

## 4. OpenAI API



# OpenAI API Key

<https://platform.openai.com/>

The image shows a sequence of steps to create an OpenAI API key, overlaid on a browser window. The steps are numbered 1 through 5:

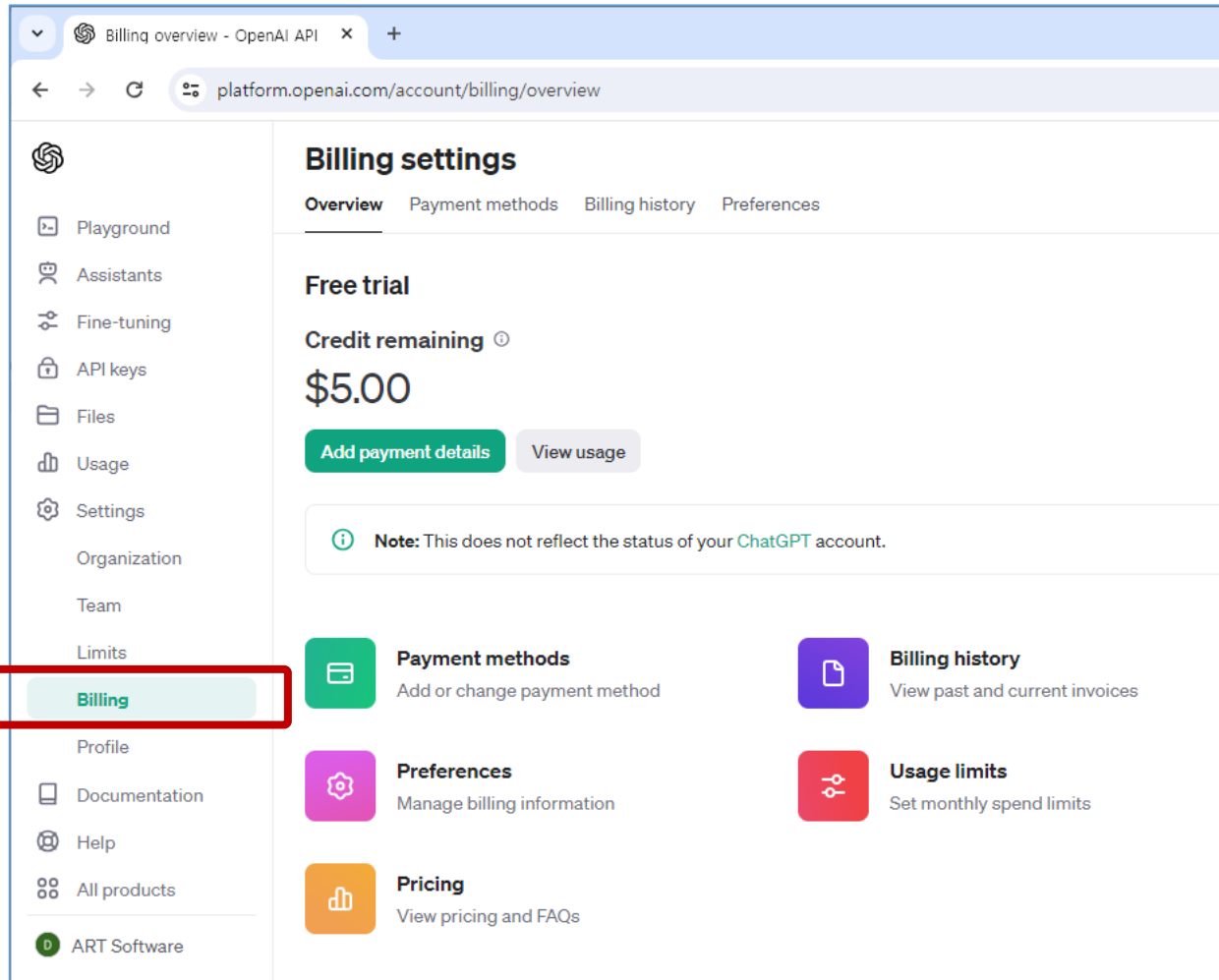
- 1**: Click the OpenAI logo in the top left corner of the browser window.
- 2**: Click the 'API keys' option in the left-hand navigation menu.
- 3**: Click the '+ Create new secret key' button at the bottom of the 'API keys' page.
- 4**: In the 'Create new secret key' dialog, enter a name (e.g., 'Test Key') in the 'Name' field.
- 5**: In the 'Save your key' dialog, click the 'Copy' button to copy the generated API key.

