

OpenAI Platform

Introduction

This API reference describes the RESTful, streaming, and realtime APIs you can use to interact with the OpenAI platform. REST APIs are usable via HTTP in any environment that supports HTTP requests. Language-specific SDKs are listed [on the libraries page](#).

Authentication

The OpenAI API uses API keys for authentication. Create, manage, and learn more about API keys in your [organization settings](#).

Remember that your API key is a secret! Do not share it with others or expose it in any client-side code (browsers, apps). API keys should be securely loaded from an environment variable or key management service on the server.

API keys should be provided via [HTTP Bearer authentication](#).

Authorization: Bearer OPENAI_API_KEY



If you belong to multiple organizations or access projects through a legacy user API key, pass a header to specify which organization and project to use for an API request:



```
1 curl https://api.openai.com/v1/models \
2   -H "Authorization: Bearer $OPENAI_API_KEY" \
3   -H "OpenAI-Organization: org-Vz3l6T05Uqd8sE2wzs06GJv1" \
4   -H "OpenAI-Project: $PROJECT_ID"
```

Usage from these API requests counts as usage for the specified organization and project. Organization IDs can be found on your [organization settings](#) page. Project IDs can be found on your [general settings](#) page by selecting the specific project.

Debugging requests

In addition to [error codes](#) returned from API responses, you can inspect HTTP response headers containing the unique ID of a particular API request or information about rate limiting applied to your requests. Below is an incomplete list of HTTP headers returned with API responses:

API meta information

`openai-organization` : The [organization](#) associated with the request

`openai-processing-ms` : Time taken processing your API request

`openai-version` : REST API version used for this request (currently `2020-10-01`)

`x-request-id` : Unique identifier for this API request (used in troubleshooting)

Rate limiting information

`x-ratelimit-limit-requests`

`x-ratelimit-limit-tokens`

x-ratelimit-remaining-requests
x-ratelimit-remaining-tokens
x-ratelimit-reset-requests
x-ratelimit-reset-tokens

OpenAI recommends logging request IDs in production deployments for more efficient troubleshooting with our [support team](#), should the need arise. Our [official SDKs](#) provide a property on top-level response objects containing the value of the `x-request-id` header.

Backward compatibility

OpenAI is committed to providing stability to API users by avoiding breaking changes in major API versions whenever reasonably possible. This includes:

- The REST API (currently `v1`)
- Our first-party [SDKs](#) (released SDKs adhere to [semantic versioning](#))
- [Model families](#) (like `gpt-4o` or `o4-mini`)

Model prompting behavior between snapshots is subject to change. Model outputs are by their nature variable, so expect changes in prompting and model behavior between snapshots. For example, if you moved from `gpt-4o-2024-05-13` to `gpt-4o-2024-08-06`, the same `system` or `user` messages could function differently between versions. The best way to ensure consistent prompting behavior and model output is to use pinned model versions, and to implement [evals](#) for your applications.

Backwards-compatible API changes:

Adding new resources (URLs) to the REST API and SDKs

Adding new optional API parameters

Adding new properties to JSON response objects or event data

Changing the order of properties in a JSON response object

Changing the length or format of opaque strings, like resource identifiers and UUIDs

Adding new event types (in either streaming or the Realtime API)

See the [changelog](#) for a list of backwards-compatible changes and rare breaking changes.

Responses

OpenAI's most advanced interface for generating model responses. Supports text and image inputs, and text outputs. Create stateful interactions with the model, using the output of previous responses as input.

Extend the model's capabilities with built-in tools for file search, web search, computer use, and more.

Allow the model access to external systems and data using function calling.

Related guides:

[Quickstart](#)

[Text inputs and outputs](#)

[Image inputs](#)

[Structured Outputs](#)

[Function calling](#)

[Conversation state](#)

Extend the models with tools

Create a model response

POST <https://api.openai.com/v1/responses>

Creates a model response. Provide [text](#) or [image](#) inputs to generate [text](#) or [JSON](#) outputs. Have the model call your own [custom code](#) or use built-in [tools](#) like [web search](#) or [file search](#) to use your own data as input for the model's response.

Request body

background boolean or null Optional Defaults to false

Whether to run the model response in the background. [Learn more](#).

include array or null Optional

Specify additional output data to include in the model response. Currently supported values are:

`code_interpreter_call.outputs` : Includes the outputs of python code execution in code interpreter tool call items.

`computer_call_output.output.image_url` : Include image urls from the computer call output.

`file_search_call.results` : Include the search results of the file search tool call.

[Text input](#)

[Image input](#)

[File input](#)

[Web search](#)

[Fi](#)

Example request

curl [curl](#)

```
1 curl https://api.openai.com/v1/responses \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY" \
4   -d '{
5     "model": "gpt-4.1",
6     "input": "Tell me a three sentence bedti
7   }'
```

Response

```
1 {
2   "id": "resp_67ccd2bed1ec8190b14f964abc054",
3   "object": "response",
4   "created_at": 1741476542,
5   "status": "completed",
6   "error": null,
7   "incomplete_details": null,
8   "instructions": null,
9   "max_output_tokens": null,
10  "model": "gpt-4.1-2025-04-14",
11  "output": [
12    {
```

`message.input_image.image_url` : Include image urls from the input message.

`message.output_text.logprobs` : Include logprobs with assistant messages.

`reasoning.encrypted_content` : Includes an encrypted version of reasoning tokens in reasoning item outputs. This enables reasoning items to be used in multi-turn conversations when using the Responses API statelessly (like when the `store` parameter is set to `false`, or when an organization is enrolled in the zero data retention program).

input string or array Optional

Text, image, or file inputs to the model, used to generate a response.

Learn more:

[Text inputs and outputs](#)

[Image inputs](#)

[File inputs](#)

[Conversation state](#)

[Function calling](#)

▼ Show possible types

instructions string or null Optional

A system (or developer) message inserted into the model's context.

When using along with `previous_response_id`, the instructions from a previous response will not be carried over to the next response. This makes it simple to swap out system (or developer) messages in new responses.

max_output_tokens integer or null Optional

```

13     "type": "message",
14     "id": "msg_67ccd2bf17f0819081ff3bb2cf
15     "status": "completed",
16     "role": "assistant",
17     "content": [
18       {
19         "type": "output_text",
20         "text": "In a peaceful grove bene
21         "annotations": []
22       }
23     ]
24   },
25   "parallel_tool_calls": true,
26   "previous_response_id": null,
27   "reasoning": {
28     "effort": null,
29     "summary": null
30   },
31   "store": true,
32   "temperature": 1.0,
33   "text": {
34     "format": {
35       "type": "text"
36     }
37   },
38   "tool_choice": "auto",
39   "tools": [],
40   "top_p": 1.0,
41   "truncation": "disabled",
42   "usage": {
43     "input_tokens": 36,
44     "input_tokens_details": {
45       "cached_tokens": 0
46     },
47   },

```

An upper bound for the number of tokens that can be generated for a response, including visible output tokens and [reasoning tokens](#).

max_tool_calls integer or null Optional

The maximum number of total calls to built-in tools that can be processed in a response. This maximum number applies across all built-in tool calls, not per individual tool. Any further attempts to call a tool by the model will be ignored.

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string Optional

Model ID used to generate the response, like `gpt-4o` or `o3`. OpenAI offers a wide range of models with different capabilities, performance characteristics, and price points. Refer to the [model guide](#) to browse and compare available models.

parallel_tool_calls boolean or null Optional Defaults to true

Whether to allow the model to run tool calls in parallel.

previous_response_id string or null Optional

The unique ID of the previous response to the model. Use this to create multi-turn conversations. Learn more about [conversation state](#).

prompt object or null Optional

Reference to a prompt template and its variables. [Learn more](#).

```

48 "output_tokens": 87,
49 "output_tokens_details": {
50   "reasoning_tokens": 0
51 },
52 "total_tokens": 123
53 },
54 "user": null,
55 "metadata": {}

```

[^ Hide properties](#)**id** string Required

The unique identifier of the prompt template to use.

variables map Optional

Optional map of values to substitute in for variables in your prompt. The substitution values can either be strings, or other Response input types like images or files.

version string or null Optional

Optional version of the prompt template.

prompt_cache_key string Optional

Used by OpenAI to cache responses for similar requests to optimize your cache hit rates. Replaces the `user` field. [Learn more](#).

reasoning object or null Optional

o-series models only

Configuration options for [reasoning models](#).

[^ Hide properties](#)**effort** string or null Optional Defaults to medium

Constrains effort on reasoning for [reasoning models](#). Currently supported values are `minimal`, `low`, `medium`, and `high`. Reducing reasoning effort can result in faster responses and fewer tokens used on reasoning in a response.

generate_summary Deprecated string or null Optional

Deprecated: use `summary` instead.

A summary of the reasoning performed by the model. This can be useful for debugging and understanding the model's reasoning process. One of `auto`, `concise`, or `detailed`.

summary string or null Optional

A summary of the reasoning performed by the model. This can be useful for debugging and understanding the model's reasoning process. One of `auto`, `concise`, or `detailed`.

safety_identifier string Optional

A stable identifier used to help detect users of your application that may be violating OpenAI's usage policies. The IDs should be a string that uniquely identifies each user. We recommend hashing their username or email address, in order to avoid sending us any identifying information. [Learn more](#).

service_tier string or null Optional Defaults to auto

Specifies the processing type used for serving the request.

If set to 'auto', then the request will be processed with the service tier configured in the Project settings. Unless otherwise configured, the Project will use 'default'.

If set to 'default', then the request will be processed with the standard pricing and performance for the selected model.

If set to '[flex](#)' or 'priority', then the request will be processed with the corresponding service tier. [Contact sales](#) to learn more about Priority processing.

When not set, the default behavior is 'auto'.

When the `service_tier` parameter is set, the response body will include the `service_tier` value based on the processing mode actually used to serve the request. This response value may be different from the value set in the parameter.

store boolean or null Optional Defaults to true

Whether to store the generated model response for later retrieval via API.

stream boolean or null Optional Defaults to false

If set to true, the model response data will be streamed to the client as it is generated using [server-sent events](#). See the [Streaming section below](#) for more information.

stream_options object or null Optional Defaults to null

Options for streaming responses. Only set this when you set `stream: true`.

^ Hide properties

include_obfuscation boolean Optional

When true, stream obfuscation will be enabled. Stream obfuscation adds random characters to an `obfuscation` field on streaming delta events to normalize payload sizes as a mitigation to certain side-channel attacks. These obfuscation fields are included by default, but add a small amount of overhead to the data stream. You can set `include_obfuscation` to false to optimize for bandwidth if you trust the network links between your application and the OpenAI API.

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic. We generally recommend altering this or `top_p` but not both.

text object Optional

Configuration options for a text response from the model. Can be plain text or structured JSON data. Learn more:

[Text inputs and outputs](#)

[Structured Outputs](#)

^ Hide properties

format object Optional

An object specifying the format that the model must output.

Configuring `{ "type": "json_schema" }` enables Structured Outputs, which ensures the model will match your supplied JSON schema. Learn more in the [Structured Outputs guide](#).

The default format is `{ "type": "text" }` with no additional options.

Not recommended for gpt-4o and newer models:

Setting to `{ "type": "json_object" }` enables the older JSON mode, which ensures the message the model generates is valid JSON. Using `json_schema` is preferred for models that support it.

▼ Show possible types

verbosity string or null Optional Defaults to medium

Constrains the verbosity of the model's response. Lower values will result in more concise responses, while higher values will result in more verbose responses.

Currently supported values are `low`, `medium`, and `high`.

tool_choice string or object Optional

How the model should select which tool (or tools) to use when generating a response.

See the `tools` parameter to see how to specify which tools the model can call.

^ Hide possible types

Tool choice mode string

Controls which (if any) tool is called by the model.

`none` means the model will not call any tool and instead generates a message.

`auto` means the model can pick between generating a message or calling one or more tools.

`required` means the model must call one or more tools.

Allowed tools object

Constrains the tools available to the model to a pre-defined set.

[^ Hide properties](#)

mode string Required

Constrains the tools available to the model to a pre-defined set.

`auto` allows the model to pick from among the allowed tools and generate a message.

`required` requires the model to call one or more of the allowed tools.

tools array Required

A list of tool definitions that the model should be allowed to call.

For the Responses API, the list of tool definitions might look like:

```
1 [  
2   { "type": "function", "name": "get_weather" },  
3   { "type": "mcp", "server_label": "deepwiki" },  
4   { "type": "image_generation" }  
5 ]
```



[^ Hide properties](#)

type string Required

Allowed tool configuration type. Always `allowed_tools`.

Hosted tool object

Indicates that the model should use a built-in tool to generate a response.

[Learn more about built-in tools.](#)

^ Hide properties

type string Required

The type of hosted tool the model should use. Learn more about [built-in tools](#).

Allowed values are:

`file_search`

`web_search_preview`

`computer_use_preview`

`code_interpreter`

`image_generation`

Function tool object

Use this option to force the model to call a specific function.

^ Hide properties

name string Required

The name of the function to call.

type string Required

For function calling, the type is always `function`.

MCP tool object

Use this option to force the model to call a specific tool on a remote MCP server.

^ Hide properties

server_label string Required

The label of the MCP server to use.

type string Required

For MCP tools, the type is always `mcp`.

name string or null Optional

The name of the tool to call on the server.

Custom tool object

Use this option to force the model to call a specific custom tool.

^ Hide properties

name string Required

The name of the custom tool to call.

type string Required

For custom tool calling, the type is always `custom`.

tools array Optional

An array of tools the model may call while generating a response. You can specify which tool to use by setting the `tool_choice` parameter.

The two categories of tools you can provide the model are:

Built-in tools: Tools that are provided by OpenAI that extend the model's capabilities, like [web search](#) or [file search](#). Learn more about [built-in tools](#).

Function calls (custom tools): Functions that are defined by you, enabling the model to call your own code with strongly typed arguments and outputs. Learn

more about [function calling](#). You can also use custom tools to call your own code.

^ Hide possible types

Function object

Defines a function in your own code the model can choose to call. Learn more about [function calling](#).

^ Hide properties

name string Required

The name of the function to call.

parameters object Required

A JSON schema object describing the parameters of the function.

strict boolean Required

Whether to enforce strict parameter validation. Default `true`.

type string Required

The type of the function tool. Always `function`.

description string Optional

A description of the function. Used by the model to determine whether or not to call the function.

File search object

A tool that searches for relevant content from uploaded files. Learn more about the [file search tool](#).

^ Hide properties

type string Required

The type of the file search tool. Always `file_search`.

vector_store_ids array Required

The IDs of the vector stores to search.

filters object Optional

A filter to apply.

▼ Show possible types

max_num_results integer Optional

The maximum number of results to return. This number should be between 1 and 50 inclusive.

ranking_options object Optional

Ranking options for search.

▼ Show properties

Web search preview object

This tool searches the web for relevant results to use in a response. Learn more about the [web search tool](#).

^ Hide properties

type string Required

The type of the web search tool. One of `web_search_preview` or `web_search_preview_2025_03_11`.

search_context_size string Optional

High level guidance for the amount of context window space to use for the search. One of `low`, `medium`, or `high`. `medium` is the default.

user_location object Optional

The user's location.

▼ Show properties

Computer use preview object

A tool that controls a virtual computer. Learn more about the [computer tool](#).

^ Hide properties

display_height integer Required

The height of the computer display.

display_width integer Required

The width of the computer display.

environment string Required

The type of computer environment to control.

type string Required

The type of the computer use tool. Always `computer_use_preview`.

MCP tool object

Give the model access to additional tools via remote Model Context Protocol (MCP) servers. [Learn more about MCP](#).

^ Hide properties

server_label string Required

A label for this MCP server, used to identify it in tool calls.

server_url string Required

The URL for the MCP server.

type string Required

The type of the MCP tool. Always `mcp`.

allowed_tools array or object Optional

List of allowed tool names or a filter object.

[^ Hide possible types](#)

MCP allowed tools array

A string array of allowed tool names

MCP allowed tools filter object

A filter object to specify which tools are allowed.

[^ Hide properties](#)

tool_names array Optional

List of allowed tool names.

headers object or null Optional

Optional HTTP headers to send to the MCP server. Use for authentication or other purposes.

require_approval object or string Optional Defaults to always

Specify which of the MCP server's tools require approval.

[^ Hide possible types](#)

MCP tool approval filter object

[^ Hide properties](#)

always object Optional

A list of tools that always require approval.

▼ Show properties

never object Optional

A list of tools that never require approval.

▼ Show properties

MCP tool approval setting string

Specify a single approval policy for all tools. One of `always` or `never`.

When set to `always`, all tools will require approval. When set to `never`, all tools will not require approval.

server_description string Optional

Optional description of the MCP server, used to provide more context.

Code interpreter object

A tool that runs Python code to help generate a response to a prompt.

^ Hide properties

container string or object Required

The code interpreter container. Can be a container ID or an object that specifies uploaded file IDs to make available to your code.

▼ Show possible types

type string Required

The type of the code interpreter tool. Always `code_interpreter`.

Image generation tool object

A tool that generates images using a model like `gpt-image-1`.

▼ Show properties

Local shell tool object

A tool that allows the model to execute shell commands in a local environment.

▼ Show properties

Custom tool object

A custom tool that processes input using a specified format. Learn more about [custom tools](#).

^ Hide properties

name string Required

The name of the custom tool, used to identify it in tool calls.

type string Required

The type of the custom tool. Always `custom`.

description string Optional

Optional description of the custom tool, used to provide more context.

format object Optional

The input format for the custom tool. Default is unconstrained text.

▼ Show possible types

top_logprobs integer or null Optional

An integer between 0 and 20 specifying the number of most likely tokens to return at each token position, each with an associated log probability.

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or `temperature` but not both.

truncation string or null Optional Defaults to disabled

The truncation strategy to use for the model response.

`auto` : If the context of this response and previous ones exceeds the model's context window size, the model will truncate the response to fit the context window by dropping input items in the middle of the conversation.

`disabled` (default): If a model response will exceed the context window size for a model, the request will fail with a 400 error.

user Deprecated string Optional

This field is being replaced by `safety_identifier` and `prompt_cache_key`. Use `prompt_cache_key` instead to maintain caching optimizations. A stable identifier for your end-users. Used to boost cache hit rates by better bucketing similar requests and to help OpenAI detect and prevent abuse. [Learn more](#).

Returns

Returns a [Response](#) object.

Get a model response

```
GET https://api.openai.com/v1/responses/{response_id}
```

Retrieves a model response with the given ID.

Path parameters

response_id string Required

The ID of the response to retrieve.

