스 정리

Azure 리소스 정리

이 문서에서 만들어진 Azure 리소스는 Azure 구독에 요금이 청구됩니다. 앞으로 이러한 리소스가 필요하지 않을 것으로 예상되는 경우 추가 요금이 발생하지 않도록 삭제합니다.

다음 Azure 개발자 CLI 명령을 실행하여 Azure 리소스를 삭제하고 소스 코드를 제거합니다.

```
Bash  
azd down --purge --force
```

스위치는 다음을 제공합니다.

- `purge`: 삭제된 리소스는 즉시 제거됩니다. 이렇게 하면 Azure OpenAI TPM을 다시 사용할 수 있습니다.
- `force`: 삭제는 사용자 동의 없이도 일시적으로 발생합니다.

GitHub Codespaces 정리

GitHub Codespaces

GitHub Codespaces 환경을 삭제하면 계정에 대해 얻을 수 있는 코어당 무료 사용 권한을 최대화할 수 있습니다.

ⓘ 중요

GitHub 계정의 자격에 대한 자세한 내용은 [GitHub Codespaces 월별 포함된 스토리지 및 코어 시간](#)을 참조하세요.

- GitHub Codespaces 대시보드(<https://github.com/codespaces>)에 로그인합니다.
- Azure-Samples/azure-search-openai-demo GitHub 리포지토리에서 제공된 현재 실행 중인 codespace를 찾습니다.

The screenshot shows the GitHub Codespaces dashboard. In the top left, there's a sidebar with 'All' (1), 'Templates', and a repository section for 'Azure-Samples/azure-search-openai-demo'. The main area is titled 'Your codespaces' and contains three sections: 'Explore quick start templates' (Blank, React, Jupyter Notebook), 'Owned by developer-bob' (a list of codespaces), and 'See all' (link). A red box highlights the 'Owned by developer-bob' section, which lists a single codespace for the specified repository. The codespace details include the repository name, machine type (2-core • 8GB RAM • 32GB), status (Retrieving...), last used time (7 minutes ago), and a three-dot menu icon.

- codespace에 대한 상황에 맞는 메뉴를 열고 삭제를 선택합니다.

This screenshot shows the same GitHub Codespaces dashboard as the previous one, but with a context menu open over the specific codespace listed in the 'Owned by developer-bob' section. The menu options are: 'Open in ...', 'Rename', 'Export changes to a fork', 'Change machine type', 'Keep codespace', and 'Delete'. The 'Delete' option is highlighted with a red box.

도움말 보기

이 샘플 리포지토리는 문제 해결 정보[문제 해결 정보](#)를 제공합니다.

문제가 해결되지 않으면 리포지토리의 [문제](#)에 문제를 기록합니다.

다음 단계

- 엔터프라이즈 채팅 앱 GitHub 리포지토리 [↗](#)
- Azure OpenAI를 사용하여 채팅 앱 빌드 [↗](#) 모범 사례 솔루션 아키텍처
- Azure AI 검색을 사용한 생성 AI 앱의 액세스 제어 [↗](#)
- Azure API Management를 사용하여 엔터프라이즈급 OpenAI 솔루션 빌드 [↗](#)
- 하이브리드 검색 및 순위 지정 기능으로 탁월한 벡터 검색 성능 제공 [↗](#)

Azure AI 서비스 지원 및 도움말 옵션

아티클 • 2024. 02. 22.

이제 막 Azure AI 서비스의 기능을 살펴보기 시작하셨나요? 애플리케이션에 새로운 기능을 구현하는 경우를 가정해 볼 수 있습니다. 또는 서비스를 사용한 후 개선 방법에 대한 제안이 있나요? Azure AI 서비스에 대한 지원을 받고, 최신 상태를 유지하고, 피드백을 제공하고, 버그를 보고할 수 있는 옵션은 다음과 같습니다.