The background browser window shows the 'API keys' page with a table of existing keys. One key is visible with the name 'OpenAIAPIKey' and permissions set to 'All'.

NAME	Permissions	Created
OpenAIAPIKey	All	30일

# OpenAI API 무료사용

<https://platform.openai.com/account/billing/overview>



## Rate limits

MODEL	TOKEN LIMITS	REQUEST AND OTHER LIMITS
gpt-3.5-turbo : LLM	40,000 TPM	3 RPM 200 RPD
text-embedding-3-small	150,000 TPM	3 RPM 200 RPD
dall-e-3 : Text to Image		3 RPM 200 RPD
tts-1 : Text to Speech		3 RPM 200 RPD
whisper-1 : Automatic Speech Recognition		3 RPM 200 RPD

- TPM (tokens per minute)
- TPD (tokens per day)
- RPM (requests per minute)
- RPD (requests per day)
- IPM (images per minute)

- 1 token  $\sim$  4 chars in English
- 1 token  $\sim$   $\frac{3}{4}$  words
- 100 tokens  $\sim$  75 words

참고 :

<https://help.openai.com/en/articles/4936856-what-are-tokens-and-how-to-count-them>

# OpenAI API 유료사용

<https://platform.openai.com/account/billing/overview>

The screenshot shows the OpenAI API Billing overview page. The left sidebar contains navigation links: Playground, Assistants, Fine-tuning, API keys, Files, Usage, Settings, Organization, Team, Limits, Billing (highlighted with a red box), Profile, Documentation, Help, All products, and Personal. The main content area is titled "Billing settings" and includes tabs for Overview, Payment methods, Billing history, and Preferences. Under the Overview tab, it shows a "Free trial" status with "Credit remaining \$0.00" and buttons for "Add payment details" and "View usage". A note states: "Note: This does not reflect the status of your ChatGPT account." Below this, there are four cards: "Payment methods" (Add or change payment method), "Billing history" (View past and current invoices), "Preferences" (Manage billing information), and "Usage limits" (Set monthly spend limits). The "Usage limits" card is highlighted with a red box. A modal titled "Add payment details" is open on the right, showing fields for Card information (Card number, MM / YY, CVC), Name on card, and Billing address (Country, Address line 1, Address line 2, City, Postal code, State, county, province, or region). The modal includes "Cancel" and "Continue" buttons.

# 토큰(Token)

<https://platform.openai.com/tokenizer>

OpenAI Platform

platform.openai.com/tokenizer

Overview Documentation API reference Log in Sign up

GPT-3.5 & GPT-4 GPT-3 (Legacy)

Many words map to one token, but some don't: indivisible.

Unicode characters like emojis may be split into many tokens containing the underlying bytes: 🍌

Sequences of characters commonly found next to each other may be grouped together: 1234567890

Clear Show example

Tokens Characters

57 252

Many words map to one token, but some don't: indivisible.

Unicode characters like emojis may be split into many tokens containing the underlying bytes: 🍌🍌🍌🍌🍌

Sequences of characters commonly found next to each other may be grouped together: 1234567890

Text Token IDs

Tokens

57

Characters

252

[8607, 4339, 2472, 311, 832, 4037, 11, 719, 1063, 1541, 956, 25, 3687, 23936, 382, 35020, 5885, 1093, 100166, 1253, 387, 6859, 1139, 1690, 11460, 8649, 279, 16940, 5943, 25, 11410, 97, 248, 9468, 237, 122, 271, 1542, 45045, 315, 5885, 17037, 1766, 1828, 311, 1855, 1023, 1253, 387, 41141, 3871, 25, 220, 4513, 10961, 16474, 15]