Query parameters

include array Optional

Additional fields to include in the response. See the `include` parameter for Response creation above for more information.

include_obfuscation boolean Optional

When true, stream obfuscation will be enabled. Stream obfuscation adds random characters to an `obfuscation` field on streaming delta events to normalize payload sizes as a mitigation to certain side-channel attacks. These obfuscation fields are included by default, but add a small amount of overhead to the data stream. You can set `include_obfuscation` to false to optimize for bandwidth if you trust the network links between your application and the OpenAI API.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/responses/res
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "id": "resp_67cb71b351908190a308f38594876",
3   "object": "response",
4   "created_at": 1741386163,
5   "status": "completed",
6   "error": null,
7   "incomplete_details": null,
8   "instructions": null,
9   "max_output_tokens": null,
10  "model": "gpt-4o-2024-08-06",
11  "output": [
12    {
13      "type": "message",
14      "id": "msg_67cb71b3c2b0819084d481baaa",
15      "status": "completed",
16      "role": "assistant",
17      "content": [
18        {
19          "type": "output_text",
```

starting_after integer Optional

The sequence number of the event after which to start streaming.

stream boolean Optional

If set to true, the model response data will be streamed to the client as it is generated using [server-sent events](#). See the [Streaming section below](#) for more information.

Returns

The [Response](#) object matching the specified ID.

Delete a model response

```
DELETE https://api.openai.com/v1/responses/{response_id}
```

Deletes a model response with the given ID.

Path parameters**response_id** string Required

The ID of the response to delete.

Returns

```

20         "text": "Silent circuits hum, \n"
21         "annotations": []
22     }
23   ]
24 }
25 ],
26 "parallel_tool_calls": true,
27 "previous_response_id": null,
28 "reasoning": {
29   "effort": null,
30   "summary": null
31 },
32 "store": true,
33 "temperature": 1.0,
34 "text": {
35   "format": {
36     "type": "text"
37   }
38 },
39 "tool_choice": "auto",
40 "tools": [],
41 "top_p": 1.0,
```

Example request
curl ↳ Copy

```

1 curl -X DELETE https://api.openai.com/v1/res
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KE
48   "output_tokens": 18.
```

Response
Copy

```

1 {
2   "id": "resp_6786a1bec27481909a17d673315b29
3   "object": "response",
```

4 "deleted": true

A success message.

Cancel a response

POST https://api.openai.com/v1/responses/{response_id}/cancel

Cancels a model response with the given ID. Only responses created with the `background` parameter set to `true` can be cancelled. [Learn more](#).

Path parameters

response_id string Required

The ID of the response to cancel.

Returns

A [Response](#) object.

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/respo
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KE
```

Response

⌚

```
1 {
2   "id": "resp_67cb71b351908190a308f38594876",
3   "object": "response",
4   "created_at": 1741386163,
5   "status": "completed",
6   "error": null,
7   "incomplete_details": null,
8   "instructions": null,
9   "max_output_tokens": null,
10  "model": "gpt-4o-2024-08-06",
11  "output": [
12    {
13      "type": "message",
14      "id": "msg_67cb71b3c2b0819084d481baaa",
15      "status": "completed",
16      "role": "assistant",
```

List input items

GET https://api.openai.com/v1/responses/{response_id}/input_items

Returns a list of input items for a given response.

Path parameters

response_id string Required

The ID of the response to retrieve input items for.

Query parameters

after string Optional

An item ID to list items after, used in pagination.

before string Optional

An item ID to list items before, used in pagination.

```

17     "content": [
18         {
19             "type": "output_text",
20             "text": "Silent circuits hum, \n",
21             "annotations": []
22         }
23     ]
24 }
25 ],
26 "parallel_tool_calls": true,
27 "previous_response_id": null,
28 "reasoning": {
29     "effort": null,

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/responses/res_id
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4
5     "type": "text"

```

Response

```

1 {
2     "object": "list",
3     "data": [
4         {
5             "id": "msg_abc123",
6             "type": "message",
7             "role": "user",
8             "content": [
9                 {
10                     "type": "input_text",
11                     "text": "Tell me a three sentence
12                 }
13             ]

```

include array Optional

Additional fields to include in the response. See the `include` parameter for Response creation above for more information.

```

14   }
15   ],
16   "first_id": "msg_abc123",
17   "last_id": "msg_abc123",
18   "has_more": false
19 }
```

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional

The order to return the input items in. Default is `desc`.

`asc` : Return the input items in ascending order.

`desc` : Return the input items in descending order.

Returns

A list of input item objects.

The response object

background boolean or null

Whether to run the model response in the background. [Learn more](#).

OBJECT The response object

```

1  {
2    "id": "resp_67ccd3a9da748190baa7f1570fe91
```

created_at number

Unix timestamp (in seconds) of when this Response was created.

error object or null

An error object returned when the model fails to generate a Response.

▽ Show properties

id string

Unique identifier for this Response.

incomplete_details object or null

Details about why the response is incomplete.

▽ Show properties

instructions string or array

A system (or developer) message inserted into the model's context.

When using along with `previous_response_id`, the instructions from a previous response will not be carried over to the next response. This makes it simple to swap out system (or developer) messages in new responses.

▽ Show possible types

max_output_tokens integer or null

An upper bound for the number of tokens that can be generated for a response, including visible output tokens and reasoning tokens.

max_tool_calls integer or null

The maximum number of total calls to built-in tools that can be processed in a response. This maximum number applies across all built-in tool calls, not per individual

```

3   "object": "response",
4   "created_at": 1741476777,
5   "status": "completed",
6   "error": null,
7   "incomplete_details": null,
8   "instructions": null,
9   "max_output_tokens": null,
10  "model": "gpt-4o-2024-08-06",
11  "output": [
12    {
13      "type": "message",
14      "id": "msg_67ccd3acc8d48190a77525dc6d",
15      "status": "completed",
16      "role": "assistant",
17      "content": [
18        {
19          "type": "output_text",
20          "text": "The image depicts a scene",
21          "annotations": []
22        }
23      ]
24    }
25  ],
26  "parallel_tool_calls": true,
27  "previous_response_id": null,
28  "reasoning": {
29    "effort": null,
30    "summary": null
31  },
32  "store": true,
33  "temperature": 1,
34  "text": {
35    "format": {
36      "type": "text"
37    }
38  }
39
```

tool. Any further attempts to call a tool by the model will be ignored.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string

Model ID used to generate the response, like `gpt-4o` or `o3`. OpenAI offers a wide range of models with different capabilities, performance characteristics, and price points. Refer to the [model guide](#) to browse and compare available models.

object string

The object type of this resource - always set to `response`.

output array

An array of content items generated by the model.

The length and order of items in the `output` array is dependent on the model's response.

Rather than accessing the first item in the `output` array and assuming it's an `assistant` message with the content generated by the model, you might consider using the `output_text` property where supported in SDKs.

▼ Show possible types

output_text string or null SDK Only

```

38 },
39 "tool_choice": "auto",
40 "tools": [],
41 "top_p": 1,
42 "truncation": "disabled",
43 "usage": {
44   "input_tokens": 328,
45   "input_tokens_details": {
46     "cached_tokens": 0
47   },
48   "output_tokens": 52,
49   "output_tokens_details": {
50     "reasoning_tokens": 0
51   },
52   "total_tokens": 380
53 },
54 "user": null,
55 "metadata": {}

```

SDK-only convenience property that contains the aggregated text output from all `output_text` items in the `output` array, if any are present. Supported in the Python and JavaScript SDKs.

parallel_tool_calls boolean

Whether to allow the model to run tool calls in parallel.

previous_response_id string or null

The unique ID of the previous response to the model. Use this to create multi-turn conversations. Learn more about [conversation state](#).

prompt object or null

Reference to a prompt template and its variables. [Learn more](#).

▼ Show properties

prompt_cache_key string

Used by OpenAI to cache responses for similar requests to optimize your cache hit rates. Replaces the `user` field. [Learn more](#).

reasoning object or null

o-series models only

Configuration options for [reasoning models](#).

▼ Show properties

safety_identifier string

A stable identifier used to help detect users of your application that may be violating OpenAI's usage policies. The IDs should be a string that uniquely identifies each user. We recommend hashing their username or email address, in order to avoid sending us any identifying information. [Learn more](#).

service_tier string or null

Specifies the processing type used for serving the request.

If set to 'auto', then the request will be processed with the service tier configured in the Project settings. Unless otherwise configured, the Project will use 'default'.

If set to 'default', then the request will be processed with the standard pricing and performance for the selected model.

If set to '[flex](#)' or 'priority', then the request will be processed with the corresponding service tier. [Contact sales](#) to learn more about Priority processing.

When not set, the default behavior is 'auto'.

When the `service_tier` parameter is set, the response body will include the `service_tier` value based on the processing mode actually used to serve the request. This response value may be different from the value set in the parameter.

status string

The status of the response generation. One of `completed`, `failed`, `in_progress`, `cancelled`, `queued`, or `incomplete`.

temperature number or null

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic. We generally recommend altering this or `top_p` but not both.

text object

Configuration options for a text response from the model. Can be plain text or structured JSON data. Learn more:

[Text inputs and outputs](#)

[Structured Outputs](#)

✓ Show properties

tool_choice string or object

How the model should select which tool (or tools) to use when generating a response.

See the `tools` parameter to see how to specify which tools the model can call.

✓ Show possible types

tools array

An array of tools the model may call while generating a response. You can specify

which tool to use by setting the `tool_choice` parameter.

The two categories of tools you can provide the model are:

Built-in tools: Tools that are provided by OpenAI that extend the model's capabilities, like [web search](#) or [file search](#). Learn more about [built-in tools](#).

Function calls (custom tools): Functions that are defined by you, enabling the model to call your own code with strongly typed arguments and outputs. Learn more about [function calling](#). You can also use custom tools to call your own code.

^ Hide possible types

Function object

Defines a function in your own code the model can choose to call. Learn more

about [function calling](#).

✓ Show properties

File search object

A tool that searches for relevant content from uploaded files. Learn more about

the [file search tool](#).

✓ Show properties

Web search preview object

This tool searches the web for relevant results to use in a response. Learn more about the [web search tool](#).

✓ Show properties

Computer use preview object

A tool that controls a virtual computer. Learn more about the [computer tool](#).

✓ Show properties

MCP tool object

Give the model access to additional tools via remote Model Context Protocol (MCP) servers. [Learn more about MCP](#).

✓ Show properties

Code interpreter object

A tool that runs Python code to help generate a response to a prompt.

✓ Show properties

Image generation tool object

A tool that generates images using a model like `gpt-image-1`.

✓ Show properties

Local shell tool object

A tool that allows the model to execute shell commands in a local environment.

✓ Show properties

Custom tool object

A custom tool that processes input using a specified format. Learn more about [custom tools](#).

✓ Show properties

top_logprobs integer or null

An integer between 0 and 20 specifying the number of most likely tokens to return at each token position, each with an associated log probability.

top_p number or null

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or `temperature` but not both.

truncation string or null

The truncation strategy to use for the model response.

`auto` : If the context of this response and previous ones exceeds the model's context window size, the model will truncate the response to fit the context window by dropping input items in the middle of the conversation.

`disabled` (default): If a model response will exceed the context window size for a model, the request will fail with a 400 error.

usage object

Represents token usage details including input tokens, output tokens, a breakdown of output tokens, and the total tokens used.

✓ Show properties

user Deprecated string

This field is being replaced by `safety_identifier` and `prompt_cache_key`. Use `prompt_cache_key` instead to maintain caching optimizations. A stable identifier

for your end-users. Used to boost cache hit rates by better bucketing similar requests and to help OpenAI detect and prevent abuse. [Learn more.](#)

The input item list

A list of Response items.

data array

A list of items used to generate this response.

▽ Show possible types

first_id string

The ID of the first item in the list.

has_more boolean

Whether there are more items available.

last_id string

The ID of the last item in the list.

object string

The type of object returned, must be `list`.

OBJECT The input item list

```
1  {
2    "object": "list",
3    "data": [
4      {
5        "id": "msg_abc123",
6        "type": "message",
7        "role": "user",
8        "content": [
9          {
10            "type": "input_text",
11            "text": "Tell me a three sentence"
12          }
13        ]
14      }
15    ],
16    "first_id": "msg_abc123",
17    "last_id": "msg_abc123",
18    "has_more": false
19 }
```

Streaming

When you create a Response with `stream` set to `true`, the server will emit server-sent events to the client as the Response is generated. This section contains the events that are emitted by the server.

[Learn more about streaming responses.](#)

response.created

An event that is emitted when a response is created.

response object

The response that was created.

✓ Show properties

sequence_number integer

The sequence number for this event.

type string

The type of the event. Always `response.created`.

OBJECT response.created

```
1  {
2    "type": "response.created",
3    "response": {
4      "id": "resp_67ccfcdd16748190a91872c75d3",
5      "object": "response",
6      "created_at": 1741487325,
7      "status": "in_progress",
8      "error": null,
9      "incomplete_details": null,
10     "instructions": null,
11     "max_output_tokens": null,
12     "model": "gpt-4o-2024-08-06",
13     "output": [],
14     "parallel_tool_calls": true,
15     "previous_response_id": null,
16     "reasoning": {
```

```

17     "effort": null,
18     "summary": null
19   },
20   "store": true,
21   "temperature": 1,
22   "text": {
23     "format": {
24       "type": "text"
25     }
26   },
27   "tool_choice": "auto",
28   "tools": [],
29   "top_p": 1,
30   "truncation": "disabled",
31   "usage": null,
32   "user": null,
33   "metadata": {}
34 },
35   "sequence_number": 1
36 }
```

OBJECT response.in_progress



```

1  {
2    "type": "response.in_progress",
3    "response": {
4      "id": "resp_67ccfcdd16748190a91872c75d3",
5      "object": "response",
6      "created_at": 1741487325,
7      "status": "in_progress",
8      "error": null,
9      "incomplete_details": null,
10     "instructions": null,
11     "max_output_tokens": null,
12     "model": "gpt-4o-2024-08-06",
13     "output": []
}
```

response.in_progress

Emitted when the response is in progress.

response object

The response that is in progress.

✗ Show properties

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.in_progress`.

```

14 "parallel_tool_calls": true,
15 "previous_response_id": null,
16 "reasoning": {
17   "effort": null,
18   "summary": null
19 },
20 "store": true,
21 "temperature": 1,
22 "text": {
23   "format": {
24     "type": "text"
25   }
26 },
27 "tool_choice": "auto",
28 "tools": [],
29 "top_p": 1,
30 "truncation": "disabled",
31 "usage": null,
32 "user": null,
33 "metadata": {}
34 },
35 "sequence_number": 1

```

OBJECT response.completed

```

1 {
2   "type": "response.completed",
3   "response": {
4     "id": "resp_123",
5     "object": "response",
6     "created_at": 1740855869,
7     "status": "completed",
8     "error": null,
9     "incomplete_details": null,
10    "input": [],
11    "instructions": null,

```

response.completed

Emitted when the model response is complete.

response object

Properties of the completed response.

✓ Show properties

sequence_number integer

The sequence number for this event.

type string

The type of the event. Always `response.completed`.

response.failed

An event that is emitted when a response fails.

`response` object

The response that failed.

✓ Show properties

`sequence_number` integer

The sequence number of this event.

```

12     "max_output_tokens": null,
13     "model": "gpt-4o-mini-2024-07-18",
14     "output": [
15       {
16         "id": "msg_123",
17         "type": "message",
18         "role": "assistant",
19         "content": [
20           {
21             "type": "output_text",
22             "text": "In a shimmering forest",
23             "annotations": []
24           }
25         ]
26       }
27     ],
28     "previous_response_id": null,
29     "reasoning_effort": null,
30     "store": false,
31     "temperature": 1,
32     "text": {
33       "format": {
34         "type": "text"
35       }
36     }
37   }.
```

OBJECT `response.failed`

```

1  {
2    "type": "response.failed",
3    "response": {
4      "id": "resp_123",
5      "object": "response",
6      "created_at": 1740855869,
7      "status": "failed",
8      "error": {
```

type string

The type of the event. Always `response.failed`.

```

 9   "code": "server_error",
10   "message": "The model failed to gener
11 },
12   "incomplete_details": null,
13   "instructions": null,
14   "max_output_tokens": null,
15   "model": "gpt-4o-mini-2024-07-18",
16   "output": [],
17   "previous_response_id": null,
18   "reasoning_effort": null,
19   "store": false,
20   "temperature": 1,
21   "text": {
22     "format": {
23       "type": "text"
24     }
25 },
26   "tool_choice": "auto",
27   "tools": [],
28   "top_p": 1,
29   "truncation": "disabled",
30   "usage": null,
31   "user": null,
32   "metadata": {}
33 }
34 }
```

response.incomplete

An event that is emitted when a response finishes as incomplete.

response object

The response that was incomplete.

✓ Show properties

OBJECT `response.incomplete`

```

1  {
2    "type": "response.incomplete",
3    "response": {
4      "id": "resp_123",
5      "object": "response",
6      "created_at": 1740855869,
```

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.incomplete`.

```

7   "status": "incomplete",
8   "error": null,
9   "incomplete_details": {
10    "reason": "max_tokens"
11 },
12   "instructions": null,
13   "max_output_tokens": null,
14   "model": "gpt-4o-mini-2024-07-18",
15   "output": [],
16   "previous_response_id": null,
17   "reasoning_effort": null,
18   "store": false,
19   "temperature": 1,
20   "text": {
21    "format": {
22     "type": "text"
23    }
24 },
25   "tool_choice": "auto",
26   "tools": [],
27   "top_p": 1,
28   "truncation": "disabled",
29   "usage": null,
30   "user": null,
31   "metadata": {}
32 },
33   "sequence_number": 1
34 }
```

response.output_item.added

Emitted when a new output item is added.

OBJECT `response.output_item.added`



item object

The output item that was added.

✓ Show possible types

output_index integer

The index of the output item that was added.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.output_item.added`.

```

1  {
2    "type": "response.output_item.added",
3    "output_index": 0,
4    "item": {
5      "id": "msg_123",
6      "status": "in_progress",
7      "type": "message",
8      "role": "assistant",
9      "content": []
10   },
11   "sequence_number": 1
12 }
```

response.output_item.done

Emitted when an output item is marked done.

item object

The output item that was marked done.

✓ Show possible types

output_index integer

The index of the output item that was marked done.

sequence_number integerOBJECT `response.output_item.done`

```

1  {
2    "type": "response.output_item.done",
3    "output_index": 0,
4    "item": {
5      "id": "msg_123",
6      "status": "completed",
7      "type": "message",
8      "role": "assistant",
9      "content": [
10        {
11          "text": "The task has been completed."
12        }
13      ]
14    }
15 }
```

The sequence number of this event.

type string

The type of the event. Always `response.output_item.done`.

```

11     "type": "output_text",
12     "text": "In a shimmering forest und
13     "annotations": []
14   }
15   ]
16 },
17 "sequence_number": 1
18 }
```

response.content_part.added

Emitted when a new content part is added.

content_index integer

The index of the content part that was added.

item_id string

The ID of the output item that the content part was added to.

output_index integer

The index of the output item that the content part was added to.

part object

The content part that was added.

OBJECT `response.content_part.added`

```

1  {
2    "type": "response.content_part.added",
3    "item_id": "msg_123",
4    "output_index": 0,
5    "content_index": 0,
6    "part": {
7      "type": "output_text",
8      "text": "",
9      "annotations": []
10   },
11   "sequence_number": 1
12 }
```

✓ Show possible types

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.content_part.added`.

response.content_part.done

Emitted when a content part is done.

content_index integer

The index of the content part that is done.

item_id string

The ID of the output item that the content part was added to.

output_index integer

The index of the output item that the content part was added to.

part object

The content part that is done.

✓ Show possible types

OBJECT `response.content_part.done`

```
1  {
2    "type": "response.content_part.done",
3    "item_id": "msg_123",
4    "output_index": 0,
5    "content_index": 0,
6    "sequence_number": 1,
7    "part": {
8      "type": "output_text",
9      "text": "In a shimmering forest under a",
10     "annotations": []
11   }
12 }
```

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.content_part.done`.

response.output_text.delta

Emitted when there is an additional text delta.

content_index integer

The index of the content part that the text delta was added to.

delta string

The text delta that was added.

item_id string

The ID of the output item that the text delta was added to.

logprobs array

The log probabilities of the tokens in the delta.

▼ Show properties

OBJECT `response.output_text.delta`

```
1 {
2   "type": "response.output_text.delta",
3   "item_id": "msg_123",
4   "output_index": 0,
5   "content_index": 0,
6   "delta": "In",
7   "sequence_number": 1
8 }
```

output_index integer

The index of the output item that the text delta was added to.

sequence_number integer

The sequence number for this event.

type string

The type of the event. Always `response.output_text.delta`.

response.output_text.done

Emitted when text content is finalized.

content_index integer

The index of the content part that the text content is finalized.

item_id string

The ID of the output item that the text content is finalized.

logprobs array

The log probabilities of the tokens in the delta.

