

## <9.4 뉴스 기사 요약해 음성으로 변환하기>

### 실습 변경 안내

9.4 뉴스 기사 요약해 음성으로 변환하기(323~337쪽)에 사용한 네이버 API 사이트가 개편되었습니다. 이에 따라 무료로 사용할 수 있는 Groq API를 사용하도록 본문 내용을 수정했습니다. 다음 페이지부터 참고해주세요.

감사합니다.

## 9.4 뉴스 기사 요약해 음성으로 변환하기

각 기업이 가진 정보를 제공하는 API가 있습니다. 예를 들어 카카오 API 사이트에서는 카카오톡, 카카오내비, 카카오스토리, KoGPT(한국어 버전 GPT)와 관련된 정보를 API로 제공받을 수 있습니다. 주요 기업의 API 사이트는 다음과 같습니다.

- **카카오 API:** <https://developers.kakao.com/product>
- **네이버 API:** <https://www.ncloud.com/product>
- **한국전자통신연구원(ETRI) API:** <https://aiopen.etri.re.kr/serviceList>
- **구글 API:** <https://cloud.google.com/apis?hl=ko>
- **유튜브 API:** <https://developers.google.com/youtube/v3?hl=ko>
- **X(트위터) API:** <https://developer.x.com/en/docs/x-api>
- **메타(페이스북) API:** [https://developers.facebook.com/docs?locale=ko\\_KR](https://developers.facebook.com/docs?locale=ko_KR)

이 절에서는 무료로 사용할 수 있는 Groq API를 이용해 뉴스 기사를 요약하고 음성으로 변환하는 프로그램을 만들어봅시다.

### 9.4.1 뉴스 기사 수집하기

중앙일보 사이트의 경제 코너(<https://www.joongang.co.kr/money>)에서 '경제 많이 본 기사'를 수집하겠습니다.

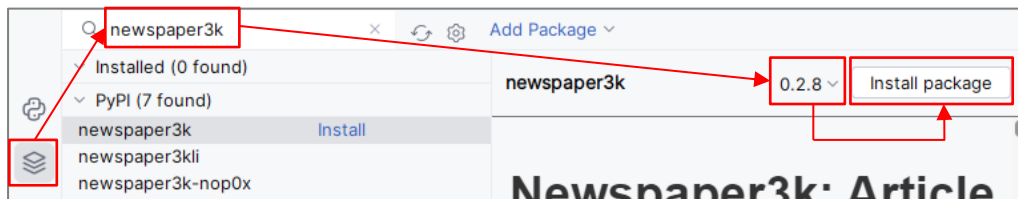
그림 9-29 중앙일보의 경제 많이 본 기사



이 실습에서는 `newspaper` 모듈을 사용합니다. `newspaper` 모듈은 뉴스 기사를 수집·분석하고 기사 본문의 텍스트와 이미지를 추출합니다. 또한 기사 본문의 기자 이름, 이메일 주소 등과 같은 불필요한 정보와 불규칙한 문단 구분, 여백 등을 보기 좋게 가공합니다.

`newspaper` 모듈은 `newspaper3k` 모듈을 설치해야 사용할 수 있습니다. 실행창의 Python Packages로 가서 `newspaper3k`(버전 0.2.8) 모듈을 설치합니다.

### 그림 9-30 newspaper3k 모듈 설치하기

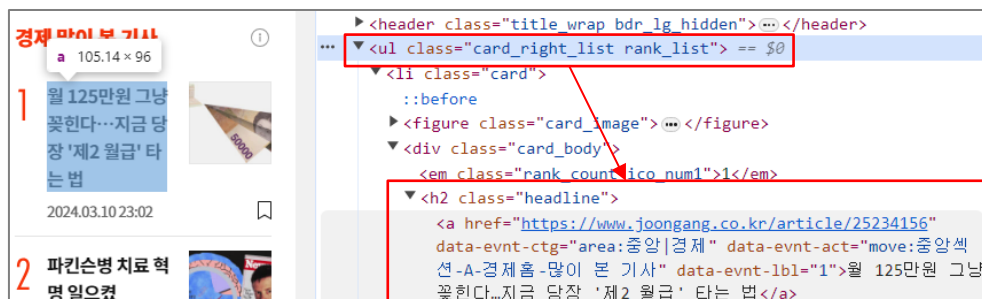


[pythonStudy] 폴더에 **ch09-뉴스기사요약** 파일을 만들고 다음 코드를 입력합니다.