Text

Token IDs



# 토큰 한도 (Token limit)

<https://platform.openai.com/docs/guides/text-generation/managing-tokens>

총 토큰 수는 API 호출에 영향을 줌

- 모델의 최대 한도 미만이어야 함, gpt-3.5-turbo 토큰 한도 4,096

총 토큰 수 : 입력 토큰 + 출력 토큰

- 토큰당 지불하는 API 호출 비용

- API 호출에 걸리는 시간

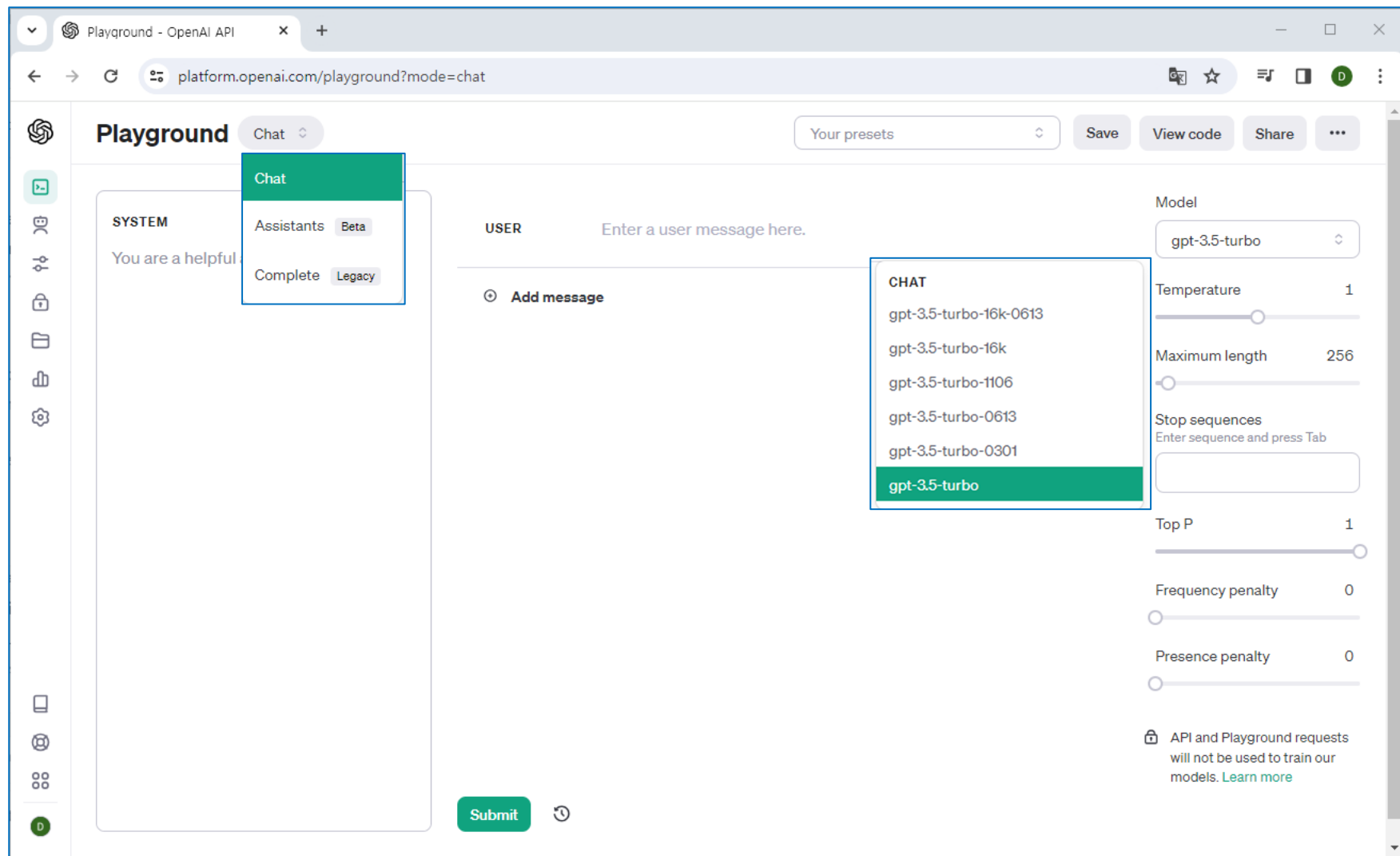
## Counting tokens for chat API calls

```
1 def num_tokens_from_messages(messages, model="gpt-3.5-turbo-0613"):
2     """Returns the number of tokens used by a list of messages."""
3     try:
4         encoding = tiktoken.encoding_for_model(model)
5     except KeyError:
6         encoding = tiktoken.get_encoding("cl100k_base")
7     if model == "gpt-3.5-turbo-0613": # note: future models may deviate from this
8         num_tokens = 0
9         for message in messages:
10             num_tokens += 4 # every message follows <im_start>{role/name}</im_start>
11             for key, value in message.items():
12                 num_tokens += len(encoding.encode(value))
13                 if key == "name": # if there's a name, the role is omitted
14                     num_tokens += -1 # role is always required and always 1 token
15             num_tokens += 2 # every reply is primed with <im_start>assistant
16         return num_tokens
17     else:
18         raise NotImplementedError(f"num_tokens_from_messages() is not presently implemented for model {model}. See https://github.com/openai/openai-python/blob/main/chatml.md for information about how messages are formatted and encoded. A future update will support other models' token counts.")
```

```
1 messages = [
2     {"role": "system", "content": "You are a helpful, pattern-following assistant."},
3     {"role": "system", "name": "example_user", "content": "New synergies will help drive growth."},
4     {"role": "system", "name": "example_assistant", "content": "Things work great! In fact, this new synergies will help drive growth."},
5     {"role": "system", "name": "example_user", "content": "Let's circle back to the beginning of the conversation."},
6     {"role": "system", "name": "example_assistant", "content": "Let's talk about the future."},
7     {"role": "user", "content": "This late pivot means we don't have time to do that."}
8 ]
9
10 model = "gpt-3.5-turbo-0613"
11
12 print(f"{num_tokens_from_messages(messages, model)} prompt tokens counted")
13 # Should show ~126 total_tokens
```

