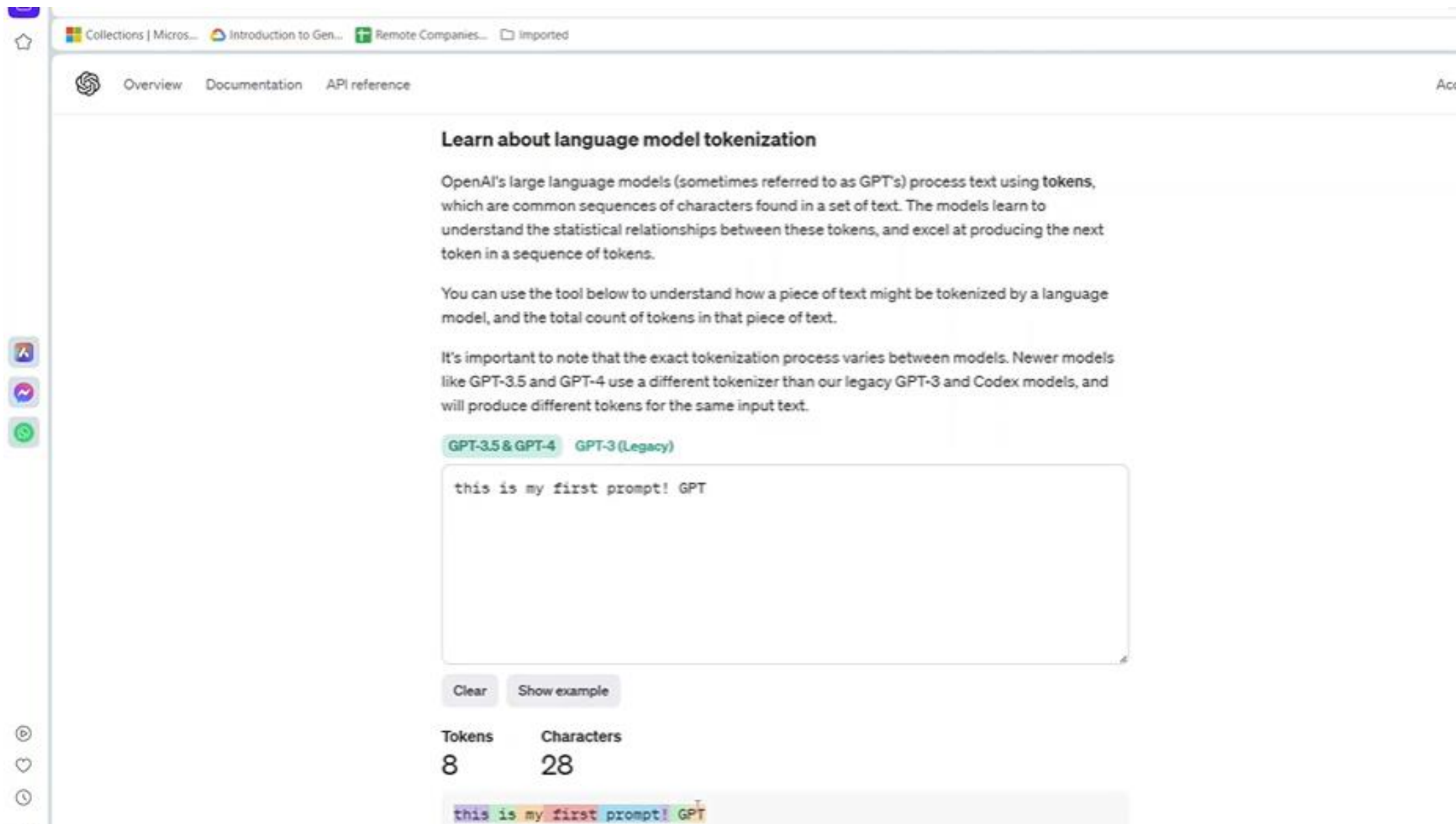


Connect to ChatGPT with an API

Please go to: colab.research.google.com

What is a Token in LLM



The screenshot shows the OpenAI documentation page titled "Learn about language model tokenization". The page explains that OpenAI's large language models (GPTs) process text using tokens, which are common sequences of characters. It provides a tool to visualize tokenization for the input text "this is my first prompt! GPT". The tool shows that the text is split into 8 tokens and 28 characters. The tokens are displayed as individual colored blocks: "this", "is", "my", "first", "prompt!", and "GPT".