❶ `bs4` 모듈로 기본적인 크롤링 코드를 작성합니다. `requests`와 `BeautifulSoup`에 빨간색 밑줄이 생기면 `[Alt]+[Enter]` 키를 눌러 `requests`와 `bs4` 모듈의 `BeautifulSoup` 명령을 추가합니다.

❷ '경제 많이 본 기사'의 제목에 해당하는 CSS 선택자를 확인합니다 ("`ul.card_right_list.rank_list h2.headline > a`"). 이 선택자를 이용해 웹 페이지에서 제목 요소를 모두 가져와 `title_list` 변수에 저장합니다. 이 선택자의 `href` 속성에는 기사 본문 링크가 들어 있으니 기사 본문 URL도 확보한 셈입니다.

### 그림 9-31 기사 제목의 CSS 선택자



- ③ 기사 제목을 띄우기 위해 `for` 반복문을 돌며 `title_list`에서 인덱스와 원소를 가져와 출력합니다.
- ④ 본문 내용을 수집할 기사 번호를 입력받아 `user_choice` 변수에 저장합니다.
- ⑤ 선택한 기사의 제목 요소를 가져와 `user_item` 변수에 저장합니다.
- ⑥ 제목 요소에서 내용을 가져와 `news_title` 변수에 저장하고, 제목 요소에서 `href` 속성값을 가져와 `news_link` 변수에 저장합니다.
- ⑦ `newspaper` 모듈의 `Article()`은 기사 본문을 수집하는 명령입니다. 첫 번째 인자에는 수집할 기사 본문의 URL을 넣고, 두 번째 인자로 기사 본문의 언어를 지정합니다. 여기서는 `news_link`에 한국어(ko)로 된 기사 본문을 수집해 `article` 변수에 저장합니다. `Article`에 빨간색 밑줄이 생기면 `[Alt]+[Enter]` 키를 눌러 `newspaper` 모듈의 `Article` 명령을 추가합니다.
- ⑧ `article.download()`로 기사 본문을 내려받고, `article.parse()`로 기사 본문을 구문 분석해 기사 제목, 내용, 작성자, 발행 날짜 등과 같은 정보를 추출합니다.
- ⑨ `article.text`로 기사 본문 텍스트를 문자열로 반환해 출력합니다.

<코드> ch09-뉴스기사요약.py

```
import requests ----- ❶ 모듈 자동 추가

from bs4 import BeautifulSoup --- ❶ 모듈 자동 추가

from newspaper import Article --- ❷ 모듈 자동 추가


code = requests.get("https://www.joongang.co.kr/money") ❶ 크롤링 코드 작성
soup = BeautifulSoup(code.text, "html.parser")          ❷ 기사 제목 가져오기
title_list = soup.select("ul.card_right_list.rank_list h2.headline > a")
# 기사 제목 띄우기                                     ❸ 기사 제목 출력
print("          [ 중앙일보 경제 뉴스 ]          ")

for index, title in enumerate(title_list):
    print(f"{index+1} - {title.text.strip()}")
print("-----")


# 수집할 기사 선택
user_choice = int(input("확인할 뉴스를 선택하세요 >> ")) - ❹ 기사 선택


# 선택한 기사의 본문 링크 가져오기
user_item = title_list[user_choice-1] --- ❺ 선택한 기사 제목 요소 가져오기
news_title= user_item.text ----- ❻ 기사 제목, 본문 링크 저장
news_link = user_item.attrs["href"]


# newspaper 모듈로 기사 본문 가공 및 출력
article = Article(news_link, language='ko') ----- ❷ 기사 본문 수집
article.download() ----- ❸ 기사 본문 다운로드 및 추출
article.parse()
news_content = article.text ----- ❹ 문자열로 본문 출력
print(news_content)
```

</코드>

코드를 실행하면 기사 제목에 이어 본문 내용을 확인하고 싶은 기사 번호를 선택 하라는 입력 상자가 나타납니다. 1~5번 중 원하는 숫자를 입력하면 본문을 수집해 가공한 결과가 출력됩니다.