✓ Show properties

output_index integerOBJECT `response.output_text.done` 

```
1 {
2   "type": "response.output_text.done",
3   "item_id": "msg_123",
4   "output_index": 0,
5   "content_index": 0,
6   "text": "In a shimmering forest under a sk
7   "sequence_number": 1
8 }
```

The index of the output item that the text content is finalized.

sequence_number integer

The sequence number for this event.

text string

The text content that is finalized.

type string

The type of the event. Always `response.output_text.done`.

response.refusal.delta

Emitted when there is a partial refusal text.

content_index integer

The index of the content part that the refusal text is added to.

delta string

The refusal text that is added.

item_id stringOBJECT `response.refusal.delta`

```
1 {
2   "type": "response.refusal.delta",
3   "item_id": "msg_123",
4   "output_index": 0,
5   "content_index": 0,
6   "delta": "refusal text so far",
7   "sequence_number": 1
8 }
```

The ID of the output item that the refusal text is added to.

output_index integer

The index of the output item that the refusal text is added to.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.refusal.delta`.

response.refusal.done

Emitted when refusal text is finalized.

content_index integer

The index of the content part that the refusal text is finalized.

item_id string

The ID of the output item that the refusal text is finalized.

output_index integer

The index of the output item that the refusal text is finalized.

refusal stringOBJECT `response.refusal.done`

```
1 {
2   "type": "response.refusal.done",
3   "item_id": "item-abc",
4   "output_index": 1,
5   "content_index": 2,
6   "refusal": "final refusal text",
7   "sequence_number": 1
8 }
```

The refusal text that is finalized.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.refusal.done`.

response.function_call_arguments.delta

Emitted when there is a partial function-call arguments delta.

delta string

The function-call arguments delta that is added.

item_id string

The ID of the output item that the function-call arguments delta is added to.

output_index integer

The index of the output item that the function-call arguments delta is added to.

sequence_number integerOBJECT `response.function_call_arguments.delta` ⓘ

```
1 {
2   "type": "response.function_call_arguments.",
3   "item_id": "item-abc",
4   "output_index": 0,
5   "delta": "{ \"arg\": "
6   "sequence_number": 1
7 }
```

The sequence number of this event.

type string

The type of the event. Always `response.function_call_arguments.delta`.

response.function_call_arguments.done

Emitted when function-call arguments are finalized.

arguments string

The function-call arguments.

item_id string

The ID of the item.

output_index integer

The index of the output item.

sequence_number integer

The sequence number of this event.

type stringOBJECT `response.function_call_arguments.done`

```
1 {
2   "type": "response.function_call_arguments.",
3   "item_id": "item-abc",
4   "output_index": 1,
5   "arguments": "{ \"arg\": 123 }",
6   "sequence_number": 1
7 }
```

response.file_search_call.in_progress

Emitted when a file search call is initiated.

item_id string

The ID of the output item that the file search call is initiated.

output_index integer

The index of the output item that the file search call is initiated.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.file_search_call.in_progress`.

OBJECT `response.file_search_call.in_progress`

```
1 {  
2   "type": "response.file_search_call.in_prog  
3   "output_index": 0,  
4   "item_id": "fs_123",  
5   "sequence_number": 1  
6 }
```

response.file_search_call.searching

Emitted when a file search is currently searching.

item_id string

The ID of the output item that the file search call is initiated.

OBJECT `response.file_search_call.searching`

```
1 {  
2   "type": "response.file_search_call.searchi  
3   "output_index": 0,
```

output_index integer

The index of the output item that the file search call is searching.

```
4   "item_id": "fs_123",
5   "sequence_number": 1
6 }
```

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.file_search_call.searching`.

response.file_search_call.completed

Emitted when a file search call is completed (results found).

item_id string

The ID of the output item that the file search call is initiated.

output_index integer

The index of the output item that the file search call is initiated.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.file_search_call.completed`.

OBJECT `response.file_search_call.completed`

```

1 {
2   "type": "response.file_search_call.completed"
3   "output_index": 0,
4   "item_id": "fs_123",
5   "sequence_number": 1
6 }
```

response.web_search_call.in_progress

Emitted when a web search call is initiated.

item_id string

Unique ID for the output item associated with the web search call.

output_index integer

OBJECT `response.web_search_call.in_progress`

```

1 {
2   "type": "response.web_search_call.in_progress"
3   "output_index": 0,
4   "item_id": "ws_123",
5
6 }
```

The index of the output item that the web search call is associated with.

sequence_number integer

The sequence number of the web search call being processed.

type string

The type of the event. Always `response.web_search_call.in_progress`.

response.web_search_call.searching

Emitted when a web search call is executing.

item_id string

Unique ID for the output item associated with the web search call.

output_index integer

The index of the output item that the web search call is associated with.

sequence_number integer

The sequence number of the web search call being processed.

type string

The type of the event. Always `response.web_search_call.searching`.

OBJECT `response.web_search_call.searching` 

```
1 {
2   "type": "response.web_search_call.searchin
3   "output_index": 0,
4   "item_id": "ws_123",
5   "sequence_number": 0
6 }
```

response.web_search_call.completed

Emitted when a web search call is completed.

item_id string

Unique ID for the output item associated with the web search call.

output_index integer

The index of the output item that the web search call is associated with.

sequence_number integer

The sequence number of the web search call being processed.

type string

The type of the event. Always `response.web_search_call.completed`.

OBJECT `response.web_search_call.completed`

```

1 {
2   "type": "response.web_search_call.complete"
3   "output_index": 0,
4   "item_id": "ws_123",
5   "sequence_number": 0
6 }
```

response.reasoning_summary_part.added

Emitted when a new reasoning summary part is added.

item_id string

The ID of the item this summary part is associated with.

OBJECT `response.reasoning_summary_part.added`

```

1 {
2   "type": "response.reasoning_summary_part."
3   "item_id": "rs_6806bfca0b2481918a57483080
```

output_index integer

The index of the output item this summary part is associated with.

part object

The summary part that was added.

▽ Show properties

```

4   "output_index": 0,
5   "summary_index": 0,
6   "part": {
7     "type": "summary_text",
8     "text": ""
9   },
10  "sequence_number": 1
11 }
```

sequence_number integer

The sequence number of this event.

summary_index integer

The index of the summary part within the reasoning summary.

type string

The type of the event. Always `response.reasoning_summary_part.added`.

response.reasoning_summary_part.done

Emitted when a reasoning summary part is completed.

item_id string

The ID of the item this summary part is associated with.

output_index integer
OBJECT `response.reasoning_summary_part.done`

```

1  {
2    "type": "response.reasoning_summary_part.",
3    "item_id": "rs_6806bfca0b2481918a57483080"
4    "output_index": 0,
5    "summary_index": 0,
6    "part": {
```

The index of the output item this summary part is associated with.

part object

The completed summary part.

▼ Show properties

```

7   "type": "summary_text",
8   "text": "**Responding to a greeting**\n
9 },
10 "sequence_number": 1
11 }
```

sequence_number integer

The sequence number of this event.

summary_index integer

The index of the summary part within the reasoning summary.

type string

The type of the event. Always `response.reasoning_summary_part.done`.

response.reasoning_summary_text.delta

Emitted when a delta is added to a reasoning summary text.

delta string

The text delta that was added to the summary.

item_id string

OBJECT `response.reasoning_summary_text.delta`

```

1 {
2   "type": "response.reasoning_summary_text.d
3   "item_id": "rs_6806bfca0b2481918a574830806
4   "output_index": 0,
5   "summary_index": 0,
```

The ID of the item this summary text delta is associated with.

```
6   "delta": "**Responding to a greeting**\n\n'
7   "sequence_number": 1
8 }
```

output_index integer

The index of the output item this summary text delta is associated with.

sequence_number integer

The sequence number of this event.

summary_index integer

The index of the summary part within the reasoning summary.

type string

The type of the event. Always `response.reasoning_summary_text.delta`.

response.reasoning_summary_text.done

Emitted when a reasoning summary text is completed.

OBJECT `response.reasoning_summary_text.done`

item_id string

The ID of the item this summary text is associated with.

```
1 {
2   "type": "response.reasoning_summary_text.d
3   "item_id": "rs_6806bfca0b2481918a574830806
4   "output_index": 0,
5   "summary_index": 0,
6   "text": "**Responding to a greeting**\n\nT
7   "sequence_number": 1
8 }
```

output_index integer

The index of the output item this summary text is associated with.

sequence_number integer

The sequence number of this event.

summary_index integer

The index of the summary part within the reasoning summary.

text string

The full text of the completed reasoning summary.

type string

The type of the event. Always `response.reasoning_summary_text.done`.

response.reasoning_text.delta

Emitted when a delta is added to a reasoning text.

content_index integer

The index of the reasoning content part this delta is associated with.

delta string

The text delta that was added to the reasoning content.

item_id stringOBJECT `response.reasoning_text.delta`

```
1 {
2   "type": "response.reasoning_text.delta",
3   "item_id": "rs_123",
4   "output_index": 0,
5   "content_index": 0,
6   "delta": "The",
7   "sequence_number": 1
8 }
```

The ID of the item this reasoning text delta is associated with.

output_index integer

The index of the output item this reasoning text delta is associated with.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `response.reasoning_text.delta`.

response.reasoning_text.done

Emitted when a reasoning text is completed.

content_index integer

The index of the reasoning content part.

item_id string

The ID of the item this reasoning text is associated with.

output_index integer

The index of the output item this reasoning text is associated with.

sequence_number integerOBJECT `response.reasoning_text.done` 

```
1 {
2   "type": "response.reasoning_text.done",
3   "item_id": "rs_123",
4   "output_index": 0,
5   "content_index": 0,
6   "text": "The user is asking...",
7   "sequence_number": 4
8 }
```

The sequence number of this event.

text string

The full text of the completed reasoning content.

type string

The type of the event. Always `response.reasoning_text.done`.

response.image_generation_call.completed

Emitted when an image generation tool call has completed and the final image is available.

item_id string

The unique identifier of the image generation item being processed.

output_index integer

The index of the output item in the response's output array.

sequence_number integer

The sequence number of this event.

type stringOBJECT `response.image_generation_call.comple...` 

```
1 {
2   "type": "response.image_generation_call.co
3   "output_index": 0,
4   "item_id": "item-123",
5   "sequence_number": 1
6 }
```

The type of the event. Always 'response.image_generation_call.completed'.

response.image_generation_call.generating

Emitted when an image generation tool call is actively generating an image (intermediate state).

item_id string

The unique identifier of the image generation item being processed.

output_index integer

The index of the output item in the response's output array.

sequence_number integer

The sequence number of the image generation item being processed.

type string

The type of the event. Always 'response.image_generation_call.generating'.

OBJECT `response.image_generation_call.generating`

```
1 {
2   "type": "response.image_generation_call.ge...
3   "output_index": 0,
4   "item_id": "item-123",
5   "sequence_number": 0
6 }
```

response.image_generation_call.in_progress

Emitted when an image generation tool call is in progress.

item_id string

The unique identifier of the image generation item being processed.

output_index integer

The index of the output item in the response's output array.

sequence_number integer

The sequence number of the image generation item being processed.

type string

The type of the event. Always 'response.image_generation_call.in_progress'.

OBJECT `response.image_generation_call.in_pro...`

```

1 {
2   "type": "response.image_generation_call.in_
3   "output_index": 0,
4   "item_id": "item-123",
5   "sequence_number": 0

```

response.image_generation_call.partial_image

Emitted when a partial image is available during image generation streaming.

item_id string

The unique identifier of the image generation item being processed.

output_index integer

The index of the output item in the response's output array.

partial_image_b64 stringOBJECT `response.image_generation_call.partia...`

```

1 {
2   "type": "response.image_generation_call.pa_
3   "output_index": 0,
4   "item_id": "item-123",
5   "sequence_number": 0,
6   "partial_image_index": 0,
7   "partial_image_b64": "..."
8 }

```

Base64-encoded partial image data, suitable for rendering as an image.

partial_image_index integer

0-based index for the partial image (backend is 1-based, but this is 0-based for the user).

sequence_number integer

The sequence number of the image generation item being processed.

type string

The type of the event. Always 'response.image_generation_call.partial_image'.

response.mcp_call_arguments.delta

Emitted when there is a delta (partial update) to the arguments of an MCP tool call.

delta string

A JSON string containing the partial update to the arguments for the MCP tool call.

item_id string

The unique identifier of the MCP tool call item being processed.

OBJECT `response.mcp_call_arguments.delta`

```
1 {
2   "type": "response.mcp_call_arguments.delta"
3   "output_index": 0,
4   "item_id": "item-abc",
5   "delta": "{",
6   "sequence_number": 1
7 }
```

output_index integer

The index of the output item in the response's output array.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_call_arguments.delta'.

response.mcp_call_arguments.done

Emitted when the arguments for an MCP tool call are finalized.

arguments string

A JSON string containing the finalized arguments for the MCP tool call.

item_id string

The unique identifier of the MCP tool call item being processed.

output_index integer

The index of the output item in the response's output array.

sequence_number integer

The sequence number of this event.

OBJECT `response.mcp_call_arguments.done` 

```
1 {
2   "type": "response.mcp_call_arguments.done"
3   "output_index": 0,
4   "item_id": "item-abc",
5   "arguments": "{\"arg1\": \"value1\", \"arg2\": \"value2\"}"
6   "sequence_number": 1
7 }
```

type string

The type of the event. Always 'response.mcp_call.arguments.done'.

response.mcp_call.completed

Emitted when an MCP tool call has completed successfully.

item_id string

The ID of the MCP tool call item that completed.

output_index integer

The index of the output item that completed.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_call.completed'.

OBJECT response.mcp_call.completed

```
1 {
2   "type": "response.mcp_call.completed",
3   "sequence_number": 1,
4   "item_id": "mcp_682d437d90a88191bf88cd03aa
5   "output_index": 0
6 }
```

response.mcp_call.failed

Emitted when an MCP tool call has failed.

item_id string

The ID of the MCP tool call item that failed.

output_index integer

The index of the output item that failed.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_call.failed'.

OBJECT response.mcp_call.failed

```
1 {  
2   "type": "response.mcp_call.failed",  
3   "sequence_number": 1,  
4   "item_id": "mcp_682d437d90a88191bf88cd03aa  
5   "output_index": 0  
6 }
```

response.mcp_call.in_progress

Emitted when an MCP tool call is in progress.

item_id string

The unique identifier of the MCP tool call item being processed.

output_index integer

The index of the output item in the response's output array.

OBJECT response.mcp_call.in_progress

```
1 {  
2   "type": "response.mcp_call.in_progress",  
3   "sequence_number": 1,  
4   "output_index": 0,  
5   "item_id": "mcp_682d437d90a88191bf88cd03aa  
6 }
```

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_call.in_progress'.

response.mcp_list_tools.completed

Emitted when the list of available MCP tools has been successfully retrieved.

item_id string

The ID of the MCP tool call item that produced this output.

output_index integer

The index of the output item that was processed.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_list_tools.completed'.

OBJECT response.mcp_list_tools.completed

```

1 {
2   "type": "response.mcp_list_tools.completed"
3   "sequence_number": 1,
4   "output_index": 0,
5   "item_id": "mcpl_682d4379df088191886b70f4e"
6 }
```

response.mcp_list_tools.failed

Emitted when the attempt to list available MCP tools has failed.

item_id string

The ID of the MCP tool call item that failed.

output_index integer

The index of the output item that failed.

OBJECT response.mcp_list_tools.failed

```

1 {
2   "type": "response.mcp_list_tools.failed",
3   "sequence_number": 1,
4   "output_index": 0,
5   "item_id": "mcpl_682d4379df088191886b70f4e"
6 }
```

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_list_tools.failed'.

response.mcp_list_tools.in_progress

Emitted when the system is in the process of retrieving the list of available MCP tools.

item_id string

The ID of the MCP tool call item that is being processed.

output_index integer

The index of the output item that is being processed.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.mcp_list_tools.in_progress'.

OBJECT response.mcp_list_tools.in_progress

```
1 {
2   "type": "response.mcp_list_tools.in_progre
3   "sequence_number": 1,
4   "output_index": 0,
5   "item_id": "mcpl_682d4379df088191886b70f4e
6 }
```

response.code_interpreter_call.in_progress

Emitted when a code interpreter call is in progress.

item_id string

The unique identifier of the code interpreter tool call item.

output_index integer

The index of the output item in the response for which the code interpreter call is in progress.

sequence_number integer

The sequence number of this event, used to order streaming events.

type string

The type of the event. Always `response.code_interpreter_call.in_progress`.

OBJECT `response.code_interpreter_call.in_pro...` 

```
1 {
2   "type": "response.code_interpreter_call.in_
3   "output_index": 0,
4   "item_id": "ci_12345",
5   "sequence_number": 1
6 }
```

response.code_interpreter_call.interpreting

Emitted when the code interpreter is actively interpreting the code snippet.

item_id string

OBJECT `response.code_interpreter_call.interp...` 

```
1 {
2   "type": "response.code_interpreter_call.in
```

The unique identifier of the code interpreter tool call item.

output_index integer

The index of the output item in the response for which the code interpreter is interpreting code.

```
3 "output_index": 4,
4 "item_id": "ci_12345",
5 "sequence_number": 1
6 }
```

sequence_number integer

The sequence number of this event, used to order streaming events.

type string

The type of the event. Always `response.code_interpreter_call.interpreting`.

response.code_interpreter_call.completed

Emitted when the code interpreter call is completed.

item_id string

The unique identifier of the code interpreter tool call item.

output_index integer

The index of the output item in the response for which the code interpreter call is completed.

OBJECT `response.code_interpreter_call.comple...`

```
1 {
2   "type": "response.code_interpreter_call.co
3   "output_index": 5,
4   "item_id": "ci_12345",
5   "sequence_number": 1
6 }
```

sequence_number integer

The sequence number of this event, used to order streaming events.

type string

The type of the event. Always `response.code_interpreter_call.completed`.

`response.code_interpreter_call_code.delta`

Emitted when a partial code snippet is streamed by the code interpreter.

delta string

The partial code snippet being streamed by the code interpreter.

item_id string

The unique identifier of the code interpreter tool call item.

output_index integer

The index of the output item in the response for which the code is being streamed.

sequence_number integer

The sequence number of this event, used to order streaming events.

type string

The type of the event. Always `response.code_interpreter_call_code.delta`.

OBJECT `response.code_interpreter_call_code.delta`

```

1 {
2   "type": "response.code_interpreter_call_code.delta",
3   "output_index": 0,
4   "item_id": "ci_12345",
5   "delta": "print('Hello, world')",
6   "sequence_number": 1
7 }
```

response.code_interpreter_call_code.done

Emitted when the code snippet is finalized by the code interpreter.

code string

The final code snippet output by the code interpreter.

item_id string

OBJECT `response.code_interpreter_call_code.done`

```

1 {
2   "type": "response.code_interpreter_call_code.done",
3   "output_index": 3,
4   "item_id": "ci_12345",
5   "code": "print('done')",
```

The unique identifier of the code interpreter tool call item.

```
6   "sequence_number": 1
```

```
7 }
```

output_index integer

The index of the output item in the response for which the code is finalized.

sequence_number integer

The sequence number of this event, used to order streaming events.

type string

The type of the event. Always `response.code_interpreter_call_code.done`.

response.output_text.annotation.added

Emitted when an annotation is added to output text content.

OBJECT `response.output_text.annotation.added`



annotation object

The annotation object being added. (See annotation schema for details.)

```
1   {
2     "type": "response.output_text.annotation.",
3     "item_id": "item-abc",
4     "output_index": 0,
5     "content_index": 0,
6     "annotation_index": 0,
7     "annotation": {
8       "type": "text_annotation",
```

annotation_index integer

The index of the annotation within the content part.

content_index integer

The index of the content part within the output item.

item_id string

The unique identifier of the item to which the annotation is being added.

output_index integer

The index of the output item in the response's output array.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always 'response.output_text.annotation.added'.

```

9   "text": "This is a test annotation",
10  "start": 0,
11  "end": 10
12 },
13 "sequence_number": 1
14 }
```

response.queued

Emitted when a response is queued and waiting to be processed.

response object

The full response object that is queued.