# 플레이그라운드

<https://platform.openai.com/playground>



- Temperature : 값이 낮을수록 가장 높은 확률의 다음 토큰을 선택하고, 높아지면 무작위성이 높아짐
- Max Length : 모델이 생성하는 토큰 최대 길이
- Stop Sequences : 모델의 토큰 생성을 중지하는 문자열
- Top P : 값이 높으면 모델이 가능성이 낮은 단어를 포함하여 더 다양한 출력을 얻을 수 있음
- Frequency Penalty : 해당 토큰이 나타난 횟수에 비례하여 페널티 적용
- Presence Penalty : 모든 반복 토큰에 동일한 페널티 적용(2번 나타나는 토큰과 10번 나타나는 토큰 모두 동일한 페널티)

※ Temperature 와 Top\_p, 그리고 Frequency Penalty와 Presence Penalty 동시 변경은 비권장함

# API 사용 방법

## Step 1: Setup Python

### ✓ Install Python

<https://www.python.org/downloads/>

### ✓ Setup a virtual environment (optional)

python -m venv myenv

Windows : myenv\Scripts\activate

Unix or Mac : source myenv/bin/activate

### ✓ Install the OpenAI Python library

pip install --upgrade openai

## Step 2: Setup your API key

Windows : setx OPENAI\_API\_KEY "your-api-key-here"

Unix or Mac : export OPENAI\_API\_KEY='your-api-key-here'

## Step 3: Sending your first API request

```
1 from openai import OpenAI
2 client = OpenAI()
3
4 completion = client.chat.completions.create(
5     model="gpt-3.5-turbo",
6     messages=[
7         {"role": "system", "content": "You are a poetic assistant, skilled in explaining"},
8         {"role": "user", "content": "Compose a poem that explains the concept of recursi"},
9     ]
10 )
11
12 print(completion.choices[0].message)
```