<실행결과>

---

[ 중앙일보 경제 뉴스 ]

1 - “ 테슬라 · 팔란티어 · 엔비디아, 전고점 뚫을 놨은 딱 하나 ”

(중략)

-----  
확인할 뉴스를 선택하세요 >> 1

추천! 더중플 - 테슬라 · 엔비디아 · 팔란티어 주가 향방은? “ 시장을 떠나지 말라 ”

(중략)

---

</실행결과>

<TIP>

코드 실행 시 **ImportError: lxml.html.clean module is now a separate project lxml\_html\_clean**와 같은 에러가 뜬다면 Python Packages 창에서 [lxml-html-clean](#) 모듈을 설치하고 다시 실행하세요.

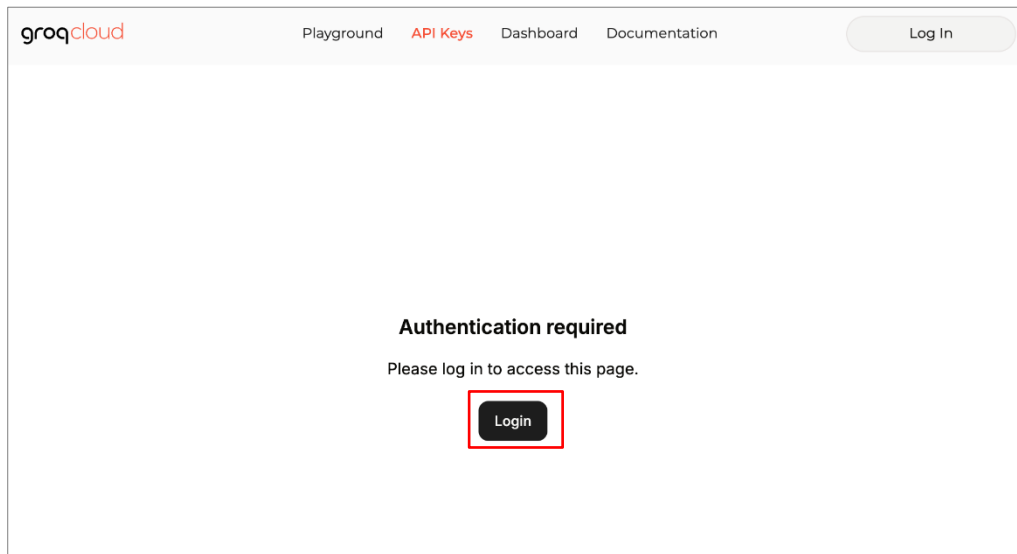
</TIP>

## 9.4.2 문서 요약 API 키 발급받기

수집한 기사 본문을 요약하기 위해 Groq의 LLM(Large Language Model, 대형 언어 모델) API를 사용하겠습니다. LLM은 사람처럼 자연스러운 문장을 생성하는 인공지능 모델로, Groq의 API는 빠르고 성능 좋은 최신 LLM을 무료로 제공해 문서 요약 작업에 유용하게 사용할 수 있습니다.