▼ Show properties

sequence_number integer

OBJECT response.queued

```

1  {
2    "type": "response.queued",
3    "response": {
4      "id": "res_123",
5      "status": "queued",
6      "created_at": "2021-01-01T00:00:00Z",
7      "updated_at": "2021-01-01T00:00:00Z"
8    },
```

The sequence number for this event.

```
9     "sequence_number": 1
10 }
```

type string

The type of the event. Always 'response.queued'.

response.custom_tool_call_input.delta

Event representing a delta (partial update) to the input of a custom tool call.

delta string

The incremental input data (delta) for the custom tool call.

item_id string

Unique identifier for the API item associated with this event.

output_index integer

The index of the output this delta applies to.

sequence_number integer

The sequence number of this event.

type string

OBJECT response.custom_tool_call_input.delta [🔗](#)

```
1 {
2   "type": "response.custom_tool_call_input.d
3   "output_index": 0,
4   "item_id": "ctc_1234567890abcdef",
5   "delta": "partial input text"
6 }
```

The event type identifier.

response.custom_tool_call_input.done

Event indicating that input for a custom tool call is complete.

input string

The complete input data for the custom tool call.

item_id string

Unique identifier for the API item associated with this event.

output_index integer

The index of the output this event applies to.

sequence_number integer

The sequence number of this event.

type string

The event type identifier.

OBJECT response.custom_tool_call_input.done

```
1 {
2   "type": "response.custom_tool_call_input.d
3   "output_index": 0,
4   "item_id": "ctc_1234567890abcdef",
5   "input": "final complete input text"
6 }
```

error

Emitted when an error occurs.

code string or null

The error code.

message string

The error message.

param string or null

The error parameter.

sequence_number integer

The sequence number of this event.

type string

The type of the event. Always `error`.

OBJECT `error`

```
1 {
2   "type": "error",
3   "code": "ERR_SOMETHING",
4   "message": "Something went wrong",
5   "param": null,
6   "sequence_number": 1
7 }
```

Webhook Events

Webhooks are HTTP requests sent by OpenAI to a URL you specify when certain events happen during the course of API usage.

[Learn more about webhooks.](#)

response.completed

Sent when a background response has been completed.

created_at integer

The Unix timestamp (in seconds) of when the model response was completed.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `response.completed`.

OBJECT `response.completed`

```
1 {  
2   "id": "evt_abc123",  
3   "type": "response.completed",  
4   "created_at": 1719168000,  
5   "data": {  
6     "id": "resp_abc123"  
7   }  
8 }
```

response.cancelled

Sent when a background response has been cancelled.

created_at integer

The Unix timestamp (in seconds) of when the model response was cancelled.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `response.cancelled`.

OBJECT `response.cancelled`

```

1 {
2   "id": "evt_abc123",
3   "type": "response.cancelled",
4   "created_at": 1719168000,
5   "data": {
6     "id": "resp_abc123"
7   }
8 }
```

response.failed

Sent when a background response has failed.

created_at integer

The Unix timestamp (in seconds) of when the model response failed.

OBJECT `response.failed`

```

1 {
2   "id": "evt_abc123",
3   "type": "response.failed",
4   "created_at": 1719168000,
```

data object

Event data payload.

▼ Show properties

```

5   "data": {
6     "id": "resp_abc123"
7   }
8 }
```

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `response.failed`.

response.incomplete

Sent when a background response has been interrupted.

created_at integer

The Unix timestamp (in seconds) of when the model response was interrupted.

data object

Event data payload.

▼ Show properties

id stringOBJECT `response.incomplete`

```

1 {
2   "id": "evt_abc123",
3   "type": "response.incomplete",
4   "created_at": 1719168000,
5   "data": {
6     "id": "resp_abc123"
7   }
8 }
```

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `response.incomplete`.

batch.completed

Sent when a batch API request has been completed.

created_at integer

The Unix timestamp (in seconds) of when the batch API request was completed.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object stringOBJECT `batch.completed`

```
1 {
2   "id": "evt_abc123",
3   "type": "batch.completed",
4   "created_at": 1719168000,
5   "data": {
6     "id": "batch_abc123"
7   }
8 }
```

The object of the event. Always `event`.

type string

The type of the event. Always `batch.cancelled`.

batch.cancelled

Sent when a batch API request has been cancelled.

created_at integer

The Unix timestamp (in seconds) of when the batch API request was cancelled.

data object

Event data payload.

▽ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `batch.cancelled`.

OBJECT `batch.cancelled`

```
1 {  
2   "id": "evt_abc123",  
3   "type": "batch.cancelled",  
4   "created_at": 1719168000,  
5   "data": {  
6     "id": "batch_abc123"  
7   }  
8 }
```

batch.expired

Sent when a batch API request has expired.

created_at integer

The Unix timestamp (in seconds) of when the batch API request expired.

OBJECT `batch.expired`

```
1 {  
2   "id": "evt_abc123",  
3   "type": "batch.expired",  
4   "created_at": 1719168000,
```

data object

Event data payload.

▼ Show properties

```

5   "data": {
6     "id": "batch_abc123"
7   }
8 }
```

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `batch.expired`.

batch.failed

Sent when a batch API request has failed.

created_at integer

The Unix timestamp (in seconds) of when the batch API request failed.

data object

Event data payload.

▼ Show properties

id stringOBJECT `batch.failed`

```

1 {
2   "id": "evt_abc123",
3   "type": "batch.failed",
4   "created_at": 1719168000,
5   "data": {
6     "id": "batch_abc123"
7   }
8 }
```

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `batch.failed`.

fine_tuning.job.succeeded

Sent when a fine-tuning job has succeeded.

created_at integer

The Unix timestamp (in seconds) of when the fine-tuning job succeeded.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

OBJECT `fine_tuning.job.succeeded`

```
1 {
2   "id": "evt_abc123",
3   "type": "fine_tuning.job.succeeded",
4   "created_at": 1719168000,
5   "data": {
6     "id": "ftjob_abc123"
7   }
8 }
```

object string

The object of the event. Always `event`.

type string

The type of the event. Always `fine_tuning.job.failed`.

fine_tuning.job.failed

Sent when a fine-tuning job has failed.

created_at integer

The Unix timestamp (in seconds) of when the fine-tuning job failed.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type stringOBJECT `fine_tuning.job.failed`

```
1 {
2   "id": "evt_abc123",
3   "type": "fine_tuning.job.failed",
4   "created_at": 1719168000,
5   "data": {
6     "id": "ftjob_abc123"
7   }
8 }
```

The type of the event. Always `fine_tuning.job.failed`.

fine_tuning.job.cancelled

Sent when a fine-tuning job has been cancelled.

created_at integer

The Unix timestamp (in seconds) of when the fine-tuning job was cancelled.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `fine_tuning.job.cancelled`.

OBJECT `fine_tuning.job.cancelled`

```
1 {
2   "id": "evt_abc123",
3   "type": "fine_tuning.job.cancelled",
4   "created_at": 1719168000,
5   "data": {
6     "id": "ftjob_abc123"
7   }
8 }
```

eval.run.succeeded

Sent when an eval run has succeeded.

created_at integer

The Unix timestamp (in seconds) of when the eval run succeeded.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `eval.run.succeeded`.

OBJECT `eval.run.succeeded`

```
1 {
2   "id": "evt_abc123",
3   "type": "eval.run.succeeded",
4   "created_at": 1719168000,
5   "data": {
6     "id": "evalrun_abc123"
7   }
8 }
```

eval.run.failed

Sent when an eval run has failed.

created_at integer

The Unix timestamp (in seconds) of when the eval run failed.

data object

Event data payload.

▼ Show properties

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `eval.run.failed`.

OBJECT eval.run.failed

```
1 {
2   "id": "evt_abc123",
3   "type": "eval.run.failed",
4   "created_at": 1719168000,
5   "data": {
6     "id": "evalrun_abc123"
7   }
8 }
```

eval.run.canceled

Sent when an eval run has been canceled.

created_at integer

OBJECT eval.run.canceled

The Unix timestamp (in seconds) of when the eval run was canceled.

data object

Event data payload.

▼ Show properties

```
1  {
2    "id": "evt_abc123",
3    "type": "eval.run.canceled",
4    "created_at": 1719168000,
5    "data": {
6      "id": "evalrun_abc123"
```

id string

The unique ID of the event.

object string

The object of the event. Always `event`.

type string

The type of the event. Always `eval.run.canceled`.

Audio

Learn how to turn audio into text or text into audio.

Related guide: [Speech to text](#)

Create speech

POST <https://api.openai.com/v1/audio/speech>

Generates audio from the input text.

Request body

input string Required

The text to generate audio for. The maximum length is 4096 characters.

model string Required

One of the available TTS models: `tts-1` , `tts-1-hd` or `gpt-4o-mini-tts` .

voice string Required

The voice to use when generating the audio. Supported voices are `alloy` , `ash` , `ballad` , `coral` , `echo` , `fable` , `onyx` , `nova` , `sage` , `shimmer` , and `verse` . Previews of the voices are available in the [Text to speech guide](#).

instructions string Optional

Control the voice of your generated audio with additional instructions. Does not work with `tts-1` or `tts-1-hd` .

response_format string Optional Defaults to mp3

The format to audio in. Supported formats are `mp3` , `opus` , `aac` , `flac` , `wav` , and `pcm` .

speed number Optional Defaults to 1

The speed of the generated audio. Select a value from `0.25` to `4.0` . `1.0` is the default.

Default

SSE Stream Format

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/audio/speech
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5   "model": "gpt-4o-mini-tts",
6   "input": "The quick brown fox jumped ove
7   "voice": "alloy"
8 }' \
9 --output speech.mp3
```

stream_format string Optional Defaults to audio

The format to stream the audio in. Supported formats are `sse` and `audio`. `sse`

is not supported for `tts-1` or `tts-1-hd`.

Returns

The audio file content or a [stream of audio events](#).

Create transcription

POST <https://api.openai.com/v1/audio/transcriptions>

[Default](#) [Streaming](#) [Logprobs](#) [Word timestamps](#)

Transcribes audio into the input language.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/audio/transcr  
2 -H "Authorization: Bearer $OPENAI_API_KEY"  
3 -H "Content-Type: multipart/form-data" \  
4 -F file="@/path/to/file/audio.mp3" \  
5 -F model="gpt-4o-transcribe"
```

Request body

file file Required

The audio file object (not file name) to transcribe, in one of these formats: flac, mp3, mp4, mpeg, mpga, m4a, ogg, wav, or webm.

model string Required

Response

⌂

```
1 {  
2   "text": "Imagine the wildest idea that yo  
3   "usage": {
```

ID of the model to use. The options are `gpt-4o-transcribe`, `gpt-4o-mini-transcribe`, and `whisper-1` (which is powered by our open source Whisper V2 model).

chunking_strategy "auto" or object Optional

Controls how the audio is cut into chunks. When set to `"auto"`, the server first normalizes loudness and then uses voice activity detection (VAD) to choose boundaries. `server_vad` object can be provided to tweak VAD detection parameters manually. If unset, the audio is transcribed as a single block.

▼ Show possible types

include[] array Optional

Additional information to include in the transcription response. `logprobs` will return the log probabilities of the tokens in the response to understand the model's confidence in the transcription. `logprobs` only works with `response_format` set to `json` and only with the models `gpt-4o-transcribe` and `gpt-4o-mini-transcribe`.

language string Optional

The language of the input audio. Supplying the input language in ISO-639-1 (e.g. `en`) format will improve accuracy and latency.

prompt string Optional

An optional text to guide the model's style or continue a previous audio segment. The `prompt` should match the audio language.

response_format string Optional Defaults to json

The format of the output, in one of these options: `json`, `text`, `srt`, `verbose_json`, or `vtt`. For `gpt-4o-transcribe` and

```

4     "type": "tokens",
5     "input_tokens": 14,
6     "input_token_details": {
7       "text_tokens": 0,
8       "audio_tokens": 14
9     },
10    "output_tokens": 45,
11    "total_tokens": 59
12  }
13 }
```

`gpt-4o-mini-transcribe`, the only supported format is `json`.

stream boolean or null Optional Defaults to false

If set to true, the model response data will be streamed to the client as it is generated using [server-sent events](#). See the [Streaming section of the Speech-to-Text guide](#) for more information.

Note: Streaming is not supported for the `whisper-1` model and will be ignored.

temperature number Optional Defaults to 0

The sampling temperature, between 0 and 1. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic. If set to 0, the model will use [log probability](#) to automatically increase the temperature until certain thresholds are hit.

timestamp_granularities[] array Optional Defaults to segment

The timestamp granularities to populate for this transcription. `response_format` must be set `verbose_json` to use timestamp granularities. Either or both of these options are supported: `word`, or `segment`. Note: There is no additional latency for segment timestamps, but generating word timestamps incurs additional latency.

Returns

The [transcription object](#), a [verbose transcription object](#) or a [stream of transcript events](#).

Create translation

POST <https://api.openai.com/v1/audio/translations>

Translates audio into English.

Request body

file file Required

The audio file object (not file name) translate, in one of these formats: flac, mp3, mp4, mpeg, mpg, m4a, ogg, wav, or webm.

model string or "whisper-1" Required

ID of the model to use. Only `whisper-1` (which is powered by our open source Whisper V2 model) is currently available.

prompt string Optional

An optional text to guide the model's style or continue a previous audio segment. The `prompt` should be in English.

response_format string Optional Defaults to json

The format of the output, in one of these options: `json`, `text`, `srt`, `verbose_json`, or `vtt`.

temperature number Optional Defaults to 0

The sampling temperature, between 0 and 1. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and

Example request

curl ⚡ ↗

```
1 curl https://api.openai.com/v1/audio/translations  
2 -H "Authorization: Bearer $OPENAI_API_KEY"  
3 -H "Content-Type: multipart/form-data" \  
4 -F file="@/path/to/file/german.m4a" \  
5 -F model="whisper-1"
```

Response

↗

```
1 {  
2   "text": "Hello, my name is Wolfgang and I  
3 }
```

deterministic. If set to 0, the model will use `log_probability` to automatically increase the temperature until certain thresholds are hit.

Returns

The translated text.

The transcription object (JSON)

Represents a transcription response returned by model, based on the provided input.

logprobs array

The log probabilities of the tokens in the transcription. Only returned with the models `gpt-4o-transcribe` and `gpt-4o-mini-transcribe` if `logprobs` is added to the `include` array.

✓ Show properties

text string

The transcribed text.

usage object

Token usage statistics for the request.

✓ Show possible types

OBJECT The transcription object (JSON)

```
1  {
2    "text": "Imagine the wildest idea that yo
3    "usage": {
4      "type": "tokens",
5      "input_tokens": 14,
6      "input_token_details": {
7        "text_tokens": 10,
8        "audio_tokens": 4
9      },
10     "output_tokens": 101,
11     "total_tokens": 115
12   }
13 }
```

The transcription object (Verbose JSON)

Represents a verbose json transcription response returned by model, based on the provided input.

duration number

The duration of the input audio.

language string

The language of the input audio.

segments array

Segments of the transcribed text and their corresponding details.

▼ Show properties

text string

The transcribed text.

usage object

Usage statistics for models billed by audio input duration.

▼ Show properties

words array

Extracted words and their corresponding timestamps.

OBJECT The transcription object (Verbose JSON)

```
1  {
2    "task": "transcribe",
3    "language": "english",
4    "duration": 8.470000267028809,
5    "text": "The beach was a popular spot on
6    "segments": [
7      {
8        "id": 0,
9        "seek": 0,
10       "start": 0.0,
11       "end": 3.319999933242798,
12       "text": " The beach was a popular spo
13       "tokens": [
14         50364, 440, 7534, 390, 257, 3743, 4
15       ],
16       "temperature": 0.0,
17       "avg_logprob": -0.2860786020755768,
18       "compression_ratio": 1.23636364936828
19       "no_speech_prob": 0.00985979475080967
20     },
21     ...
22   ],
23   "usage": {
24     "type": "duration",
25     "seconds": 9
26   }
27 }
```

✓ Show properties

26 }

Stream Event (speech.audio.delta)

Emitted for each chunk of audio data generated during speech synthesis.

audio string

A chunk of Base64-encoded audio data.

type string

The type of the event. Always `speech.audio.delta`.

OBJECT Stream Event (speech.audio.delta)

```
1 {  
2   "type": "speech.audio.delta",  
3   "audio": "base64-encoded-audio-data"  
4 }
```

Stream Event (speech.audio.done)

Emitted when the speech synthesis is complete and all audio has been streamed.

type string

The type of the event. Always `speech.audio.done`.

usage object

Token usage statistics for the request.

OBJECT Stream Event (speech.audio.done)

```
1 {  
2   "type": "speech.audio.done",  
3   "usage": {  
4     "input_tokens": 14,  
5     "output_tokens": 101,  
6     "total_tokens": 115
```

▼ Show properties

Stream Event (transcript.text.delta)

Emitted when there is an additional text delta. This is also the first event emitted when the transcription starts. Only emitted when you create a transcription with the `Stream` parameter set to `true`.

delta string

The text delta that was additionally transcribed.

logprobs array

The log probabilities of the delta. Only included if you create a transcription with the `include[]` parameter set to `logprobs`.

▼ Show properties

type string

The type of the event. Always `transcript.text.delta`.

OBJECT Stream Event (transcript.text.delta)

```
1 {
2   "type": "transcript.text.delta",
3   "delta": " wonderful"
4 }
```

Stream Event (transcript.text.done)

Emitted when the transcription is complete. Contains the complete transcription text. Only emitted when you [create a transcription](#) with the `Stream` parameter set to `true`.

logprobs array

The log probabilities of the individual tokens in the transcription. Only included if you [create a transcription](#) with the `include[]` parameter set to `logprobs`.

▼ Show properties

text string

The text that was transcribed.

type string

The type of the event. Always `transcript.text.done`.

usage object

Usage statistics for models billed by token usage.

▼ Show properties

OBJECT Stream Event (`transcript.text.done`)

```

1  {
2    "type": "transcript.text.done",
3    "text": "I see skies of blue and clouds o
4    "usage": {
5      "type": "tokens",
6      "input_tokens": 14,
7      "input_token_details": {
8        "text_tokens": 10,
9        "audio_tokens": 4
10     },
11    "output_tokens": 31,
12    "total_tokens": 45
13  }
14 }
```

Images

Given a prompt and/or an input image, the model will generate a new image. Related guide:

[Image generation](#)

Create image

POST <https://api.openai.com/v1/images/generations>

Creates an image given a prompt. [Learn more.](#)

Request body

prompt string Required

A text description of the desired image(s). The maximum length is 32000 characters for `gpt-image-1`, 1000 characters for `dall-e-2` and 4000 characters for `dall-e-3`.

background string or null Optional Defaults to auto

Allows to set transparency for the background of the generated image(s). This parameter is only supported for `gpt-image-1`. Must be one of `transparent`, `opaque` or `auto` (default value). When `auto` is used, the model will automatically determine the best background for the image.