Azure 지원 요청 만들기

A

지금 클라우드를 시작하려는 개발자든 비즈니스에 중요한 전략적 애플리케이션을 배포하려는 대규모 조직이든 관계없이 가장 적합한 [Azure 지원 옵션 및 플랜 선택](#)의 범위를 탐색합니다. Azure 고객은 Azure Portal에서 지원 요청을 만들고 관리할 수 있습니다.

- [Azure Portal](#)
- [미국 정부의 Azure Portal](#)

Microsoft Q&A에 질문 게시

Microsoft 엔지니어, Azure MVP(가장 귀중한 전문가) 또는 전문가 커뮤니티의 기술 제품 관련 질문에 대한 빠르고 안정적인 답변을 얻으려면 Azure가 커뮤니티 지원을 위해 선호하는 대상인 [Microsoft Q&A](#)에 참여하세요.

검색을 사용하여 문제에 대한 답변을 찾을 수 없으면 Microsoft Q&A에 새 질문을 제출합니다. 질문을 할 때 다음 태그 중 하나를 사용합니다.

- [Azure AI 서비스](#)

비전

- [Azure AI Vision](#)
- [Custom Vision](#)
- [Face](#)
- [문서 인텔리전스](#)
- [Video Indexer](#)

언어

- [Immersive Reader](#)
- [언어 이해\(LUIS\)](#)

- QnA Maker
- 언어 서비스
- Translator

음성

- Speech Service

의사 결정

- Anomaly Detector
- Content Moderator
- Metrics Advisor
- Personalizer

Azure OpenAI

- Azure OpenAI

Stack Overflow에 질문을 게시합니다.



가장 큰 커뮤니티 개발자 에코시스템의 개발자 질문에 대한 답변을 보려면 Stack Overflow에서 질문하세요.

Stack Overflow에 새 질문을 제출하는 경우 질문을 만들 때 다음 태그 중 하나 이상을 사용하세요.

- Azure AI 서비스 ↗

비전

- Azure AI Vision ↗
- Custom Vision ↗
- Face ↗
- 문서 인텔리전스 ↗
- Video Indexer ↗

언어

- Immersive Reader ↗
- 언어 이해(LUIS) ↗
- QnA Maker ↗
- 언어 서비스 ↗

- Translator ↗

음성

- Speech Service ↗

의사 결정

- Anomaly Detector ↗
- Content Moderator ↗
- Metrics Advisor ↗
- Personalizer ↗

Azure OpenAI

- Azure OpenAI ↗

피드백 제출

새로운 기능을 요청하려면 <https://feedback.azure.com> 에 게시합니다. Azure AI 서비스 와 해당 API가 개발하는 애플리케이션에 더 잘 작동하도록 만들기 위한 아이디어를 공유하세요.

- Azure AI 서비스 ↗

비전

- Azure AI Vision ↗
- Custom Vision ↗
- Face ↗
- 문서 인텔리전스 ↗
- Video Indexer ↗

언어

- Immersive Reader ↗
- 언어 이해(LUIS) ↗
- QnA Maker ↗
- 언어 서비스 ↗
- Translator ↗

음성

- Speech Service ↗

의사 결정

- [Anomaly Detector ↗](#)
- [Content Moderator ↗](#)
- [Metrics Advisor ↗](#)
- [Personalizer ↗](#)

최신 소식 수신

새 릴리스의 기능이나 Azure 블로그의 뉴스에 대한 최신 정보를 얻으면 프로그래밍 오류, 서비스 버그 또는 아직 Azure AI 서비스에서 사용할 수 없는 기능 간의 차이점을 찾는데 도움이 될 수 있습니다.

- [Azure 업데이트 ↗](#)에서 제품 업데이트, 로드맵 및 공지 사항에 대해 자세히 알아봅니다.
- Azure AI 서비스에 대한 소식은 [Azure 블로그 ↗](#)에서 공유됩니다.
- Azure AI 서비스에 대한 [Reddit 대화에 참여하세요 ↗](#).

다음 단계

[Azure AI 서비스란?](#)