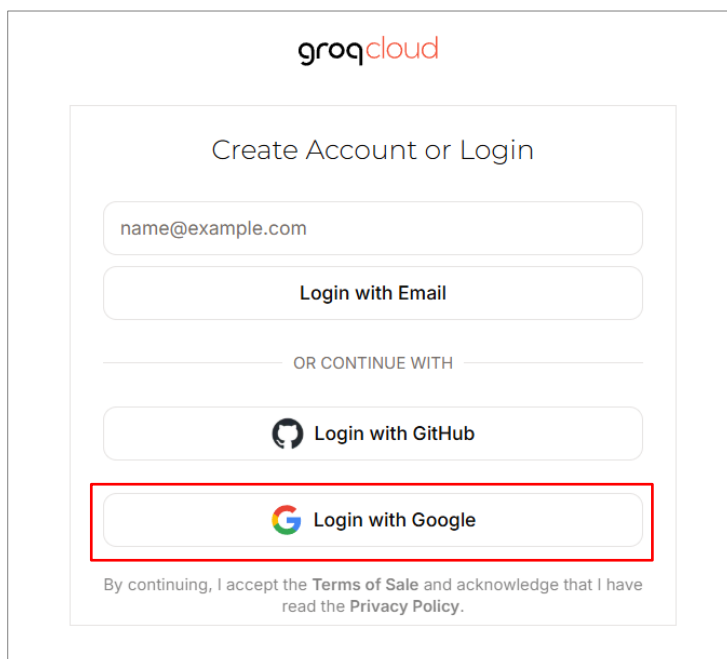
Groq API 사이트(<https://console.groq.com/keys>)에 접속하고 화면 중앙에 있는 [Login] 버튼을 클릭합니다.

그림 9-32 Groq API 사이트 화면



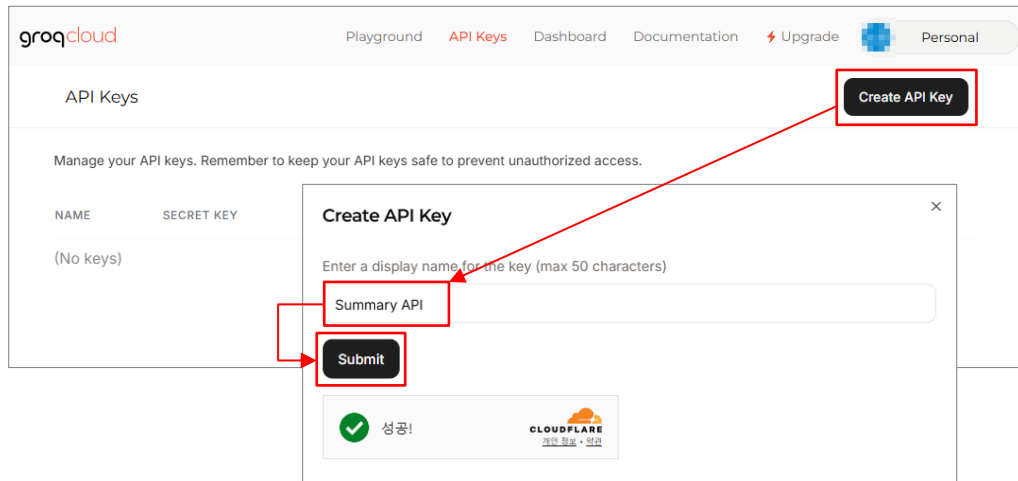
Google 계정으로 로그인하면 간편하게 로그인할 수 있습니다. 로그인 화면에서 [Login with Google]을 클릭해 로그인을 진행합니다.

그림 9-33 로그인 화면



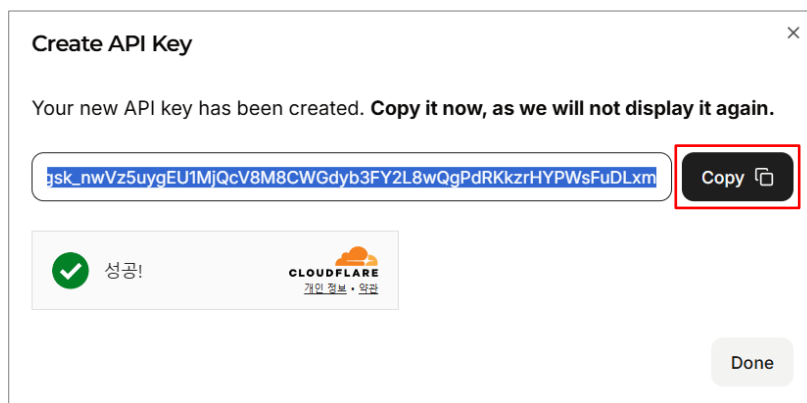
로그인이 완료되면 [Create API Key] 버튼을 클릭해 API Key를 발급받습니다. API 이름은 자유롭게 지어도 됩니다. 여기서는 **Summary API**라고 짓고 [Submit] 버튼을 클릭합니다.