# 실습



openai\_api.ipynb

How\_to\_count\_tokens\_with\_tiktoken.ipynb  **GitHub**

chunking.ipynb

information\_retrieval.ipynb

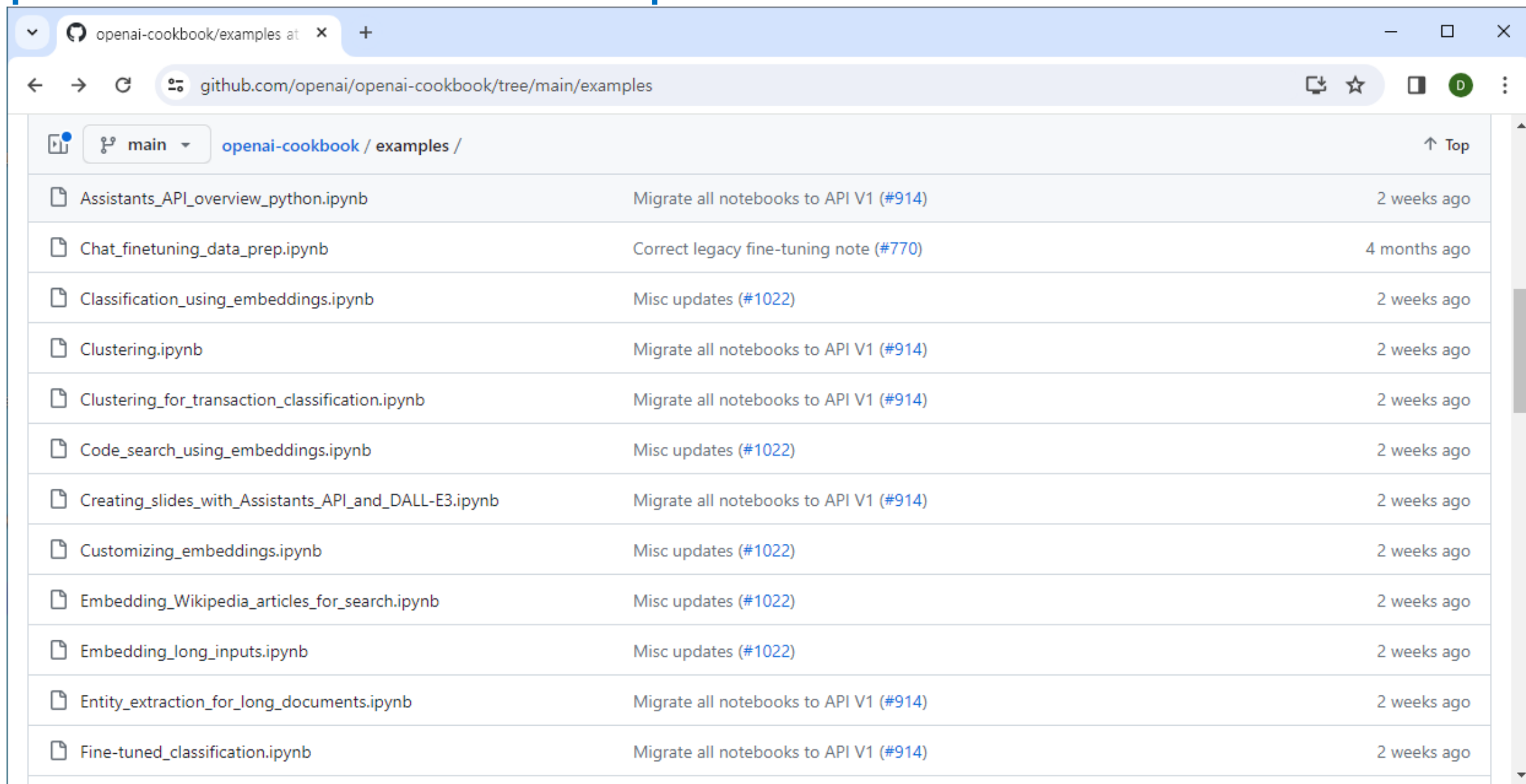
ReAct.ipynb

pe-lecture.ipynb

**colab**

# OpenAI Cookbook examples

<https://github.com/openai/openai-cookbook/tree/main/examples>

A screenshot of a web browser displaying the GitHub repository page for OpenAI Cookbook examples. The browser's address bar shows the URL 'github.com/openai/openai-cookbook/tree/main/examples'. The page header includes a 'main' branch selector and a 'Top' link. The main content is a table listing 12 Jupyter Notebook files, each with a description of the update and the time since the last update. The updates are categorized into 'Migrate all notebooks to API V1 (#914)' and 'Misc updates (#1022)'.

Assistants_API_overview_python.ipynb	Migrate all notebooks to API V1 (#914)	2 weeks ago
Chat_finetuning_data_prep.ipynb	Correct legacy fine-tuning note (#770)	4 months ago
Classification_using_embeddings.ipynb	Misc updates (#1022)	2 weeks ago
Clustering.ipynb	Migrate all notebooks to API V1 (#914)	2 weeks ago
Clustering_for_transaction_classification.ipynb	Migrate all notebooks to API V1 (#914)	2 weeks ago
Code_search_using_embeddings.ipynb	Misc updates (#1022)	2 weeks ago
Creating_slides_with_Assistants_API_and_DALL-E3.ipynb	Migrate all notebooks to API V1 (#914)	2 weeks ago
Customizing_embeddings.ipynb	Misc updates (#1022)	2 weeks ago
Embedding_Wikipedia_articles_for_search.ipynb	Misc updates (#1022)	2 weeks ago
Embedding_long_inputs.ipynb	Misc updates (#1022)	2 weeks ago
Entity_extraction_for_long_documents.ipynb	Migrate all notebooks to API V1 (#914)	2 weeks ago
Fine-tuned_classification.ipynb	Migrate all notebooks to API V1 (#914)	2 weeks ago

# Thank you