Learn about language model tokenization

OpenAI's large language models (sometimes referred to as GPT's) process text using **tokens**, which are common sequences of characters found in a set of text. The models learn to understand the statistical relationships between these tokens, and excel at producing the next token in a sequence of tokens.

You can use the tool below to understand how a piece of text might be tokenized by a language model, and the total count of tokens in that piece of text.

It's important to note that the exact tokenization process varies between models. Newer models like GPT-3.5 and GPT-4 use a different tokenizer than our legacy GPT-3 and Codex models, and will produce different tokens for the same input text.

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

this is my first prompt! GPT

Clear Show example

Tokens	Characters
8	28

this is my first prompt! GPT

Chat GPT-API - Create Subscription

The screenshot shows the OpenAI Billing settings interface. On the left is a sidebar with navigation options: Playground, Assistants, Fine-tuning, API keys, Files, Usage, Settings, Organization, Team, Limits, Billing (highlighted), and Profile. Below these are links for 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. The 'Overview' tab is active, showing a 'Pay as you go' section with a 'Credit balance' of '\$4.93'. A warning box indicates 'Auto recharge is off' and provides a link to 'Enable auto recharge'. Below this are buttons for 'Add to credit balance' and 'Cancel plan'. At the bottom, there are five tiles for 'Payment methods', 'Billing history', 'Preferences', 'Usage limits', and 'Pricing'.

Collections | Micros... Introduction to Gen... Remote Companies... Imported

Billing settings

Overview Payment methods Billing history Preferences

Pay as you go

Credit balance ⓘ

\$4.93

ⓘ Auto recharge is off
When your credit balance reaches \$0, your API requests will stop working. Enable automatic recharge to automatically keep your credit balance topped up.
[Enable auto recharge](#)