그림 9-34 API Key 발급



API Key가 생성되면 [Copy] 버튼을 클릭해 API Key를 복사한 후 메모장에 붙여넣고 [Done] 버튼을 클릭합니다. API Key는 잊어버리지 않도록 메모장에 저장해두세요.

그림 9-35 API Key 복사

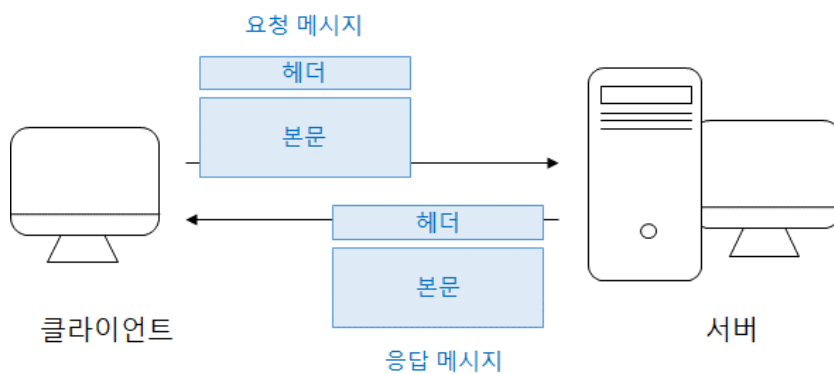




### 9.4.3 뉴스 기사 요약하기

발급받은 API를 사용해 뉴스 기사를 요약해봅시다. 그 전에 이해를 돕기 위해 **6.2.2절 웹 페이지의 동작 원리**에서 배웠던 내용을 떠올려보겠습니다. 클라이언트와 서버는 요청 메시지와 응답 메시지를 주고받으며 통신한다고 설명했는데, API 호출도 이와 마찬가지로입니다. 클라이언트에서 API 요청 메시지를 보내면 서버가 이를 받아 처리한 후 응답 메시지를 반환합니다. 요청 메시지와 응답 메시지는 각각 헤더와 본문으로 구성되며, 헤더에는 요청과 관련된 각종 정보가 담기고 본문에는 실제 전송되는 데이터가 담깁니다.

그림 9-36 API 호출 시 주고받는 메시지



다시 실습으로 돌아와, 문서 요약 API의 설명서를 살펴보기 위해 Groq API 설명서 (<https://console.groq.com/docs/quickstart>)에 접속합니다. 설명서를 보면 Groq API를 사용하려면 `groq`이라는 파이썬 모듈을 설치해야 된다고 합니다.

그림 9-37 Groq API 설명서

## Quickstart

Get up and running with the Groq API in a few minutes.

### Create an API Key

Please visit [here](#) to create an API Key.

### Set up your API Key (recommended)

Configure your API key as an environment variable. This approach streamlines your API usage by eliminating the need to include your API key in each request. Moreover, it enhances security by minimizing the risk of inadvertently including your API key in your codebase.