If `transparent`, the output format needs to support transparency, so it should be set to either `png` (default value) or `webp`.

model string Optional Defaults to dall-e-2

The model to use for image generation. One of `dall-e-2`, `dall-e-3`, or `gpt-image-1`. Defaults to `dall-e-2` unless a parameter specific to `gpt-image-1` is used.

moderation string or null Optional Defaults to auto

[Generate image](#) [Streaming](#)

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/images/generations
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -d '{
5   "model": "gpt-image-1",
6   "prompt": "A cute baby sea otter",
7   "n": 1,
8   "size": "1024x1024"
9 }'
```

Response

⌂

```
1 {
2   "created": 1713833628,
3   "data": [
4     {
5       "b64_json": "..."
6     }
7   ],
8   "usage": {
9     "total_tokens": 100,
10    "input_tokens": 50,
11    "output_tokens": 50,
12    "input_tokens_details": {
13      "text_tokens": 10,
14      "image_tokens": 40
15    }
16 }
```

Control the content-moderation level for images generated by `gpt-image-1`. Must be either `low` for less restrictive filtering or `auto` (default value).

16 }
17 }

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.

output_compression integer or null Optional Defaults to 100

The compression level (0-100%) for the generated images. This parameter is only supported for `gpt-image-1` with the `webp` or `jpeg` output formats, and defaults to 100.

output_format string or null Optional Defaults to png

The format in which the generated images are returned. This parameter is only supported for `gpt-image-1`. Must be one of `png`, `jpeg`, or `webp`.

partial_images integer or null Optional Defaults to 0

The number of partial images to generate. This parameter is used for streaming responses that return partial images. Value must be between 0 and 3. When set to 0, the response will be a single image sent in one streaming event.

Note that the final image may be sent before the full number of partial images are generated if the full image is generated more quickly.

quality string or null Optional Defaults to auto

The quality of the image that will be generated.

`auto` (default value) will automatically select the best quality for the given model.

`high`, `medium` and `low` are supported for `gpt-image-1`.

`hd` and `standard` are supported for `dall-e-3`.

`standard` is the only option for `dall-e-2`.

response_format string or null Optional Defaults to url

The format in which generated images with `dall-e-2` and `dall-e-3` are returned. Must be one of `url` or `b64_json`. URLs are only valid for 60 minutes after the image has been generated. This parameter isn't supported for `gpt-image-1` which will always return base64-encoded images.

size string or null Optional Defaults to auto

The size of the generated images. Must be one of `1024x1024`, `1536x1024` (landscape), `1024x1536` (portrait), or `auto` (default value) for `gpt-image-1`, one of `256x256`, `512x512`, or `1024x1024` for `dall-e-2`, and one of `1024x1024`, `1792x1024`, or `1024x1792` for `dall-e-3`.

stream boolean or null Optional Defaults to false

Generate the image in streaming mode. Defaults to `false`. See the [Image generation guide](#) for more information. This parameter is only supported for `gpt-image-1`.

style string or null Optional Defaults to vivid

The style of the generated images. This parameter is only supported for `dall-e-3`. Must be one of `vivid` or `natural`. Vivid causes the model to lean towards generating hyper-real and dramatic images. Natural causes the model to produce more natural, less hyper-real looking images.

user string Optional

A unique identifier representing your end-user, which can help OpenAI to monitor and detect abuse. [Learn more](#).

Returns

Returns an [image](#) object.

Create image edit

POST <https://api.openai.com/v1/images/edits>

Creates an edited or extended image given one or more source images and a prompt. This endpoint only supports `gpt-image-1` and `dall-e-2`.

Request body

image string or array Required

The image(s) to edit. Must be a supported image file or an array of images.

For `gpt-image-1`, each image should be a `png`, `webp`, or `jpg` file less than 50MB. You can provide up to 16 images.

For `dall-e-2`, you can only provide one image, and it should be a square `png` file less than 4MB.

prompt string Required

A text description of the desired image(s). The maximum length is 1000 characters for `dall-e-2`, and 32000 characters for `gpt-image-1`.

[Edit image](#) [Streaming](#)

Example request

curl ⌂

```
1 curl -s -D >(grep -i x-request-id >&2) \
2   -o >(jq -r '.data[0].b64_json' | base64 -
3   -X POST "https://api.openai.com/v1/images
4   -H "Authorization: Bearer $OPENAI_API_KEY
5   -F "model=gpt-image-1" \
6   -F "image[]=@body-lotion.png" \
7   -F "image[]=@bath-bomb.png" \
8   -F "image[]=@incense-kit.png" \
9   -F "image[]=@soap.png" \
10  -F 'prompt=Create a lovely gift basket wi
```

background string or null Optional Defaults to auto

Allows to set transparency for the background of the generated image(s). This parameter is only supported for `gpt-image-1`. Must be one of `transparent`, `opaque` or `auto` (default value). When `auto` is used, the model will automatically determine the best background for the image.

If `transparent`, the output format needs to support transparency, so it should be set to either `png` (default value) or `webp`.

input_fidelity string or null Optional Defaults to low

Control how much effort the model will exert to match the style and features, especially facial features, of input images. This parameter is only supported for `gpt-image-1`. Supports `high` and `low`. Defaults to `low`.

mask file Optional

An additional image whose fully transparent areas (e.g. where alpha is zero) indicate where `image` should be edited. If there are multiple images provided, the mask will be applied on the first image. Must be a valid PNG file, less than 4MB, and have the same dimensions as `image`.

model string Optional Defaults to dall-e-2

The model to use for image generation. Only `dall-e-2` and `gpt-image-1` are supported. Defaults to `dall-e-2` unless a parameter specific to `gpt-image-1` is used.

n integer or null Optional Defaults to 1

The number of images to generate. Must be between 1 and 10.

output_compression integer or null Optional Defaults to 100

The compression level (0-100%) for the generated images. This parameter is only supported for `gpt-image-1` with the `webp` or `jpeg` output formats, and defaults to 100.

output_format string or null Optional Defaults to png

The format in which the generated images are returned. This parameter is only supported for `gpt-image-1`. Must be one of `png`, `jpeg`, or `webp`. The default value is `png`.

partial_images integer or null Optional Defaults to 0

The number of partial images to generate. This parameter is used for streaming responses that return partial images. Value must be between 0 and 3. When set to 0, the response will be a single image sent in one streaming event.

Note that the final image may be sent before the full number of partial images are generated if the full image is generated more quickly.

quality string or null Optional Defaults to auto

The quality of the image that will be generated. `high`, `medium` and `low` are only supported for `gpt-image-1`. `dall-e-2` only supports `standard` quality. Defaults to `auto`.

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`. URLs are only valid for 60 minutes after the image has been generated. This parameter is only supported for `dall-e-2`, as `gpt-image-1` will always return base64-encoded images.

size string or null Optional Defaults to 1024x1024

The size of the generated images. Must be one of `1024x1024`, `1536x1024` (landscape), `1024x1536` (portrait), or `auto` (default value) for `gpt-image-1`, and one of `256x256`, `512x512`, or `1024x1024` for `dall-e-2`.

stream boolean or null Optional Defaults to false

Edit the image in streaming mode. Defaults to `false`. See the [Image generation guide](#) for more information.

user string Optional

A unique identifier representing your end-user, which can help OpenAI to monitor and detect abuse. [Learn more](#).

Returns

Returns an `image` object.

Create image variation

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

Creates a variation of a given image. This endpoint only supports `dall-e-2`.

Request body

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/images/variation
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -F image="@otter.png" \
4 -F n=2 \
5 -F size="1024x1024"
```

image file Required

The image to use as the basis for the variation(s). Must be a valid PNG file, less than 4MB, and square.

model string or "dall-e-2" Optional Defaults to dall-e-2

The model to use for image generation. Only `dall-e-2` is supported at this time.

n integer or null Optional Defaults to 1

The number of images to generate. Must be between 1 and 10.

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`. URLs are only valid for 60 minutes after the image has been generated.

size string or null Optional Defaults to 1024x1024

The size of the generated images. Must be one of `256x256`, `512x512`, or `1024x1024`.

user string Optional

A unique identifier representing your end-user, which can help OpenAI to monitor and detect abuse. [Learn more](#).

Response

```
1  {
2    "created": 1589478378,
3    "data": [
4      {
5        "url": "https://..."
6      },
7      {
8        "url": "https://..."
9      }
10   ]
11 }
```

Returns

Returns a list of `image` objects.

The image generation response

The response from the image generation endpoint.

background string

The background parameter used for the image generation. Either `transparent` or `opaque`.

created integer

The Unix timestamp (in seconds) of when the image was created.

data array

The list of generated images.

✓ Show properties

output_format string

The output format of the image generation. Either `png`, `webp`, or `jpeg`.

quality string

The quality of the image generated. Either `low`, `medium`, or `high`.

size string

The size of the image generated. Either `1024x1024`, `1024x1536`, or `1536x1024`.

usage object

For `gpt-image-1` only, the token usage information for the image generation.

✓ Show properties

OBJECT The image generation response

```

1  {
2    "created": 1713833628,
3    "data": [
4      {
5        "b64_json": "..."
6      }
7    ],
8    "background": "transparent",
9    "output_format": "png",
10   "size": "1024x1024",
11   "quality": "high",
12   "usage": {
13     "total_tokens": 100,
14     "input_tokens": 50,
15     "output_tokens": 50,
16     "input_tokens_details": {
17       "text_tokens": 10,
18       "image_tokens": 40
19     }
20   }
21 }
```

Image Streaming

Stream image generation and editing in real time with server-sent events.

[Learn more about image streaming.](#)

image_generation.partial_image

Emitted when a partial image is available during image generation streaming.

b64_json string

Base64-encoded partial image data, suitable for rendering as an image.

background string

The background setting for the requested image.

created_at integer

The Unix timestamp when the event was created.

output_format string

The output format for the requested image.

partial_image_index integer

OBJECT image_generation.partial_image

```
1  {
2    "type": "image_generation.partial_image",
3    "b64_json": "...",
4    "created_at": 1620000000,
5    "size": "1024x1024",
6    "quality": "high",
7    "background": "transparent",
8    "output_format": "png",
9    "partial_image_index": 0
10 }
```

0-based index for the partial image (streaming).

quality string

The quality setting for the requested image.

size string

The size of the requested image.

type string

The type of the event. Always `image_generation.partial_image`.

image_generation.completed

Emitted when image generation has completed and the final image is available.

b64_json string

Base64-encoded image data, suitable for rendering as an image.

background string

The background setting for the generated image.

created_at integer

The Unix timestamp when the event was created.

output_formatOBJECT `image_generation.completed`

```
1  {
2    "type": "image_generation.completed",
3    "b64_json": "...",
4    "created_at": 1620000000,
5    "size": "1024x1024",
6    "quality": "high",
7    "background": "transparent",
8    "output_format": "png",
9    "usage": {
10      "total_tokens": 100,
11      "input_tokens": 50,
12      "output_tokens": 50,
13      "input_tokens_details": {
```

string
The output format for the generated image.

quality **string**
The quality setting for the generated image.

size **string**
The size of the generated image.

type **string**
The type of the event. Always `image_generation.completed`.

usage **object**
For `gpt-image-1` only, the token usage information for the image generation.

▼ Show properties

```
14     "text_tokens": 10,
15     "image_tokens": 40
16   }
17 }
18 }
```

image_edit.partial_image

Emitted when a partial image is available during image editing streaming.

b64_json **string**
Base64-encoded partial image data, suitable for rendering as an image.

OBJECT `image_edit.partial_image`

```
1  {
2    "type": "image_edit.partial_image",
3    "b64_json": "...",
4    "created_at": 1620000000,
```

background string

The background setting for the requested edited image.

created_at integer

The Unix timestamp when the event was created.

output_format string

The output format for the requested edited image.

partial_image_index integer

0-based index for the partial image (streaming).

quality string

The quality setting for the requested edited image.

size string

The size of the requested edited image.

type string

The type of the event. Always `image_edit.partial_image`.

```
5   "size": "1024x1024",
6   "quality": "high",
7   "background": "transparent",
8   "output_format": "png",
9   "partial_image_index": 0
10 }
```

image_edit.completed

Emitted when image editing has completed and the final image is available.

OBJECT `image_edit.completed`



b64_json string

Base64-encoded final edited image data, suitable for rendering as an image.

background string

The background setting for the edited image.

created_at integer

The Unix timestamp when the event was created.

output_format string

The output format for the edited image.

quality string

The quality setting for the edited image.

size string

The size of the edited image.

type string

The type of the event. Always `image_edit.completed`.

usage object

For `gpt-image-1` only, the token usage information for the image generation.

▼ Show properties

```
1  {
2    "type": "image_edit.completed",
3    "b64_json": "...",
4    "created_at": 1620000000,
5    "size": "1024x1024",
6    "quality": "high",
7    "background": "transparent",
8    "output_format": "png",
9    "usage": {
10      "total_tokens": 100,
11      "input_tokens": 50,
12      "output_tokens": 50,
13      "input_tokens_details": {
14        "text_tokens": 10,
15        "image_tokens": 40
16      }
17    }
```

Embeddings

Get a vector representation of a given input that can be easily consumed by machine learning models and algorithms. Related guide: [Embeddings](#)

Create embeddings

POST <https://api.openai.com/v1/embeddings>

Creates an embedding vector representing the input text.

Request body

input string or array Required

Input text to embed, encoded as a string or array of tokens. To embed multiple inputs in a single request, pass an array of strings or array of token arrays. The input must not exceed the max input tokens for the model (8192 tokens for all embedding models), cannot be an empty string, and any array must be 2048 dimensions or less.

[Example Python code](#) for counting tokens. In addition to the per-input token limit, all embedding models enforce a maximum of 300,000 tokens summed across all inputs in a single request.

model string Required

ID of the model to use. You can use the [List models](#) API to see all of your available models, or see our [Model overview](#) for descriptions of them.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/embeddings \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -d '{
5     "input": "The food was delicious and the
6     "model": "text-embedding-ada-002",
7     "encoding_format": "float"
8   }'
```

Response

⌚

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "object": "embedding",
6       "embedding": [
7         0.0023064255,
8         -0.009327292,
9         .... (1536 floats total for ada-002
```

dimensions integer Optional

The number of dimensions the resulting output embeddings should have. Only supported in `text-embedding-3` and later models.

encoding_format string Optional Defaults to float

The format to return the embeddings in. Can be either `float` or `base64`.

user string Optional

A unique identifier representing your end-user, which can help OpenAI to monitor and detect abuse. [Learn more](#).

```

10      -0.0028842222,
11    ],
12    "index": 0
13  }
14  ],
15  "model": "text-embedding-ada-002",
16  "usage": {
17    "prompt_tokens": 8,
18    "total_tokens": 8
19  }
20 }
```

Returns

A list of [embedding](#) objects.

The embedding object

Represents an embedding vector returned by embedding endpoint.

embedding array

The embedding vector, which is a list of floats. The length of vector depends on the model as listed in the [embedding guide](#).

index integer

OBJECT The embedding object

```

1  {
2    "object": "embedding",
3    "embedding": [
4      0.0023064255,
5      -0.009327292,
6      .... (1536 floats total for ada-002)
```

The index of the embedding in the list of embeddings.

object string

The object type, which is always "embedding".

```

7      -0.0028842222,
8      ],
9      "index": 0
10 }
```

Evals

Create, manage, and run evals in the OpenAI platform. Related guide: [Evals](#)

Create eval

POST <https://api.openai.com/v1/evals>

Create the structure of an evaluation that can be used to test a model's performance. An evaluation is a set of testing criteria and the config for a data source, which dictates the schema of the data used in the evaluation. After creating an evaluation, you can run it on different models and model parameters. We support several types of graders and datasources. For more information, see the [Evals guide](#).

Request body

data_source_config object Required

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/evals \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -d '{
5     "name": "Sentiment",
6     "data_source_config": {
7       "type": "stored_completions",
8       "metadata": {
9         "usecase": "chatbot"
10       }
11     },
12     "testing_criteria": [
```

The configuration for the data source used for the evaluation runs. Dictates the schema of the data used in the evaluation.

✓ Show possible types

testing_criteria array Required

A list of graders for all eval runs in this group. Graders can reference variables in the data source using double curly braces notation, like `{{item.variable_name}}`. To reference the model's output, use the `sample` namespace (ie, `{{sample.output_text}}`).

✓ Show possible types

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string Optional

The name of the evaluation.

Returns

The created `Eval` object.

```

13    {
14      "type": "label_model",
15      "model": "o3-mini",
16      "input": [
17        {
18          "role": "developer",
19          "content": "Classify the se
20        },
21        {
22          "role": "user",
23          "content": "Statement: {{it
24        }
25      ],
26      "passing_labels": [
27        "positive"
28      ],
29      "labels": [
30        "positive",
31        "neutral",
32        "negative"
33      ],
34      "name": "Example label grader"
35    }
36  ]
37 }'
```

Response

```

1  {
2    "object": "eval",
3    "id": "eval_67b7fa9a81a88190ab4aa417e397e
4    "data_source_config": {
5      "type": "stored_completions",
6      "metadata": {
7        "usecase": "chatbot"
```

Get an eval

GET https://api.openai.com/v1/evals/{eval_id}

Get an evaluation by ID.

Path parameters

eval_id string Required

The ID of the evaluation to retrieve.

Returns

The [Eval](#) object matching the specified ID.

```

8   },
9   "schema": {
10    "type": "object".

```

Example request

curl ⚡

```

1 curl https://api.openai.com/v1/evals/eval_67
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json"

```

Response

```

1  {
2   "object": "eval",
3   "id": "eval_67abd54d9b0081909a86353f6fb93
4   "data_source_config": {
5     "type": "custom",
6     "schema": {
7       "type": "object",
8       "properties": {
9         "item": {
10          "type": "object",
11          "properties": {
12            "input": {
13              "type": "string"
14            },
15            "ground_truth": {
16              "type": "string"
17            }
18          },
19          "required": [
20            "input",
21            "ground_truth"
22          ]
23        }

```

```

24     },
25     "required": [
26       "item"
27     ]
28   },
29 },
30 "testing_criteria": [

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/evals/eval_67
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -d '{"name": "Updated Eval", "metadata": {

```

Response

⌂

```

1 {
2   "object": "eval",
3   "id": "eval_67abd54d9b0081909a86353f6fb93",
4   "data_source_config": {
5     "type": "custom",
6     "schema": {
7       "type": "object",
8       "properties": {
9         "item": {
10           "type": "object",
11           "properties": {
12             "input": {
13               "type": "string"
14             },
15             "ground_truth": {
16               "type": "string"
17             }
18           },

```

Update an eval

POST https://api.openai.com/v1/evals/{eval_id}

Update certain properties of an evaluation.

Path parameters

eval_id string **Required**

The ID of the evaluation to update.

Request body

metadata map **Optional**

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string **Optional**

Rename the evaluation.

Returns

The [Eval](#) object matching the updated version.

Delete an eval

```
DELETE https://api.openai.com/v1/evals/{eval_id}
```

Delete an evaluation.

Path parameters

eval_id string Required

The ID of the evaluation to delete.

Returns

A deletion confirmation object.

```

19     "required": [
20       "input",
21       "ground_truth"
22     ]
23   },
24   "required": [
25     "item"
26   ]
27 }
28 }
29 }
```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/evals/eval_ab
2 -X DELETE \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
36   "reference": "{$item.ground_truth}"
```

Response

```

1 {
2   "object": "eval.deleted",
3   "deleted": true,
4   "eval_id": "eval_abc123"
5 }
```

List evals

GET <https://api.openai.com/v1/evals>

List evaluations for a project.

Query parameters

after string Optional

Identifier for the last eval from the previous pagination request.

limit integer Optional Defaults to 20

Number of evals to retrieve.

order string Optional Defaults to asc

Sort order for evals by timestamp. Use `asc` for ascending order or `desc` for descending order.

order_by string Optional Defaults to `created_at`

Evals can be ordered by creation time or last updated time. Use `created_at` for creation time or `updated_at` for last updated time.

Returns

A list of [evals](#) matching the specified filters.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/evals?limit=1
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "eval_67abd54d9b0081909a86353f6",
6       "object": "eval",
7       "data_source_config": {
8         "type": "stored_completions",
9         "metadata": {
10           "usecase": "push_notifications_su
11         },
12         "schema": {
13           "type": "object",
14           "properties": {
15             "item": {
16               "type": "object"
17             },
18             "sample": {
19               "type": "object"
20             }
21           },
22           "required": [
23             "item",
24             "sample"
25           ]
26         }
27     }
28   ]
29 }
```

Get eval runs

GET https://api.openai.com/v1/evals/{eval_id}/runs

Get a list of runs for an evaluation.

Path parameters

eval_id string **Required**

The ID of the evaluation to retrieve runs for.

Query parameters

after string **Optional**

Identifier for the last run from the previous pagination request.

limit integer **Optional** Defaults to 20

Number of runs to retrieve.

order string **Optional** Defaults to asc

Sort order for runs by timestamp. Use `asc` for ascending order or `desc` for descending order. Defaults to `asc`.

status string **Optional**

Filter runs by status. One of `queued` | `in_progress` | `failed` | `completed` | `canceled`.