Add to credit balance Cancel plan

Payment methods
Add or change payment method

Billing history
View past and current invoices

Preferences
Manage billing information

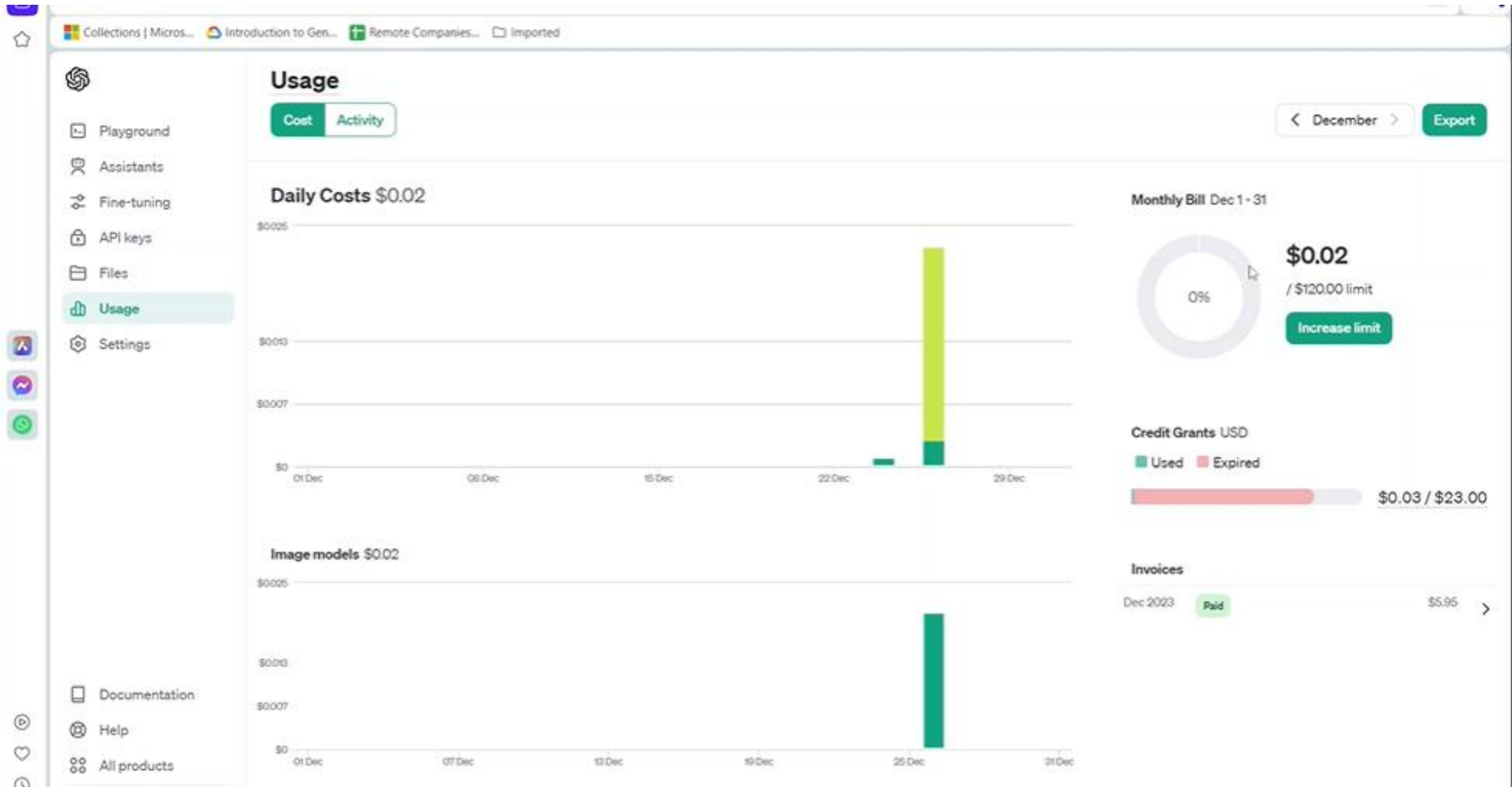
Usage limits
Set monthly spend limits

Pricing
View pricing and FAQs

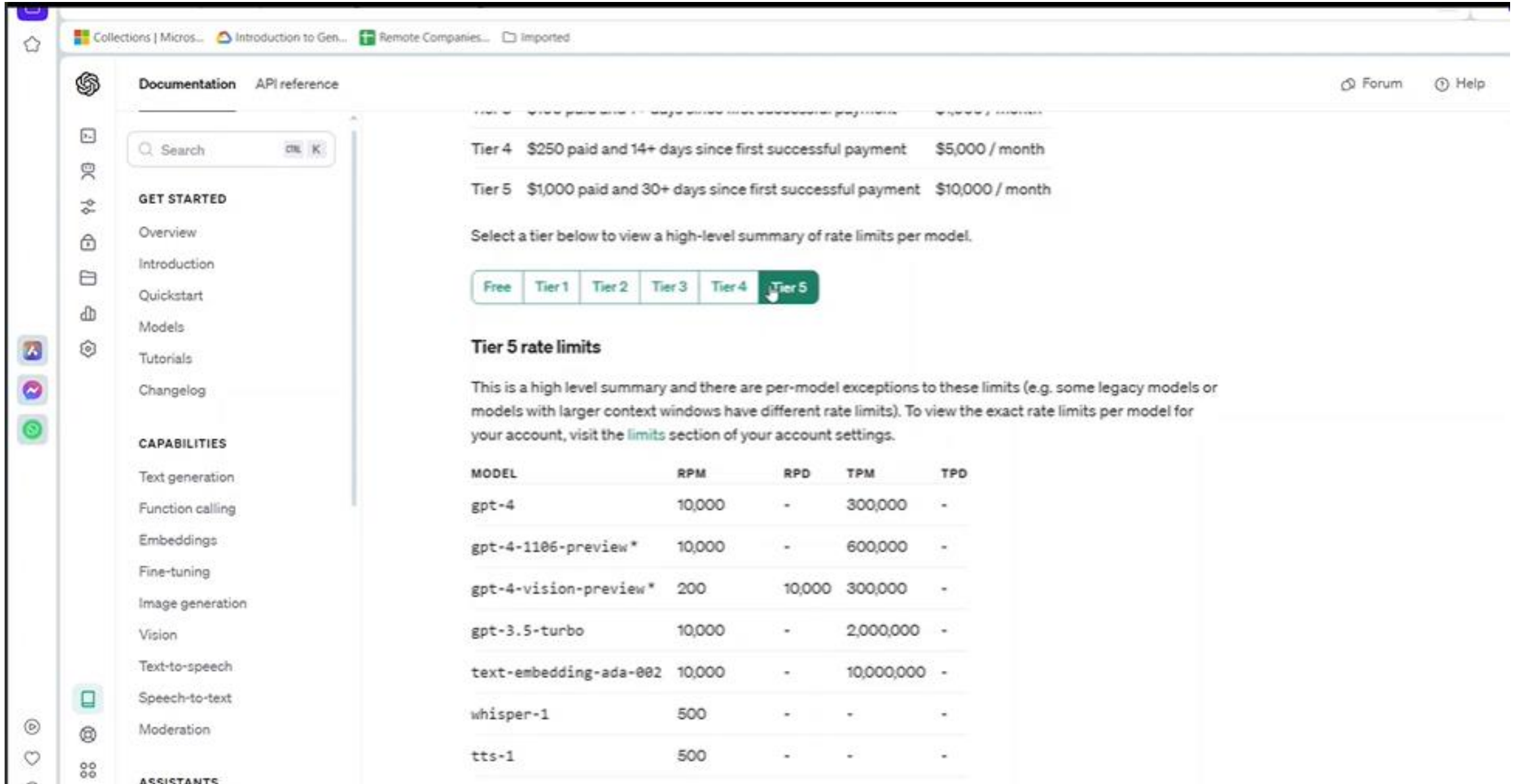
Playground Assistants Fine-tuning API keys Files Usage Settings Organization Team Limits **Billing** Profile