In your terminal of choice:

```
export GROQ_API_KEY=<your-api-key-here>
```

### Requesting your first chat completion

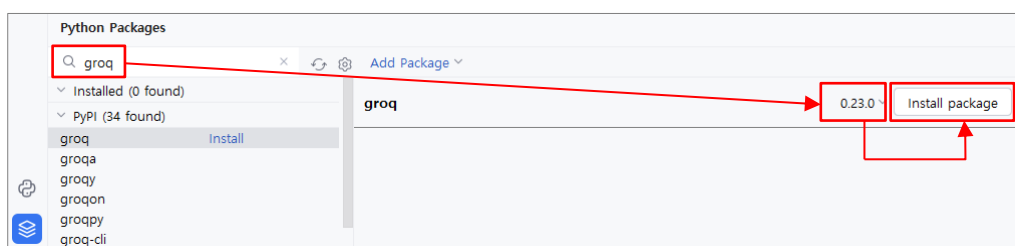
[curl](#) [JavaScript](#) [Python](#) [JSON](#)

#### Install the Groq Python library:

```
pip install groq
```

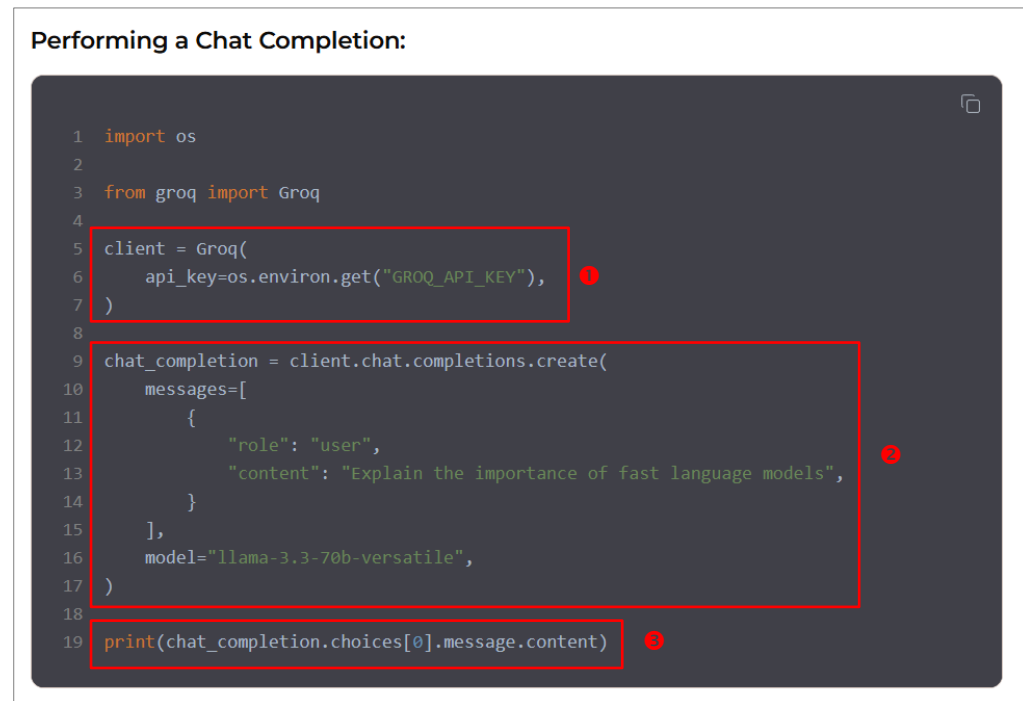
실행창의 Python Packages에서 [groq](#)(버전 0.23.0) 모듈을 설치합니다.

그림 9-38 groq 모듈 설치



Groq API 설명서에서 스크롤을 내려보면 **Performing a Chat Completion:** 코드가 있습니다. 이는 Groq API 호출 및 응답 코드로, 다음 그림에 표시한 것처럼 ❶번은 발급받은 API Key를 넣어주는 부분이고, ❷번은 API 서버에 보낼 요청 메시지를 작성하는 부분, ❸번은 API 서버에서 받은 응답 메시지를 출력하는 부분입니다.

그림 9-39 Groq API 호출 및 응답 코드



여기서 ❷번의 "content"와 model에 들어가는 값을 주목해주세요.