```
27 },
28 "testing_criteria": [
29 {
```

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/evals/egroup_
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json"
```

26 "type": "message"

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "object": "eval.run",
6       "id": "evalrun_67e0c7d31560819090d60c",
7       "eval_id": "eval_67e0c726d560819083f1",
8       "report_url": "https://platform.openai.com/r/eval_67e0c726d560819083f1",
9       "status": "completed",
10      "model": "o3-mini",
11      "name": "bulk_with_negative_examples",
12      "created_at": 1742784467,
13      "result_counts": {
14        "total": 1,
15        "errored": 0,
16        "failed": 0,
17        "passed": 1
18      },
19      "per_model_usage": [
20        {
21          "model_name": "o3-mini",
22          "invocation_count": 1,
23          "prompt_tokens": 563,
```

Returns

A list of [EvalRun](#) objects matching the specified ID.

Get an eval run

```
GET https://api.openai.com/v1/evals/{eval_id}/runs/{run_id}
```

Get an evaluation run by ID.

Path parameters

eval_id string Required

The ID of the evaluation to retrieve runs for.

run_id string Required

The ID of the run to retrieve.

Returns

The [EvalRun](#) object matching the specified ID.

```

24     "completion_tokens": 874,
25     "total_tokens": 1437,
26     "cached_tokens": 0
27   }
28 ],
29   "per_testing_criteria_results": [
30     {
31       "testing_criteria": "Push Notific
32       "passed": 1,
33       "failed": 0
34     }
35   ],

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/evals/eval_67
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json"
42     "item": {

```

Response

```

1   {
2     "object": "eval.run",
3     "id": "evalrun_67abd54d60ec8190832b46859",
4     "eval_id": "eval_67abd54d9b0081909a86353",
5     "report_url": "https://platform.openai.c
6     "status": "queued",
7     "model": "gpt-4o-mini",
8     "name": "gpt-4o-mini",
9     "created_at": 1743092069,
10    "result_counts": {
11      "total": 0,
12      "errored": 0,
13      "failed": 0,
14      "passed": 0

```

```

15 },
16 "per_model_usage": null,
17 "per_testing_criteria_results": null,
18 "data_source": {
19   "type": "completions",
20   "source": {
21     "type": "file_content",
22     "content": [
23       {
24         "item": {
25           "input": "Tech Company Launche",
26           "ground_truth": "Technology"
27         }
28       },
29     ]
30   }
31 }
```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/evals/eval_67
2   -X POST \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
4   -H "Content-Type: application/json" \
5   -d '{"name":"gpt-4o-mini","data_source":{}}
```

Response ⌂

```

1 {
2   "object": "eval.run",
3   "id": "evalrun_67e57965b480819094274e3a32",
4   "eval_id": "eval_67e57965b548190aaa83ada",
5   "report_url": "https://platform.openai.co",
6   "status": "queued",
7   "model": "gpt-4o-mini",
8   "name": "gpt-4o-mini",
9   "created_at": 1743092069,
10  "result_counts": {
```

Create eval run

POST https://api.openai.com/v1/evals/{eval_id}/runs

Kicks off a new run for a given evaluation, specifying the data source, and what model configuration to use to test. The datasource will be validated against the schema specified in the config of the evaluation.

Path parameters

eval_id string Required

The ID of the evaluation to create a run for.

Request body

data_source object Required

Details about the run's data source.

✓ Show possible types

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string Optional

The name of the run.

Returns

The [EvalRun](#) object matching the specified ID.

Cancel eval run

POST https://api.openai.com/v1/evals/{eval_id}/runs/{run_id}

Cancel an ongoing evaluation run.

Path parameters

```

11 "total": 0,
12 "errored": 0,
13 "failed": 0,
14 "passed": 0
15 },
16 "per_model_usage": null,
17 "per_testing_criteria_results": null,
18 "data_source": {
19   "type": "completions",
20   "source": {
21     "type": "file_content",
22     "content": [
23       {
24         "item": {
25           "input": "Tech Company Launches"
26           "ground_truth": "Technology"
27         }
28       }
29     ]
30   },
31   "input_messages": {
32     "type": "template",
33     "template": [
34       {
35         "type": "message",
36         "role": "developer",
37         "content": {
38           "type": "input_text"
39         }
40       }
41     ]
42   }
43 }
```

Example request

curl ⚡

```

1 curl https://api.openai.com/v1/evals/eval_67
2 -X POST \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -H "Content-Type: application/json"
```

eval_id string Required

The ID of the evaluation whose run you want to cancel.

run_id string Required

The ID of the run to cancel.

Returns

The updated [EvalRun](#) object reflecting that the run is canceled.

Response

```

1  {
2    "object": "eval.run",
3    "id": "evalrun_67abd54d60ec8190832b46859",
4    "eval_id": "eval_67abd54d9b0081909a86353",
5    "report_url": "https://platform.openai.c",
6    "status": "canceled",
7    "model": "gpt-4o-mini",
8    "name": "gpt-4o-mini",
9    "created_at": 1743092069,
10   "result_counts": {
11     "total": 0,
12     "errored": 0,
13     "failed": 0,
14     "passed": 0
15   },
16   "per_model_usage": null,
17   "per_testing_criteria_results": null,
18   "data_source": {
19     "type": "completions",
20     "source": {
21       "type": "file_content",
22       "content": [
23         {
24           "item": {
25             "input": "Tech Company Launche",
26             "ground_truth": "Technology"
27           }
28         }
29       ]
30     }
31   }
32 }
```

Example request

curl : {

```

1 curl https://api.openai.com/v1/evals/eval_12
2   -X DELETE \
```

Delete eval run

`DELETE https://api.openai.com/v1/evals/{eval_id}/runs/{run_id}`

Delete an eval run.

Path parameters

eval_id string Required

The ID of the evaluation to delete the run from.

run_id string Required

The ID of the run to delete.

Returns

An object containing the status of the delete operation.

Get an output item of an eval run

```
GET https://api.openai.com/v1/evals/{eval_id}/runs/{run_id}/output_items/{output_item_id}
```

Get an evaluation run output item by ID.

Path parameters

eval_id string Required

The ID of the evaluation to retrieve runs for.

```
3   -H "Authorization: Bearer $OPENAI_API_KEY"
35
36   {
37     "object": "eval.run.deleted",
38     "deleted": true,
39     "run_id": "evalrun_abc456"
40   }
```

```
44   "ground_truth": "Business"
45 }
46 },
47 {
48   "item": {
49     "input": "National Team Qualif
50     "ground_truth": "Sports"
51   }
52 },
53 {
54   "item": {
55     "input": "Stock Markets Rally"
56     "ground_truth": "Markets"
```

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/evals/eval_67
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json"
```

Response

```
1   {
2     "object": "eval.run.output_item",
3     "id": "outputitem_67e5796c28e081909917bf7"
```

output_item_id string Required

The ID of the output item to retrieve.

run_id string Required

The ID of the run to retrieve.

Returns

The [EvalRunOutputItem](#) object matching the specified ID.

Get eval run output items

```
GET https://api.openai.com/v1/evals/{eval_id}/runs/{run_id}/output_items
```

Get a list of output items for an evaluation run.

Path parameters**eval_id** string Required

```

4   "created_at": 1743092076,
5   "run_id": "evalrun_67abd54d60ec8190832b46
6   "eval_id": "eval_67abd54d9b0081909a86353f
7   "status": "pass",
8   "datasource_item_id": 5,
9   "datasource_item": {
10    "input": "Stock Markets Rally After Pos
11    "ground_truth": "Markets"
12  },
13  "results": [
14    {
15      "name": "String check-a2486074-d803-4
16      "sample": null,
17      "passed": true,
18      "score": 1.0
19    }
20  ],
21  "sample": {
22    "input": [
23      {
24        "role": "developer",
25        "content": "Categorize a given news
26        "tool_call_id": null,
27        "tool_calls": null,
28        "function_call": null
29      }
30    ]
31  }
32 }
```

Example request
curl - D

```

1 curl https://api.openai.com/v1/evals/egroup_
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json"
4
5   "function_call": null
6 }
```

Response
D

The ID of the evaluation to retrieve runs for.

run_id string Required

The ID of the run to retrieve output items for.

Query parameters

after string Optional

Identifier for the last output item from the previous pagination request.

limit integer Optional Defaults to 20

Number of output items to retrieve.

order string Optional Defaults to asc

Sort order for output items by timestamp. Use `asc` for ascending order or `desc` for descending order. Defaults to `asc`.

status string Optional

Filter output items by status. Use `failed` to filter by failed output items or `pass` to filter by passed output items.

Returns

A list of `EvalRunOutputItem` objects matching the specified ID.

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "eval.run.output_item",
6        "id": "outputitem_67e5796c28e08190991",
7        "created_at": 1743092076,
8        "run_id": "evalrun_67abd54d60ec819083",
9        "eval_id": "eval_67abd54d9b0081909a86",
10       "status": "pass",
11       "datasource_item_id": 5,
12       "datasource_item": {
13         "input": "Stock Markets Rally After",
14         "ground_truth": "Markets"
15       },
16       "results": [
17         {
18           "name": "String check-a2486074-d8",
19           "sample": null,
20           "passed": true,
21           "score": 1.0
22         }
23       ],
24       "sample": {
25         "input": [
26           {
27             "role": "developer",
28             "content": "Categorize a given",
29             "tool_call_id": null,
30             "tool_calls": null,
31             "function_call": null
32           },
33           {
34             "role": "user",
35             "content": "Stock Markets Rally"
36           }
37         ]
38       }
39     ]
40   }
41 
```

The eval object

An Eval object with a data source config and testing criteria. An Eval represents a task to be done for your LLM integration. Like:

Improve the quality of my chatbot

See how well my chatbot handles customer support

Check if o4-mini is better at my usecase than gpt-4o

created_at integer

The Unix timestamp (in seconds) for when the eval was created.

data_source_config object

Configuration of data sources used in runs of the evaluation.

✓ Show possible types

id string

Unique identifier for the evaluation.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string

The name of the evaluation.

```

36         "tool_call_id": null,
37         "tool_calls": null,
38         "function_call": null

```

OBJECT The eval object

```

1  {
2   "object": "eval",
3   "id": "eval_67abd54d9b0081909a86353f6fb93
4   "data_source_config": {
5     "type": "custom",
6     "item_schema": {
7       "type": "object",
8       "properties": {
9         "label": {"type": "string"},
10      },
11      "required": ["label"]
12    },
13    "include_sample_schema": true
14  },
15  "testing_criteria": [
16    {
17      "name": "My string check grader",
18      "type": "string_check",
19      "input": "{{sample.output_text}}",
20      "reference": "{{item.label}}",
21      "operation": "eq",
22    }
23  ],
24  "name": "External Data Eval",
25  "created_at": 1739314509,
26  "metadata": {
27    "test": "synthetics",
28  }
29 }

```

object string

The object type.

testing_criteria array

A list of testing criteria.

▼ Show possible types

The eval run object

A schema representing an evaluation run.

created_at integer

Unix timestamp (in seconds) when the evaluation run was created.

data_source object

Information about the run's data source.

▼ Show possible types

error object

An object representing an error response from the Eval API.

▼ Show properties

eval_id string

The identifier of the associated evaluation.

OBJECT The eval run object

```
1  {
2    "object": "eval.run",
3    "id": "evalrun_67e57965b480819094274e3a3",
4    "eval_id": "eval_67e579652b548190aaa83ad",
5    "report_url": "https://platform.openai.c",
6    "status": "queued",
7    "model": "gpt-4o-mini",
8    "name": "gpt-4o-mini",
9    "created_at": 1743092069,
10   "result_counts": {
11     "total": 0,
12     "errored": 0,
13     "failed": 0,
14     "passed": 0
15   },
16   "per_model_usage": null,
17   "per_testing_criteria_results": null,
```

id string

Unique identifier for the evaluation run.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string

The model that is evaluated, if applicable.

name string

The name of the evaluation run.

object string

The type of the object. Always "eval.run".

per_model_usage array

Usage statistics for each model during the evaluation run.

▼ Show properties

per_testing_criteria_results array

Results per testing criteria applied during the evaluation run.

▼ Show properties

report_url string

```

18 "data_source": {
19   "type": "completions",
20   "source": {
21     "type": "file_content",
22     "content": [
23       {
24         "item": {
25           "input": "Tech Company Launche
26           "ground_truth": "Technology"
27         }
28       },
29       {
30         "item": {
31           "input": "Central Bank Increases
32           "ground_truth": "Markets"
33         }
34       },
35       {
36         "item": {
37           "input": "International Summit
38           "ground_truth": "World"
39         }
40       },
41       {
42         "item": {
43           "input": "Major Retailer Repor
44           "ground_truth": "Business"
45         }
46       },
47       {
48         "item": {
49           "input": "National Team Qualif
50           "ground_truth": "Sports"
51         }
52       },
53     ]
54   }
55 }
```

The URL to the rendered evaluation run report on the UI dashboard.

result_counts object

Counters summarizing the outcomes of the evaluation run.

▼ Show properties

status string

The status of the evaluation run.

The eval run output item object

A schema representing an evaluation run output item.

created_at integer

Unix timestamp (in seconds) when the evaluation run was created.

datasource_item object

Details of the input data source item.

datasource_item_id integer

The identifier for the data source item.

eval_id string

The identifier of the evaluation group.

```

53     {
54         "item": {
55             "input": "Stock Markets Rally",
56             "ground_truth": "Markets"
57         }
58     },
59     {
60         "item": {
61             "input": "Global Manufacturer",
62             "ground_truth": "Business"
63         }
64     },
65     {
66         "item": {
67             "input": "Breakthrough in Renewable Energy",
68             "ground_truth": "Technology"
69         }
70     }

```

OBJECT The eval run output item object

```

1  {
2      "object": "eval.run.output_item",
3      "id": "outputitem_67abd55eb6548190bb58074",
4      "run_id": "evalrun_67abd54d60ec8190832b46",
5      "eval_id": "eval_67abd54d9b0081909a86353f",
6      "created_at": 1739314509,
7      "status": "pass",
8      "datasource_item_id": 137,
9      "datasource_item": {
10         "teacher": "To grade essays, I only consider the most common words in the student's response.",
11         "student": "I am a student who is trying to learn English grammar and punctuation rules."
12     },
13     "results": [
14         {
15             "name": "String Check Grader",

```

id string

Unique identifier for the evaluation run output item.

object string

The type of the object. Always "eval.run.output_item".

results array

A list of results from the evaluation run.

✓ Show properties

run_id string

The identifier of the evaluation run associated with this output item.

sample object

A sample containing the input and output of the evaluation run.

✓ Show properties

status string

The status of the evaluation run.

Fine-tuning

Manage fine-tuning jobs to tailor a model to your specific training data. Related:

```

16     "type": "string-check-grader",
17     "score": 1.0,
18     "passed": true,
19   },
20 ],
21 "sample": {
22   "input": [
23     {
24       "role": "system",
25       "content": "You are an evaluator bo
26     },
27     {
28       "role": "user",
29       "content": "You are assessing..."
30     }
31   ],
32   "output": [
33     {
34       "role": "assistant",
35       "content": "The rubric is not clear
36     }
37   ],
38   "finish_reason": "stop",
39   "model": "gpt-4o-2024-08-06",
40   "usage": {
41     "total_tokens": 521,
42     "completion_tokens": 2,
43     "prompt_tokens": 519,
44     "cached_tokens": 0
45   },
46   "error": null,
47   "temperature": 1.0,
48   "max_completion_tokens": 2048,
49   "top_p": 1.0,
50   "seed": 42

```

Create fine-tuning job

POST https://api.openai.com/v1/fine_tuning/jobs

Creates a fine-tuning job which begins the process of creating a new model from a given dataset.

Response includes details of the enqueued job including job status and the name of the fine-tuned models once complete.

[Learn more about fine-tuning](#)

Request body

model string Required

The name of the model to fine-tune. You can select one of the [supported models](#).

training_file string Required

The ID of an uploaded file that contains training data.

See [upload file](#) for how to upload a file.

Your dataset must be formatted as a JSONL file. Additionally, you must upload your file with the purpose `fine-tune`.

The contents of the file should differ depending on if the model uses the [chat_completions](#) format, or if the fine-tuning method uses the [preference](#) format.

See the [fine-tuning guide](#) for more details.

hyperparameters Deprecated object Optional

The hyperparameters used for the fine-tuning job. This value is now deprecated in favor of `method`, and should be passed in under the `method` parameter.

```

51   }
124 }
125 }.
Default Epochs DPO Reinforcement Validation
127     "type": "message",

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/fine_tuning/j
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -d '{
5   "training_file": "file-BK7bzQj3FfZFXr7Db
6   "model": "gpt-4o-mini"
7 }'

```

138 "seed": 42.

Response

```

1 {
2   "object": "fine_tuning.job",
3   "id": "ftjob-abc123",
4   "model": "gpt-4o-mini-2024-07-18",
5   "created_at": 1721764800,
6   "fine_tuned_model": null,
7   "organization_id": "org-123",
8   "result_files": [],
9   "status": "queued",
10  "validation_file": null,
11  "training_file": "file-abc123",
12  "method": {
13    "type": "supervised",
14    "supervised": {
15      "hyperparameters": {
16        "batch_size": "auto",
17        "learning_rate_multiplier": "auto",

```

✓ Show properties

integrations array or null Optional

A list of integrations to enable for your fine-tuning job.

✓ Show properties

```
18     "n_epochs": "auto",
19   }
20   }
21 },
22 "metadata": null
23 }
```

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

method object Optional

The method used for fine-tuning.

✓ Show properties

seed integer or null Optional

The seed controls the reproducibility of the job. Passing in the same seed and job parameters should produce the same results, but may differ in rare cases. If a seed is not specified, one will be generated for you.

suffix string or null Optional Defaults to null

A string of up to 64 characters that will be added to your fine-tuned model name.

For example, a `suffix` of "custom-model-name" would produce a model name like

`ft:gpt-4o-mini:openai:custom-model-name:7p4lURel`.

validation_file string or null Optional

The ID of an uploaded file that contains validation data.

If you provide this file, the data is used to generate validation metrics periodically during fine-tuning. These metrics can be viewed in the fine-tuning results file. The same data should not be present in both train and validation files.

Your dataset must be formatted as a JSONL file. You must upload your file with the purpose `fine-tune`.

See the [fine-tuning guide](#) for more details.

Returns

A [fine-tuning.job](#) object.

List fine-tuning jobs

GET https://api.openai.com/v1/fine_tuning/jobs

List your organization's fine-tuning jobs

Query parameters

after string Optional

Identifier for the last job from the previous pagination request.

limit integer Optional Defaults to 20

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/fine_tuning/jobs  
2 -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {  
2   "object": "list",  
3   "data": [  
4     {  
5       "object": "fine_tuning.job",  
6       "id": "ft_1234567890",  
7       "model": "text-davinci-003",  
8       "status": "succeeded",  
9       "created": 1628000000,  
10      "modified": 1628000000,  
11      "size_gb": 1.2,  
12      "billed_size_gb": 1.2,  
13      "cost": 0.012,  
14      "usage": {  
15        "prompt_tokens": 1000,  
16        "completion_tokens": 1000,  
17        "total_tokens": 2000  
18      },  
19      "log_file": "ft_1234567890.log",  
20      "error": null  
21    }  
22  ]  
23}
```

Number of fine-tuning jobs to retrieve.

metadata object or null Optional

Optional metadata filter. To filter, use the syntax `metadata[k]=v`. Alternatively, set `metadata=null` to indicate no metadata.

Returns

A list of paginated [fine-tuning job](#) objects.

```

6     "id": "ftjob-abc123",
7     "model": "gpt-4o-mini-2024-07-18",
8     "created_at": 1721764800,
9     "fine_tuned_model": null,
10    "organization_id": "org-123",
11    "result_files": [],
12    "status": "queued",
13    "validation_file": null,
14    "training_file": "file-abc123",
15    "metadata": {
16      "key": "value"
17    }
18  },
19  { ... },
20  { ... }
21 ], "has_more": true
22 }
```

List fine-tuning events

```
GET https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}/events
```

Get status updates for a fine-tuning job.

Path parameters

fine_tuning_job_id string Required

The ID of the fine-tuning job to get events for.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/fine_tuning/j
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```

1  {
2    "object": "list",
3    "data": [
4      {
```

Query parameters

after string Optional

Identifier for the last event from the previous pagination request.

limit integer Optional Defaults to 20

Number of events to retrieve.