Documentation Help All products Personal

Chat GPT-API -Pricing



Chat GPT-API - Tiers and Limits



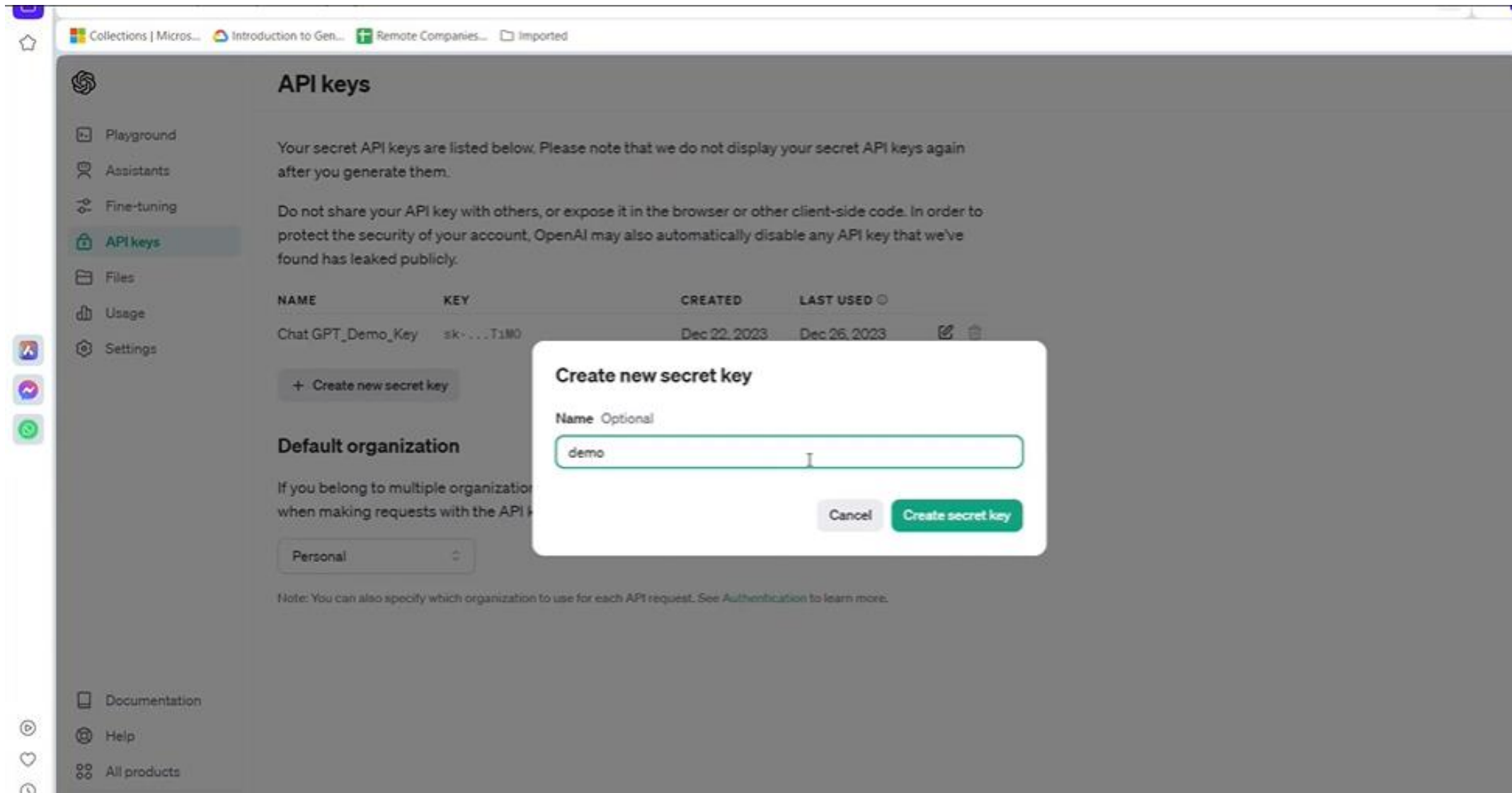
The screenshot shows the OpenAI API documentation page. On the left is a sidebar with navigation links under 'GET STARTED' (Overview, Introduction, Quickstart, Models, Tutorials, Changelog) and 'CAPABILITIES' (Text generation, Function calling, Embeddings, Fine-tuning, Image generation, Vision, Text-to-speech, Speech-to-text, Moderation). The main content area is titled 'Documentation' and 'API reference'. It shows a search bar and a list of tiers: Free, Tier 1, Tier 2, Tier 3, Tier 4, and Tier 5. Tier 5 is selected. Below the tiers, there is a section for 'Tier 5 rate limits' which includes a table of rate limits per model.