- **"content"**: LLM 모델에 요청할 문장을 입력합니다. ChatGPT에게 궁금한 것을 물어보거나 어떤 작업을 명령할 때 작성하는 프롬프트와 같습니다. 지금은 수집한 뉴스 기사를 요약해야 하므로 **"이 뉴스 기사를 요약해줘."**라고 작성하면 됩니다.
- **model**: 사용할 LLM 모델을 설정합니다. Groq에서는 DeepSeek, Llama, Qwen 등의 LLM 모델을 지원하며, 지원하는 모델 리스트는 Rate Limits(요금 제한) 페이지(<https://console.groq.com/docs/rate-limits>)에서 확인할 수 있습니다. 본 실습에서는 Llama 계열의 **llama3-70b-8192** 모델을 사용하겠습니다.

뉴스 기사를 요약하기 위한 Groq API 호출 및 응답 코드를 작성하면 다음과 같습니다.

<코드>

---

```
from groq import Groq

client = Groq(
    api_key="발급받은_API_Key_입력"
)

chat_completion = client.chat.completions.create(
    messages=[
        {
            "role": "user",
            "content": f"다음 뉴스 본문을 한국어로 3줄로 간략히 요약해줘. 각
                        요약 항목은 줄바꿈해. {뉴스_본문}" --- 프롬프트 입력
        }
    ],
    model="llama3-70b-8192", ----- 사용할 LLM 모델 설정
)

└ LLM 모델 응답 메시지 출력

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

---

</형식>

**ch09-뉴스기사요약** 파일로 돌아와 Groq API 호출 및 응답 코드를 추가합니다.

- 1 groq 모듈을 추가합니다.
- 2 발급받은 API Key를 저장하고, 서버에 보낼 요청 메시지를 작성합니다. 요청 메시지의 "content"에는 뉴스 본문을 3줄로 요약해달라는 문장과 함께 수집한 뉴스 본문인 `news_content`를 문자열 포매팅으로 문자열에 넣어줍니다.
- 3 구분선을 출력한 다음 LLM 모델의 응답 메시지를 출력합니다.

<코드> ch09-뉴스기사요약.py

---

```
from newspaper import Article
```

```
from groq import Groq ----- ❶
```

```
(중략)
```

```
print(news_content)
```

```
# Groq API 호출 및 응답받기 --- ❷
```

```
client = Groq(
```

```
    api_key="발급받은_API_Key_입력"
```

```
)
```

```
chat_completion = client.chat.completions.create(
```

```
    messages=[
```

```
        {
```

```
            "role": "user",
```

```
            "content": f"다음 뉴스 본문을 한국어로 3줄로 간략히 요약해줘. 각  
                        요약 항목은 줄바꿈해. {news_content}"
```

```
        }
```

```
    ],
```

```
    model="llama3-70b-8192",
```

```
)
```

```
# 기사 요약문 출력 ----- ❸
```

```
print("-----")
```

```
print("[기사 요약 결과]")
```

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

---

코드 실행 결과 3줄로 요약된 결과가 잘 출력됩니다.

#### 그림 9-40 문서 요약 결과

```
-----
[기사 요약 결과]
Here is a 3-sentence summary of the article in Korean:

현재 미국 기술주가는 하락세이지만, 테슬라·엔비디아·팔란티어 같은 강자들은 경쟁력과 성장성을 유지하고 있습니다.

테슬라의 주가는 완전자율주행 서비스의 성공 여부에 크게 좌우될 것이며, 엔비디아는 AI칩 성능이 우수해 주가 회복 요인이 많습

팔란티어는 시가총액이 300조원에 달하는 대기업이지만, 여전히 민간 부문 성장 잠재력이 높아 주가가 전고점을 넘어설 가능성이
```

### 9.4.4 요약 결과를 음성으로 변환하기

끝으로 요약한 문서를 음성으로 변환해봅시다. 이 작업을 하려면 `gtts`와 `playsound` 모듈이 필요하므로 코드 상단에 두 모듈을 불러오는 문장을 추가합니다.

- 1 `gtts`(Google Text-to-Speech)는 구글의 텍스트를 음성으로 변환하는 모듈로, 별도의 API 키를 발급받지 않고 무료로 사용할 수 있습니다. `from` 다음의 `gtts`에 빨간색 밑줄이 생기므로 실행창의 Python Packages로 가서 `gtts`(버전 2.5.1) 모듈을 설치합니다.
- 2 `playsound`는 오디오 파일을 재생하는 모듈입니다. `from` 다음의 `playsound`에 빨간색 밑줄이 생기므로 실행창의 Python Packages로 가서 `playsound`(버전 1.2.2) 모듈을 설치합니다.

<코드> ch09-뉴스기사요약.py