Returns

A list of fine-tuning event objects.

```

5      "object": "fine_tuning.job.event",
6      "id": "ft-event-ddTJfwuMVpfLXse00Am0G
7      "created_at": 1721764800,
8      "level": "info",
9      "message": "Fine tuning job successfu
10     "data": null,
11     "type": "message"
12   },
13   {
14     "object": "fine_tuning.job.event",
15     "id": "ft-event-tyiGuB72evQncpH87xe50
16     "created_at": 1721764800,
17     "level": "info",
18     "message": "New fine-tuned model crea
19     "data": null,
20     "type": "message"
21   }
22 ],
23   "has_more": true
24 }
```

List fine-tuning checkpoints

```
GET https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}/checkpoints
```

List checkpoints for a fine-tuning job.

Path parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/fine_tuning/j
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

fine_tuning_job_id string Required

The ID of the fine-tuning job to get checkpoints for.

Query parameters**after** string Optional

Identifier for the last checkpoint ID from the previous pagination request.

limit integer Optional Defaults to 10

Number of checkpoints to retrieve.

Returns

A list of fine-tuning [checkpoint objects](#) for a fine-tuning job.

List checkpoint permissions

```
GET https://api.openai.com/v1/fine_tuning/checkpoints/{fine_tuned_model_checkpoint}/permissions
```

NOTE: This endpoint requires an [admin API key](#).

Organization owners can use this endpoint to view all permissions for a fine-tuned model checkpoint.

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "fine_tuning.job.checkpoint",
6        "id": "ftckpt_zc4Q7MP6XxulcVzj4MZdwsA",
7        "created_at": 1721764867,
8        "fine_tuned_model_checkpoint": "ft:gp",
9        "metrics": {
10          "full_valid_loss": 0.134,
11          "full_valid_mean_token_accuracy": 0
12        },
13        "fine_tuning_job_id": "ftjob-abc123",
14        "step_number": 2000
15      },
16      {
17        "object": "fine_tuning.job.checkpoint",
18        "id": "ftckpt_enQCFmOTGj3syEpYVhBRLTS",
19        "created_at": 1721764800,
20        "fine_tuned_model_checkpoint": "ft:gp",
21        "metrics": {
22          "full_valid_loss": 0.167,
23          "full_valid_mean_token_accuracy": 0
24        },
25        "fine_tuning_job_id": "ftjob-abc123",
26        "step_number": 1000
27      }
28    ],

```

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/fine_tuning/c
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Path parameters

fine_tuned_model_checkpoint string Required

The ID of the fine-tuned model checkpoint to get permissions for.

Query parameters

after string Optional

Identifier for the last permission ID from the previous pagination request.

limit integer Optional Defaults to 10

Number of permissions to retrieve.

order string Optional Defaults to descending

The order in which to retrieve permissions.

project_id string Optional

The ID of the project to get permissions for.

Response

```
1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "checkpoint.permission",
6        "id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",
7        "created_at": 1721764867,
8        "project_id": "proj_abGMw1llN8IrBb6Sv"
9      },
10     {
11       "object": "checkpoint.permission",
12       "id": "cp_enQCFmOTGj3syEpYVhBRLTSy",
13       "created_at": 1721764800,
14       "project_id": "proj_iqGMw1llN8IrBb6Sv"
15     },
16   ],
17   "first_id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",
18   "last_id": "cp_enQCFmOTGj3syEpYVhBRLTSy",
19   "has_more": false
```

Returns

A list of fine-tuned model checkpoint [permission objects](#) for a fine-tuned model checkpoint.

Create checkpoint permissions

```
POST https://api.openai.com/v1/fine_tuning/checkpoints/{fine_tuned_model_checkpoint}/permissions
```

NOTE: Calling this endpoint requires an [admin API key](#).

This enables organization owners to share fine-tuned models with other projects in their organization.

Path parameters

fine_tuned_model_checkpoint string Required

The ID of the fine-tuned model checkpoint to create a permission for.

Request body

project_ids array Required

The project identifiers to grant access to.

Returns

A list of fine-tuned model checkpoint [permission objects](#) for a fine-tuned model checkpoint.

Example request

curl ⚡ ↗

```
1 curl https://api.openai.com/v1/fine_tuning/c
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -d '{"project_ids": ["proj_abGMw1llN8IrBb6Sv
4     ]}'
```

Response

↗

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "object": "checkpoint.permission",
6       "id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",
7       "created_at": 1721764867,
8       "project_id": "proj_abGMw1llN8IrBb6Sv
9     }
10   ],
11   "first_id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",
12   "last_id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",
13   "has_more": false
14 }
```

Delete checkpoint permission

```
DELETE https://api.openai.com/v1/fine_tuning/checkpoints/{fine_tune  
d_model_checkpoint}/permissions/{permission_id}
```

NOTE: This endpoint requires an [admin API key](#).

Organization owners can use this endpoint to delete a permission for a fine-tuned model checkpoint.

Path parameters

fine_tuned_model_checkpoint string Required

The ID of the fine-tuned model checkpoint to delete a permission for.

permission_id string Required

The ID of the fine-tuned model checkpoint permission to delete.

Returns

The deletion status of the fine-tuned model checkpoint [permission object](#).

Example request

curl ⚡ [Copy](#)

```
1 curl https://api.openai.com/v1/fine_tuning/c  
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {  
2   "object": "checkpoint.permission",  
3   "id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",  
4   "deleted": true  
5 }
```

Retrieve fine-tuning job

```
GET https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}
```

Get info about a fine-tuning job.

[Learn more about fine-tuning](#)

Path parameters

fine_tuning_job_id string Required

The ID of the fine-tuning job.

Returns

The [fine-tuning](#) object with the given ID.

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/fine_tuning/j
```

Response

```
1  {
2      "object": "fine_tuning.job",
3      "id": "ftjob-abc123",
4      "model": "davinci-002",
5      "created_at": 1692661014,
6      "finished_at": 1692661190,
7      "fine_tuned_model": "ft:davinci-002:my-or",
8      "organization_id": "org-123",
9      "result_files": [
10          "file-abc123"
11      ],
12      "status": "succeeded",
13      "validation_file": null,
14      "training_file": "file-abc123",
15      "hyperparameters": {
16          "n_epochs": 4,
17          "batch_size": 1,
18          "learning_rate_multiplier": 1.0
19      },
20      "trained_tokens": 5768,
21      "integrations": [],
22      "seed": 0,
23      "estimated_finish": 0,
24      "method": {
25          "type": "supervised",
26          "supervised": {
27              "hyperparameters": {
28                  "n_epochs": 4,
29                  "batch_size": 1,
```

Cancel fine-tuning

```
POST https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}/cancel
```

Immediately cancel a fine-tune job.

Path parameters

fine_tuning_job_id string Required

The ID of the fine-tuning job to cancel.

Returns

The cancelled [fine-tuning](#) object.

```
30         "learning_rate_multiplier": 1.0
31     }
32 }
```

Example request

curl ⚡ [Copy](#)

```
1 curl -X POST https://api.openai.com/v1/fine_
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "object": "fine_tuning.job",
3   "id": "ftjob-abc123",
4   "model": "gpt-4o-mini-2024-07-18",
5   "created_at": 1721764800,
6   "fine_tuned_model": null,
7   "organization_id": "org-123",
8   "result_files": [],
9   "status": "cancelled",
10  "validation_file": "file-abc123",
11  "training_file": "file-abc123"
12 }
```

Resume fine-tuning

```
POST https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}/resume
```

Resume a fine-tune job.

Example request

curl ⚡ [Copy](#)

```
1 curl -X POST https://api.openai.com/v1/fine_
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Path parameters

fine_tuning_job_id string Required

The ID of the fine-tuning job to resume.

Returns

The resumed fine-tuning object.

Response

```

1  {
2    "object": "fine_tuning.job",
3    "id": "ftjob-abc123",
4    "model": "gpt-4o-mini-2024-07-18",
5    "created_at": 1721764800,
6    "fine_tuned_model": null,
7    "organization_id": "org-123",
8    "result_files": [],
9    "status": "queued",
10   "validation_file": "file-abc123",
11   "training_file": "file-abc123"
12 }
```

Pause fine-tuning

```
POST https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}/pause
```

Pause a fine-tune job.

Path parameters

fine_tuning_job_id string Required

The ID of the fine-tuning job to pause.

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/fine_tuning/jobs/{fine_tuning_job_id}/pause
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```

1  {
2    "object": "fine_tuning.job",
3    "id": "ftjob-abc123",
4    "model": "gpt-4o-mini-2024-07-18",
5    "created_at": 1721764800,
```

Returns

The paused [fine-tuning](#) object.

```

6   "fine_tuned_model": null,
7   "organization_id": "org-123",
8   "result_files": [],
9   "status": "paused",
10  "validation_file": "file-abc123",
11  "training_file": "file-abc123"
12 }
```

Training format for chat models using the supervised method

The per-line training example of a fine-tuning input file for chat models using the supervised method. Input messages may contain text or image content only. Audio and file input messages are not currently supported for fine-tuning.

functions Deprecated array

A list of functions the model may generate JSON inputs for.

✓ Show properties

messages array

✓ Show possible types

parallel_tool_calls boolean

Whether to enable [parallel function calling](#) during tool use.

tools array

A list of tools the model may generate JSON inputs for.

OBJECT Training format for chat models using...

```

1  {
2   "messages": [
3     { "role": "user", "content": "What is the capital of France?" },
4     {
5       "role": "assistant",
6       "tool_calls": [
7         {
8           "id": "call_id",
9           "type": "function",
10          "function": {
11            "name": "get_current_weather",
12            "arguments": "{\"location\": \"Paris, France\"}"
13          }
14        }
15      ]
16    }
17  ],
18  "parallel_tool_calls": false,
19  "tools": [
```

✓ Show properties

Training format for chat models using the preference method

The per-line training example of a fine-tuning input file for chat models using the dpo method. Input messages may contain text or image content only. Audio and file input messages are not currently supported for fine-tuning.

input object

✓ Show properties

non_preferred_output array

The non-preferred completion message for the output.

✓ Show possible types

preferred_output array

The preferred completion message for the output.

```

20  {
21    "type": "function",
22    "function": {
23      "name": "get_current_weather",
24      "description": "Get the current weather in a city",
25      "parameters": {
26        "type": "object",
27        "properties": {
28          "location": {
29            "type": "string",
30            "description": "The city and state/province where the weather should be reported"
31          },
32          "format": { "type": "string", "enum": ["location", "format"] }
33        }
34      }
35    }
36  }

```

OBJECT Training format for chat models using...

```

1  {
2    "input": {
3      "messages": [
4        { "role": "user", "content": "What is the weather like in San Francisco?" }
5      ],
6      "preferred_output": [
7        {
8          "role": "assistant",
9          "content": "The weather in San Francisco is currently sunny with a temperature of 75°F."
10        }
11      ]
12    },
13    "non_preferred_output": [
14      {
15        "role": "assistant",
16        "content": "The weather in San Francisco is currently sunny with a temperature of 75°F."
17      }
18    ]
19  }

```

✓ Show possible types

```
18 ]
19 }
```

Training format for reasoning models using the reinforcement method

Per-line training example for reinforcement fine-tuning. Note that

`messages` and `tools` are the only reserved keywords. Any other arbitrary key-value data can be included on training datapoints and will be available to reference during grading under the `{{ item.XXX }}` template variable. Input messages may contain text or image content only. Audio and file input messages are not currently supported for fine-tuning.

messages array

✓ Show possible types

tools array

A list of tools the model may generate JSON inputs for.

✓ Show properties

OBJECT Training format for reasoning models ... 

```
1 {
2   "messages": [
3     {
4       "role": "user",
5       "content": "Your task is to take a ch
6     },
7   ],
8   # Any other JSON data can be inserted int
9   "reference_answer": {
10     "donor_bond_counts": 5,
11     "acceptor_bond_counts": 7
12   }
13 }
```

The fine-tuning job object

The `fine_tuning.job` object represents a fine-tuning job that has been created through the API.

created_at integer

The Unix timestamp (in seconds) for when the fine-tuning job was created.

error object or null

For fine-tuning jobs that have `failed`, this will contain more information on the cause of the failure.

▼ Show properties

estimated_finish integer or null

The Unix timestamp (in seconds) for when the fine-tuning job is estimated to finish.

The value will be null if the fine-tuning job is not running.

fine_tuned_model string or null

The name of the fine-tuned model that is being created. The value will be null if the fine-tuning job is still running.

finished_at integer or null

The Unix timestamp (in seconds) for when the fine-tuning job was finished. The value will be null if the fine-tuning job is still running.

hyperparameters object

The hyperparameters used for the fine-tuning job. This value will only be returned when running `supervised` jobs.

▼ Show properties

id string

OBJECT The fine-tuning job object

```

1  {
2    "object": "fine_tuning.job",
3    "id": "ftjob-abc123",
4    "model": "davinci-002",
5    "created_at": 1692661014,
6    "finished_at": 1692661190,
7    "fine_tuned_model": "ft:davinci-002:my-or",
8    "organization_id": "org-123",
9    "result_files": [
10      "file-abc123"
11    ],
12    "status": "succeeded",
13    "validation_file": null,
14    "training_file": "file-abc123",
15    "hyperparameters": {
16      "n_epochs": 4,
17      "batch_size": 1,
18      "learning_rate_multiplier": 1.0
19    },
20    "trained_tokens": 5768,
21    "integrations": [],
22    "seed": 0,
23    "estimated_finish": 0,
24    "method": {
25      "type": "supervised",
26      "supervised": {
27        "hyperparameters": {
28          "n_epochs": 4,
29          "batch_size": 1,
30          "learning_rate_multiplier": 1.0
31        }
32      }
33    }
}

```

The object identifier, which can be referenced in the API endpoints.

integrations array or null

A list of integrations to enable for this fine-tuning job.

▼ Show possible types

```
34     "metadata": {  
35         "key": "value"  
36     }  
37 }
```

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

method object

The method used for fine-tuning.

▼ Show properties

model string

The base model that is being fine-tuned.

object string

The object type, which is always "fine_tuning.job".

organization_id string

The organization that owns the fine-tuning job.

result_files array

The compiled results file ID(s) for the fine-tuning job. You can retrieve the results with the [Files API](#).

seed integer

The seed used for the fine-tuning job.

status string

The current status of the fine-tuning job, which can be either `validating_files`, `queued`, `running`, `succeeded`, `failed`, or `cancelled`.

trained_tokens integer or null

The total number of billable tokens processed by this fine-tuning job. The value will be null if the fine-tuning job is still running.

training_file string

The file ID used for training. You can retrieve the training data with the [Files API](#).

validation_file string or null

The file ID used for validation. You can retrieve the validation results with the [Files API](#).

The fine-tuning job event object

Fine-tuning job event object

created_at integer

OBJECT The fine-tuning job event object

```
1 {
2   "object": "fine_tuning.job.event",
```

The Unix timestamp (in seconds) for when the fine-tuning job was created.

data object

The data associated with the event.

id string

The object identifier.

level string

The log level of the event.

message string

The message of the event.

object string

The object type, which is always "fine_tuning.job.event".

type string

The type of event.

```

3   "id": "ftevent-abc123"
4   "created_at": 1677610602,
5   "level": "info",
6   "message": "Created fine-tuning job",
7   "data": {},
8   "type": "message"
9 }
```

The fine-tuning job checkpoint object

The `fine_tuning.job.checkpoint` object represents a model checkpoint for a fine-tuning job that is ready to use.

created_at integer

OBJECT The fine-tuning job checkpoint object

```

1  {
2    "object": "fine_tuning.job.checkpoint",
```

The Unix timestamp (in seconds) for when the checkpoint was created.

fine_tuned_model_checkpoint string

The name of the fine-tuned checkpoint model that is created.

fine_tuning_job_id string

The name of the fine-tuning job that this checkpoint was created from.

id string

The checkpoint identifier, which can be referenced in the API endpoints.

metrics object

Metrics at the step number during the fine-tuning job.

▼ Show properties

object string

The object type, which is always "fine_tuning.job.checkpoint".

step_number integer

The step number that the checkpoint was created at.

```

3   "id": "ftckpt_qtZ5Gyk4BLq1SfLFWp3Rt03P",
4   "created_at": 1712211699,
5   "fine_tuned_model_checkpoint": "ft:gpt-4o",
6   "fine_tuning_job_id": "ftjob-fpbNQ3H1GrMe",
7   "metrics": {
8     "step": 88,
9     "train_loss": 0.478,
10    "train_mean_token_accuracy": 0.924,
11    "valid_loss": 10.112,
12    "valid_mean_token_accuracy": 0.145,
13    "full_valid_loss": 0.567,
14    "full_valid_mean_token_accuracy": 0.944
15  },
16  "step_number": 88
17 }
```

The fine-tuned model checkpoint permission object

The `checkpoint.permission` object represents a permission for a fine-tuned model checkpoint.

OBJECT The fine-tuned model checkpoint perm...



created_at integer

The Unix timestamp (in seconds) for when the permission was created.

id string

The permission identifier, which can be referenced in the API endpoints.

object string

The object type, which is always "checkpoint.permission".

project_id string

The project identifier that the permission is for.

```
1 {  
2   "object": "checkpoint.permission",  
3   "id": "cp_zc4Q7MP6XxulcVzj4MZdwsAB",  
4   "created_at": 1712211699,  
5   "project_id": "proj_abGMw1llN8IrBb6SvvY5A1"
```

Graders

Manage and run graders in the OpenAI platform. Related guide: [Graders](#)

String Check Grader

A StringCheckGrader object that performs a string comparison between input and reference using a specified operation.

input string

OBJECT String Check Grader

```
1 {  
2   "type": "string_check",  
3   "name": "Example string check grader",
```

The input text. This may include template strings.

name string

The name of the grader.

operation string

The string check operation to perform. One of `eq`, `ne`, `like`, or `ilike`.

```

4   "input": "{{sample.output_text}}",
5   "reference": "{{item.label}}",
6   "operation": "eq"
7 }
```

reference string

The reference text. This may include template strings.

type string

The object type, which is always `string_check`.

Text Similarity Grader

A `TextSimilarityGrader` object which grades text based on similarity metrics.

evaluation_metric string

The evaluation metric to use. One of `fuzzy_match`, `bleu`, `gleu`, `meteor`, `rouge_1`, `rouge_2`, `rouge_3`, `rouge_4`, `rouge_5`, or `rouge_l`.

input string

The text being graded.

OBJECT Text Similarity Grader

```

1 {
2   "type": "text_similarity",
3   "name": "Example text similarity grader",
4   "input": "{{sample.output_text}}",
5   "reference": "{{item.label}}",
6   "evaluation_metric": "fuzzy_match"
7 }
```

name string

The name of the grader.

reference string

The text being graded against.

type string

The type of grader.

Score Model Grader

A ScoreModelGrader object that uses a model to assign a score to the input.

input array

The input text. This may include template strings.

✓ Show properties

model string

The model to use for the evaluation.

name string

The name of the grader.

range array

OBJECT Score Model Grader

```
1  {
2    "type": "score_model",
3    "name": "Example score model grader",
4    "input": [
5      {
6        "role": "user",
7        "content": (
8          "Score how close the refere
9          " Return just a floating po
10         " Reference answer: {{item.
11         " Model answer: {{sample.ou
12       ),
13     }
14   ],
15   "model": "gpt-4o-2024-08-06",
```

The range of the score. Defaults to `[0, 1]`.

sampling_params object

The sampling parameters for the model.

```
16 "sampling_params": {
17   "temperature": 1,
18   "top_p": 1,
19   "seed": 42,
20 },
21 }
```

type string

The object type, which is always `score_model`.

Label Model Grader

A LabelModelGrader object which uses a model to assign labels to each item in the evaluation.

input array

▽ Show properties

labels array

The labels to assign to each item in the evaluation.

model string

The model to use for the evaluation. Must support structured outputs.

name string

The name of the grader.

OBJECT Label Model Grader

```
1 {
2   "name": "First label grader",
3   "type": "label_model",
4   "model": "gpt-4o-2024-08-06",
5   "input": [
6     {
7       "type": "message",
8       "role": "system",
9       "content": {
10         "type": "input_text",
11         "text": "Classify the sentiment of "
12       }
13     },
14     {
15       "type": "message",
16       "role": "user",
17       "content": {
```

passing_labels array

The labels that indicate a passing result. Must be a subset of labels.

type string

The object type, which is always `label_model`.