Tier 5 rate limits

This is a high level summary and there are per-model exceptions to these limits (e.g. some legacy models or models with larger context windows have different rate limits). To view the exact rate limits per model for your account, visit the [limits](#) section of your account settings.

MODEL	RPM	RPD	TPM	TPD
gpt-4	10,000	-	300,000	-
gpt-4-1106-preview*	10,000	-	600,000	-
gpt-4-vision-preview*	200	10,000	300,000	-
gpt-3.5-turbo	10,000	-	2,000,000	-
text-embedding-ada-002	10,000	-	10,000,000	-
whisper-1	500	-	-	-
tts-1	500	-	-	-

Chat GPT-API -GET the API Key



The screenshot shows the OpenAI API keys management interface. On the left is a sidebar with navigation links: Playground, Assistants, Fine-tuning, API keys (selected), Files, Usage, and Settings. The main content area is titled 'API keys' and contains a warning about not sharing keys and a table of existing keys. A modal dialog 'Create new secret key' is open in the foreground, featuring a text input for a name (currently 'demo') and two buttons: 'Cancel' and 'Create secret key'.

API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically disable any API key that we've found has leaked publicly.

NAME	KEY	CREATED	LAST USED
Chat GPT_Demo_Key	sk-...T1M0	Dec 22, 2023	Dec 26, 2023

+ Create new secret key

Create new secret key

Name Optional

demo

Cancel Create secret key

Default organization

If you belong to multiple organization when making requests with the API

Personal

Note: You can also specify which organization to use for each API request. See [Authentication](#) to learn more.

Chat GPT-API -APIs and Endpoints

The screenshot displays the OpenAI API documentation for the `POST https://api.openai.com/v1/moderations` endpoint. The left sidebar shows the navigation menu with sections like Documentation, API reference, and ENDPOINTS. The main content area describes the endpoint's purpose: "Classifies if text violates OpenAI's Content Policy". It details the request body parameters: `input` (string or array, required) and `model` (string, optional, defaulting to `text-moderation-latest`). The response section shows a JSON object with moderation results for various categories like harassment, self-harm, sexual/minors, hate/threatening, violence/graphic, self-harm/intent, self-harm/instructions, harassment/threatening, and violence. An example request and response are provided in code blocks on the right.

Endpoint: `POST https://api.openai.com/v1/moderations`

Description: Classifies if text violates OpenAI's Content Policy

Request body

input string or array **Required**
The input text to classify

model string Optional Defaults to `text-moderation-latest`
Two content moderation models are available: `text-moderation-stable` and `text-moderation-latest`.
The default is `text-moderation-latest` which will be automatically upgraded over time. This ensures you are always using our most accurate model. If you use `text-moderation-stable`, we will provide advanced notice before updating the model. Accuracy of `text-moderation-stable` may be slightly lower than for `text-moderation-latest`.

Returns

A `moderation` object.