```
from groq import Groq

from gtts import gTTS ----- ❶ gtts 모듈 설치
from playsound import playsound --- ❷ playsound 모듈 설치
```

</코드>

<TIP>

`gtts` 모듈로 텍스트 음성 변환을 너무 자주 요청하거나 너무 긴 텍스트를 음성으로 변환하려고 하면 거절당할 수도 있습니다. 이때는 불가피하게 유료 버전을 사용해야 하는데, '구글 클라우드 tts api', '네이버 tts api', '카카오 tts api'를 검색하면 유료 tts API를 찾을 수 있습니다.

<TIP>

코드의 맨 아래에 다음 문장을 추가합니다.

- ❶ `gTTS()`는 `text`에 음성으로 변환할 텍스트를 받고 `lang`에 변환할 언어(ko, 한국어)를 지정해 텍스트를 음성으로 변환하는 명령입니다. 변환한 음성은 `comment_to_voice`에 저장합니다.
- ❷ 변환한 음성을 `comment_to_voice.save()` 명령을 활용해 `"news.mp3"` 파일로 저장합니다.
- ❸ `playsound()` 명령으로 `"news.mp3"` 파일을 재생합니다.

<코드> ch09-뉴스기사요약.py

---

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

```
# 텍스트-음성 변환

comment_to_voice = gTTS(text=chat_completion.choices[0].message.content,
                        lang="ko")           ↳ ❶ 요약 문서를 음성으로 변환

comment_to_voice.save("news.mp3") -- ❷ mp3 파일로 저장

playsound("news.mp3") ----- ❸ mp3 파일 재생
```

</코드>

코드를 실행하면 요약된 기사가 음성으로 재생됩니다. 이렇게 텍스트를 음성으로 변환하는 TTS API를 활용하면 매일 『뉴욕타임스』의 기사를 수집하고 음성으로 변환해 이를 영어 듣기 학습에 사용할 수 있습니다. 또한 매일 아침 컴퓨터를 켤 때 그날 수신된 이메일을 읽는 프로그램을 만들 수도 있습니다. 반대로 음성을 텍스트로 변환하는 음성 인식 API를 활용하면 회의 중 녹음한 음성 파일을 텍스트로 변환하고 요약해 회의록을 작성하는 프로그램을 만들 수 있습니다. 이처럼 텍스트-음성 변환 또는 음성-텍스트 변환 API는 일상과 업무에 유용하게 사용할 수 있는 도구입니다.

<TIP>

실행 시 **ModuleNotFoundError: No module named 'AppKit'**와 같은 에러가 뜬다면 Python Packages 창에서 [pyobjc](#) 모듈을 설치하고 다시 실행하세요.

</TIP>

## 1 분 퀴즈

3 앞에서 실습한 코드에 대한 설명 중 옳지 않은 것을 고르세요. ④

- ① [newspaper](#)는 뉴스 기사를 크롤링하고 기사 본문 내용을 가공하는 모듈이다.
- ② Groq API는 DeepSeek, Llama, Qwen 등 다양한 LLM 모델을 무료로 제공한다.
- ③ Groq API를 사용하면 문서 요약 이외에 뉴스 기사 내용 기반 보고서 작성, 새로운 말투의 문장 생성, 문서의 제목 생성 등의 작업을 명령할 수 있다.
- ④ 구글의 [gtts](#) 모듈은 별도의 API 키를 발급받아야 사용할 수 있다.