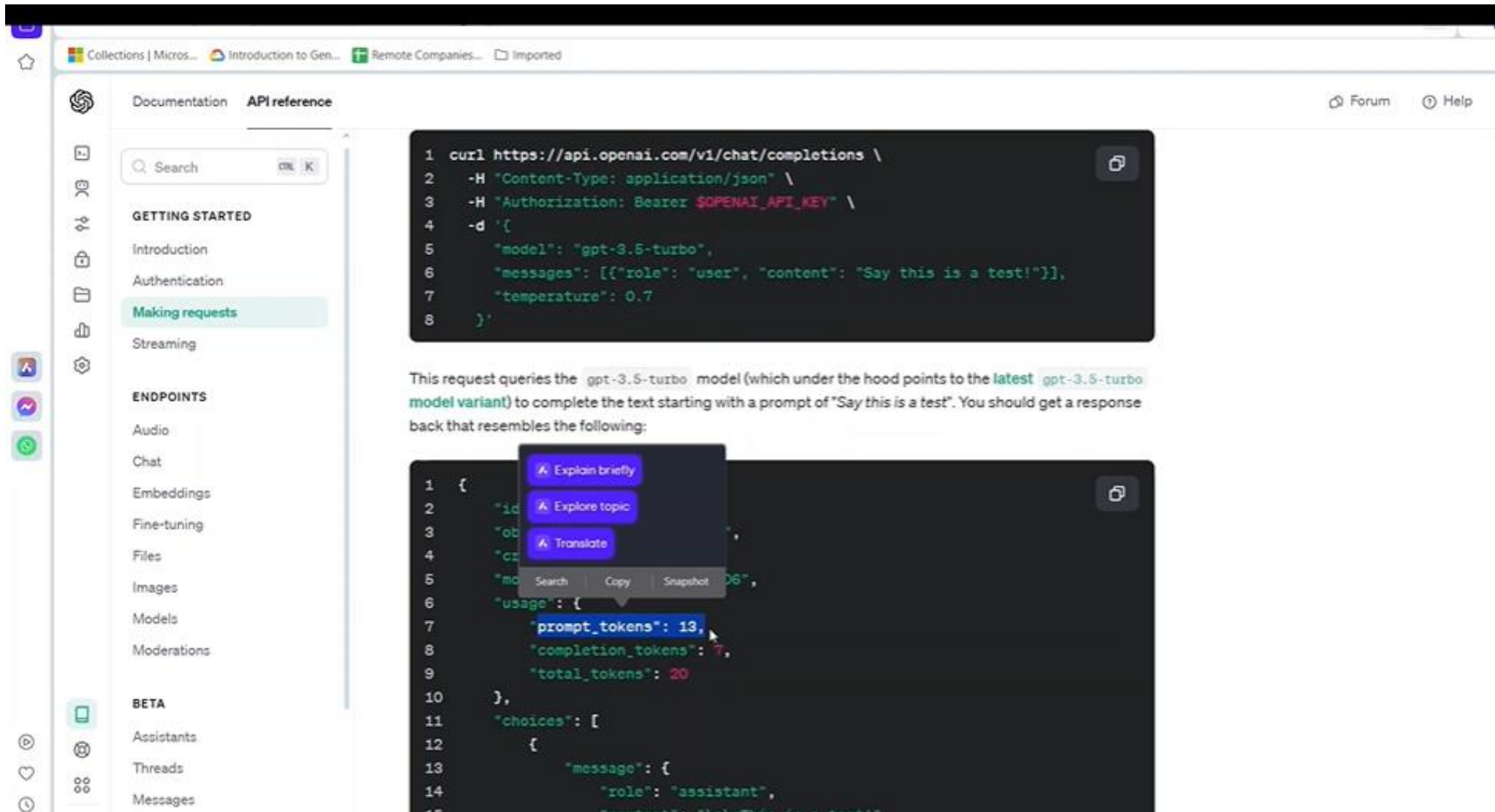
Example request

```
curl https://api.openai.com/v1/moderations \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-d '{
  "input": "I want to kill them."
}'
```

Response

```
{
  "moderations": [
    {
      "category_scores": {
        "sexual": 1.2262071e-06,
        "hate": 0.019366258,
        "harassment": 0.019366258,
        "self-harm": 0.019366258,
        "violence": 0.019366258,
        "hate/threatening": 0.019366258,
        "sexual/minors": 0.019366258,
        "self-harm/intent": 0.019366258,
        "self-harm/instructions": 0.019366258,
        "harassment/threatening": 0.019366258,
        "violence/graphic": 0.019366258
      },
      "category": "none",
      "flagged": false
    }
  ]
}
```


Chat GPT-API -Create the API Request



The screenshot displays the OpenAI API documentation page for the ChatGPT API. The left sidebar contains a navigation menu with sections like 'GETTING STARTED', 'ENDPOINTS', and 'BETA'. The main content area is titled 'API reference' and shows a 'curl' command for making a chat completion request. Below the command, a paragraph explains that the request queries the 'gpt-3.5-turbo' model to complete a prompt. A JSON response is shown below, with a tooltip overlaying the 'prompt_tokens' field, offering options to 'Explain briefly', 'Explore topic', and 'Translate'.

```
1 curl https://api.openai.com/v1/chat/completions \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY" \
4   -d '{
5     "model": "gpt-3.5-turbo",
6     "messages": [{"role": "user", "content": "Say this is a test!"}],
7     "temperature": 0.7
8   }'
```

This request queries the `gpt-3.5-turbo` model (which under the hood points to the latest `gpt-3.5-turbo` model variant) to complete the text starting with a prompt of "Say this is a test". You should get a response back that resembles the following:

```
1 {
2   "id": "chatcmpl-123",
3   "object": "chat.completion",
4   "created": 1677651200,
5   "model": "gpt-3.5-turbo",
6   "usage": {
7     "prompt_tokens": 13,
8     "completion_tokens": 7,
9     "total_tokens": 20
10  },
11  "choices": [
12    {
13      "message": {
14        "role": "assistant",
15        "content": "Hello! This is a test!"
```

Chat GPT-API - DALL-E API

The screenshot displays the OpenAI API documentation for the 'Create image' endpoint. The left sidebar contains navigation links for 'GETTING STARTED' (Introduction, Authentication, Making requests, Streaming) and 'ENDPOINTS' (Audio, Chat, Embeddings, Fine-tuning, Files, Images, Models, Moderations). The 'Images' section is expanded, showing 'Create image' as the selected endpoint.

Create image

POST `https://api.openai.com/v1/images/generations`

Creates an image given a prompt.

Request body

prompt string **Required**
A text description of the desired image(s). The maximum length is 1000 characters for `dall-e-2` and 4000 characters for `dall-e-3`.

model string **Optional** Defaults to `dall-e-2`
The model to use for image generation.

n integer or null **Optional** Defaults to 1
The number of images to generate. Must be between 1 and 10. For `dall-e-3`, only `n=1` is supported.

quality string **Optional** Defaults to `standard`
The quality of the image that will be generated. `hd` creates images with finer details and greater consistency across the image. This param is only supported for `dall-e-3`.

response_format string or null **Optional** Defaults to `url`
The format in which the generated images are returned. Must be one of `url` or `b64_json`.

size string or null **Optional** Defaults to `1024x1024`
The size of the generated images. Must be one of `256x256`, `512x512`, or `1024x1024` for `dall-e-2`. Must be one of `1024x1024`, `1792x1024`, or `1024x1792` for `dall-e-3`.

Example request

```
curl https://api.openai.com/v1/images/generations \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-d '{
  "model": "dall-e-3",
  "prompt": "A cute baby sea otter",
  "n": 1,
  "size": "1024x1024"
}'
```

Response

```
{
  "created": 1589478376,
  "data": [
    {
      "url": "https://..."
    },
    {
      "url": "https://..."
    }
  ]
}
```

Chat GPT- API - Moderation API

You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.

History New Import POST https://api.openai.com/ + ...

https://api.openai.com/v1/moderations Save

POST https://api.openai.com/v1/moderations Send

Params Authorization Headers (11) **Body** Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary JSON

1 `{ "input": "I am feeling good today" }`

Body Cookies (2) Headers (14) Test Results Status: 200 OK Time: 783 ms Size: 1.53 KB Save Response

Pretty Raw Preview Visualize JSON

```
1
2  {
3    "id": "modr-8cvLyGd80KchEgmArrNctIHHiqFn9",
4    "model": "text-moderation-006",
5    "results": [
6      {
7        "flagged": false,
8        "categories": {
9          "sexual": false,
10         "hate": false,
11         "harassment": false,
12         "self-harm": false,
13         "sexual/minors": false,
14         "hate/threatening": false,
15         "violence/graphic": false,
16         "self-harm/intent": false,
17         "self-harm/instructions": false,
```