```

18     "type": "input_text",
19     "text": "Statement: {{item.response
20   }}
21   }
22 ],
23 "passing_labels": [
24   "positive"
25 ],
26 "labels": [
27   "positive",
28   "neutral",
29   "negative"
30 ]
31 }
```

Python Grader

A PythonGrader object that runs a python script on the input.

image_tag string

The image tag to use for the python script.

name string

The name of the grader.

source string

The source code of the python script.

type string

The object type, which is always `python`.

OBJECT Python Grader

```

1 {
2   "type": "python",
3   "name": "Example python grader",
4   "image_tag": "2025-05-08",
5   "source": """
6   def grade(sample: dict, item: dict) -> float:
7     """
8       Returns 1.0 if `output_text` equals `label`
9     """
10    output = sample.get("output_text")
11    label = item.get("label")
12    return 1.0 if output == label else 0.0
13
14 }
```

Multi Grader

A MultiGrader object combines the output of multiple graders to produce a single score.

calculate_output string

A formula to calculate the output based on grader results.

graders object

▽ Show possible types

name string

The name of the grader.

type string

The object type, which is always `multi`.

OBJECT Multi Grader



```
1  {
2    "type": "multi",
3    "name": "example multi grader",
4    "graders": [
5      {
6        "type": "text_similarity",
7        "name": "example text similarity grader",
8        "input": "The graded text",
9        "reference": "The reference text",
10       "evaluation_metric": "fuzzy_match"
11     },
12     {
13       "type": "string_check",
14       "name": "Example string check grader",
15       "input": "{{sample.output_text}}",
16       "reference": "{{item.label}}",
17       "operation": "eq"
18     }
19   ],
20   "calculate_output": "0.5 * text_similarit
21 }
```

Run grader Beta

POST https://api.openai.com/v1/fine_tuning/alpha/graders/run

Run a grader.

Request body

grader object Required

The grader used for the fine-tuning job.

▽ Show possible types

model_sample string Required

The model sample to be evaluated. This value will be used to populate the `sample` namespace. See [the guide](#) for more details. The `output_json` variable will be populated if the model sample is a valid JSON string.

item object Optional

The dataset item provided to the grader. This will be used to populate the `item` namespace. See [the guide](#) for more details.

Returns

The results from the grader run.

Example request

curl ⚡ [Copy](#)

```
1 curl -X POST https://api.openai.com/v1/fine
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -d '{
5     "grader": {
6         "type": "score_model",
7         "name": "Example score model grader",
8         "input": [
9             {
10                 "role": "user",
11                 "content": "Score how close the r
12             }
13         ],
14         "model": "gpt-4o-2024-08-06",
15         "sampling_params": {
16             "temperature": 1,
17             "top_p": 1,
18             "seed": 42
19         }
20     },
21     "item": {
22         "reference_answer": "fuzzy wuzzy was
23     },
24     "model_sample": "fuzzy wuzzy was a bear
25 }'
```

Response

[Copy](#)

Validate grader Beta

```
1  {
2      "reward": 1.0,
3      "metadata": {
4          "name": "Example score model grader",
5          "type": "score_model",
6          "errors": {
7              "formula_parse_error": false,
8              "sample_parse_error": false,
9              "truncated_observation_error": false,
10             "unresponsive_reward_error": false,
11             "invalid_variable_error": false,
12             "other_error": false,
13             "python_grader_server_error": false,
14             "python_grader_server_error_type": nu,
15             "python_grader_runtime_error": false,
16             "python_grader_runtime_error_details": ,
17             "model_grader_server_error": false,
18             "model_grader_refusal_error": false,
19             "model_grader_parse_error": false,
20             "model_grader_server_error_details": ,
21         },
22         "execution_time": 4.365238428115845,
23         "scores": {},
24         "token_usage": {
25             "prompt_tokens": 190,
26             "total_tokens": 324,
27             "completion_tokens": 134,
28             "cached_tokens": 0
29         },
30         "sampled_model_name": "gpt-4o-2024-08-0
31     },
32     "sub_rewards": {},
33     "model_grader_token_usage_per_model": {
34         "gpt-4o-2024-08-06": {
35             "prompt_tokens": 190,
```

POST https://api.openai.com/v1/fine_tuning/alpha/graders/validate

Validate a grader.

Request body

grader object Required

The grader used for the fine-tuning job.

▽ Show possible types

Returns

The validated grader object.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/fine_tuning/
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -d '{
5     "grader": {
6       "type": "string_check",
7       "name": "Example string check grader"
8       "input": "{{sample.output_text}}",
9       "reference": "{{item.label}}",
10      "operation": "eq"
11    }
12  }'
```

Response

⌚

```
1 {
2   "grader": {
3     "type": "string_check",
4     "name": "Example string check grader",
5     "input": "{{sample.output_text}}",
6     "reference": "{{item.label}}",
7     "operation": "eq"
8   }
9 }
```

Batch

Create large batches of API requests for asynchronous processing. The Batch API returns completions within 24 hours for a 50% discount. Related guide: [Batch](#)

Create batch

POST <https://api.openai.com/v1/batches>

Creates and executes a batch from an uploaded file of requests

Request body

completion_window string Required

The time frame within which the batch should be processed. Currently only `24h` is supported.

endpoint string Required

The endpoint to be used for all requests in the batch. Currently `/v1/responses`, `/v1/chat/completions`, `/v1/embeddings`, and `/v1/completions` are supported. Note that `/v1/embeddings` batches are also restricted to a maximum of 50,000 embedding inputs across all requests in the batch.

input_file_id string Required

The ID of an uploaded file that contains requests for the new batch.

See [upload file](#) for how to upload a file.

Your input file must be formatted as a [JSONL file](#), and must be uploaded with the purpose `batch`. The file can contain up to 50,000 requests, and can be up to 200

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/batches \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -d '{
5     "input_file_id": "file-abc123",
6     "endpoint": "/v1/chat/completions",
7     "completion_window": "24h"
8   }'
```

Response

⌚

```
1 {
2   "id": "batch_abc123",
3   "object": "batch",
4   "endpoint": "/v1/chat/completions",
5   "errors": null,
6   "input_file_id": "file-abc123",
7   "completion_window": "24h",
8   "status": "validating",
9   "output_file_id": null,
10  "error_file_id": null,
11  "created_at": 1711471533,
12  "in_progress_at": null,
```

MB in size.

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

Returns

The created [Batch](#) object.

```

13 "expires_at": null,
14 "finalizing_at": null,
15 "completed_at": null,
16 "failed_at": null,
17 "expired_at": null,
18 "canceling_at": null,
19 "cancelled_at": null,
20 "request_counts": {
21   "total": 0,
22   "completed": 0,
23   "failed": 0
24 },
25 "metadata": {
26   "customer_id": "user_123456789",
27   "batch_description": "Nightly eval job"
28 }
29 }
```

Retrieve batch

GET https://api.openai.com/v1/batches/{batch_id}

Retrieves a batch.

Path parameters

batch_id string Required

The ID of the batch to retrieve.

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/batches/batch_id
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
```

Response

```

1 {
2   "id": "batch_abc123",
```

Returns

The [Batch](#) object matching the specified ID.

Cancel batch

```
POST https://api.openai.com/v1/batches/{batch_id}/cancel
```

Cancels an in-progress batch. The batch will be in status `canceling` for up to 10 minutes, before changing to `cancelled`, where it will have partial results (if any) available in the output file.

Path parameters

```

3   "object": "batch",
4   "endpoint": "/v1/completions",
5   "errors": null,
6   "input_file_id": "file-abc123",
7   "completion_window": "24h",
8   "status": "completed",
9   "output_file_id": "file-cvaTdG",
10  "error_file_id": "file-H0WS94",
11  "created_at": 1711471533,
12  "in_progress_at": 1711471538,
13  "expires_at": 1711557933,
14  "finalizing_at": 1711493133,
15  "completed_at": 1711493163,
16  "failed_at": null,
17  "expired_at": null,
18  "cancelling_at": null,
19  "cancelled_at": null,
20  "request_counts": {
21    "total": 100,
22    "completed": 95,
23    "failed": 5
24  },
25  "metadata": {
26    "customer_id": "user_123456789",
27    "batch_description": "Nightly eval job"
28  }
29

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/batches/batch_id
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -X POST

```

batch_id string Required

The ID of the batch to cancel.

Returns

The [Batch](#) object matching the specified ID.

Response

```

1  {
2    "id": "batch_abc123",
3    "object": "batch",
4    "endpoint": "/v1/chat/completions",
5    "errors": null,
6    "input_file_id": "file-abc123",
7    "completion_window": "24h",
8    "status": "cancelling",
9    "output_file_id": null,
10   "error_file_id": null,
11   "created_at": 1711471533,
12   "in_progress_at": 1711471538,
13   "expires_at": 1711557933,
14   "finalizing_at": null,
15   "completed_at": null,
16   "failed_at": null,
17   "expired_at": null,
18   "cancelling_at": 1711475133,
19   "cancelled_at": null,
20   "request_counts": {
21     "total": 100,
22     "completed": 23,
23     "failed": 1
24   },
25   "metadata": {
26     "customer_id": "user_123456789",
27     "batch_detailed_status": "Cancelling"
28   }

```

Example request
curl copy

```

1 curl https://api.openai.com/v1/batches?limit=10
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json"

```

List batch

GET <https://api.openai.com/v1/batches>

List your organization's batches.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

Returns

A list of paginated [Batch](#) objects.

The batch object

cancelled_at integer

The Unix timestamp (in seconds) for when the batch was cancelled.

Response

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "id": "batch_abc123",
6        "object": "batch",
7        "endpoint": "/v1/chat/completions",
8        "errors": null,
9        "input_file_id": "file-abc123",
10       "completion_window": "24h",
11       "status": "completed",
12       "output_file_id": "file-cvaTdG",
13       "error_file_id": "file-H0WS94",
14       "created_at": 1711471533,
15       "in_progress_at": 1711471538,
16       "expires_at": 1711557933,
17       "finalizing_at": 1711493133,
18       "completed_at": 1711493163,
19       "failed_at": null,
20       "expired_at": null,
21       "cancelling_at": null,
22       "cancelled_at": null,
23       "request_counts": {
24         "total": 100,
25         "completed": 95,
26         "failed": 5
27       },
28       "metadata": {
29         "customer_id": "user_123456789",

```

OBJECT The batch object

```

1  {
2    "id": "batch_abc123",

```

cancelling_at integer

The Unix timestamp (in seconds) for when the batch started cancelling.

completed_at integer

The Unix timestamp (in seconds) for when the batch was completed.

completion_window string

The time frame within which the batch should be processed.

created_at integer

The Unix timestamp (in seconds) for when the batch was created.

endpoint string

The OpenAI API endpoint used by the batch.

error_file_id string

The ID of the file containing the outputs of requests with errors.

errors object

>Show properties

expired_at integer

The Unix timestamp (in seconds) for when the batch expired.

expires_at integer

The Unix timestamp (in seconds) for when the batch will expire.

failed_at integer

```

3   "object": "batch",
4   "endpoint": "/v1/completions",
5   "errors": null,
6   "input_file_id": "file-abc123",
7   "completion_window": "24h",
8   "status": "completed",
9   "output_file_id": "file-cvaTdG",
10  "error_file_id": "file-H0WS94",
11  "created_at": 1711471533,
12  "in_progress_at": 1711471538,
13  "expires_at": 1711557933,
14  "finalizing_at": 1711493133,
15  "completed_at": 1711493163,
16  "failed_at": null,
17  "expired_at": null,
18  "cancelling_at": null,
19  "cancelled_at": null,
20  "request_counts": {
21    "total": 100,
22    "completed": 95,
23    "failed": 5
24  },
25  "metadata": {
26    "customer_id": "user_123456789",
27    "batch_description": "Nightly eval job"
28  }
29 }
```

The Unix timestamp (in seconds) for when the batch failed.

finalizing_at integer

The Unix timestamp (in seconds) for when the batch started finalizing.

id string

in_progress_at integer

The Unix timestamp (in seconds) for when the batch started processing.

input_file_id string

The ID of the input file for the batch.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

object string

The object type, which is always `batch`.

output_file_id string

The ID of the file containing the outputs of successfully executed requests.

request_counts object

The request counts for different statuses within the batch.

✓ Show properties

status string

The current status of the batch.

The request input object

The per-line object of the batch input file

custom_id string

A developer-provided per-request id that will be used to match outputs to inputs.

Must be unique for each request in a batch.

method string

The HTTP method to be used for the request. Currently only `POST` is supported.

url string

The OpenAI API relative URL to be used for the request. Currently

`/v1/chat/completions` , `/v1/embeddings` , and `/v1/completions` are supported.

OBJECT The request input object



```
{"custom_id": "request-1", "method": "POST", "url": "https://api.openai.com/v1/completions"}
```

The request output object

The per-line object of the batch output and error files

custom_id string

A developer-provided per-request id that will be used to match outputs to inputs.

error object or null

For requests that failed with a non-HTTP error, this will contain more information on the cause of the failure.

▼ Show properties

id string

response object or null

▼ Show properties

OBJECT The request output object

```
{"id": "batch_req_wnaDys", "custom_id": "reqe...
```

Files

Files are used to upload documents that can be used with features like [Assistants](#), [Fine-tuning](#), and [Batch API](#).

Upload file

POST https://api.openai.com/v1/files

Upload a file that can be used across various endpoints. Individual files can be up to 512 MB, and the size of all files uploaded by one organization can be up to 100 GB.

The Assistants API supports files up to 2 million tokens and of specific file types. See the [Assistants Tools guide](#) for details.

The Fine-tuning API only supports `.jsonl` files. The input also has certain required formats for fine-tuning [chat](#) or [completions](#) models.

The Batch API only supports `.jsonl` files up to 200 MB in size. The input also has a specific required [format](#).

Please [contact us](#) if you need to increase these storage limits.

Request body

file file Required

The File object (not file name) to be uploaded.

purpose string Required

The intended purpose of the uploaded file. One of: - `assistants` : Used in the Assistants API - `batch` : Used in the Batch API - `fine-tune` : Used for fine-tuning - `vision` : Images used for vision fine-tuning - `user_data` : Flexible file type for any purpose - `evals` : Used for eval data sets

Returns

The uploaded `File` object.

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/files \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -F purpose="fine-tune" \
4   -F file="@mydata.jsonl"
```

Response

```
1 {
2   "id": "file-abc123",
3   "object": "file",
4   "bytes": 120000,
5   "created_at": 1677610602,
6   "filename": "mydata.jsonl",
7   "purpose": "fine-tune",
8 }
```

List files

GET <https://api.openai.com/v1/files>

Returns a list of files.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 10000

A limit on the number of objects to be returned. Limit can range between 1 and 10,000, and the default is 10,000.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

purpose string Optional

Only return files with the given purpose.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/files \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "file-abc123",
6       "object": "file",
7       "bytes": 175,
8       "created_at": 1613677385,
9       "filename": "salesOverview.pdf",
10      "purpose": "assistants",
11    },
12    {
13      "id": "file-abc456",
14      "object": "file",
15      "bytes": 140,
16      "created_at": 1613779121,
17      "filename": "puppy.jsonl",
18      "purpose": "fine-tune",
19    }
20  ],
```

Returns

A list of [File](#) objects.

```

21 "first_id": "file-abc123",
22 "last_id": "file-abc456",
23 "has_more": false
24 }
```

Retrieve file

GET https://api.openai.com/v1/files/{file_id}

Returns information about a specific file.

Path parameters

file_id string Required

The ID of the file to use for this request.

Returns

The [File](#) object matching the specified ID.

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/files/file-ab
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```

1 {
2   "id": "file-abc123",
3   "object": "file",
4   "bytes": 120000,
5   "created_at": 1677610602,
6   "filename": "mydata.jsonl",
7   "purpose": "fine-tune",
8 }
```

Delete file

```
DELETE https://api.openai.com/v1/files/{file_id}
```

Delete a file.

Path parameters

file_id string **Required**

The ID of the file to use for this request.

Returns

Deletion status.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/files/file-ab
2   -X DELETE \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

⌚

```
1 {
2   "id": "file-abc123",
3   "object": "file",
4   "deleted": true
5 }
```

Retrieve file content

```
GET https://api.openai.com/v1/files/{file_id}/content
```

Returns the contents of the specified file.

Path parameters

file_id string **Required**

The ID of the file to use for this request.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/files/file-ab
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Returns

The file content.

The file object

The `File` object represents a document that has been uploaded to OpenAI.

bytes integer

The size of the file, in bytes.

created_at integer

The Unix timestamp (in seconds) for when the file was created.

expires_at integer

The Unix timestamp (in seconds) for when the file will expire.

filename string

The name of the file.

id string

The file identifier, which can be referenced in the API endpoints.

OBJECT The file object

```
1 {
2   "id": "file-abc123",
3   "object": "file",
4   "bytes": 120000,
5   "created_at": 1677610602,
6   "expires_at": 1680202602,
7   "filename": "salesOverview.pdf",
8   "purpose": "assistants",
9 }
```

object string

The object type, which is always `file`.

purpose string

The intended purpose of the file. Supported values are `assistants`,
`assistants_output`, `batch`, `batch_output`, `fine-tune`,
`fine-tune-results`, `vision`, and `user_data`.

status Deprecated string

Deprecated. The current status of the file, which can be either `uploaded`,
`processed`, or `error`.

status_details Deprecated string

Deprecated. For details on why a fine-tuning training file failed validation, see the
`error` field on `fine_tuning.job`.

Uploads

Allows you to upload large files in multiple parts.

Create upload

POST https://api.openai.com/v1/uploads

Creates an intermediate [Upload](#) object that you can add [Parts](#) to. Currently, an Upload can accept at most 8 GB in total and expires after an hour after you create it.

Once you complete the Upload, we will create a [File](#) object that contains all the parts you uploaded. This File is usable in the rest of our platform as a regular File object.

For certain `purpose` values, the correct `mime_type` must be specified.

Please refer to documentation for the [supported MIME types for your use case](#).

For guidance on the proper filename extensions for each purpose, please follow the documentation on [creating a File](#).

Request body

bytes integer Required

The number of bytes in the file you are uploading.

filename string Required

The name of the file to upload.

mime_type string Required

The MIME type of the file.

This must fall within the supported MIME types for your file purpose. See the [supported MIME types for assistants and vision](#).

purpose string Required

The intended purpose of the uploaded file.

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/uploads \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -d '{
4     "purpose": "fine-tune",
5     "filename": "training_examples.jsonl",
6     "bytes": 2147483648,
7     "mime_type": "text/json"
8   }'
```

Response

```
1 {
2   "id": "upload_abc123",
3   "object": "upload",
4   "bytes": 2147483648,
5   "created_at": 1719184911,
6   "filename": "training_examples.jsonl",
7   "purpose": "fine-tune",
8   "status": "pending",
9   "expires_at": 1719127296
10 }
```

See the [documentation on File purposes](#).

Returns

The [Upload](#) object with status `pending`.

Add upload part

POST https://api.openai.com/v1/uploads/{upload_id}/parts

Adds a [Part](#) to an [Upload](#) object. A Part represents a chunk of bytes from the file you are trying to upload.

Each Part can be at most 64 MB, and you can add Parts until you hit the Upload maximum of 8 GB.

It is possible to add multiple Parts in parallel. You can decide the intended order of the Parts when you [complete the Upload](#).

Path parameters

upload_id string Required

The ID of the Upload.

Request body

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/uploads/upload_id
2   -F data="aHR0cHM6Ly9hcGkub3BlbmFpLmNvbS92M
```

Response

```
1 {
2   "id": "part_def456",
3   "object": "upload.part",
4   "created_at": 1719185911,
5   "upload_id": "upload_abc123"
6 }
```

data file Required

The chunk of bytes for this Part.

Returns

The upload [Part](#) object.

Complete upload

```
POST https://api.openai.com/v1/uploads/{upload_id}/complete
```

Completes the [Upload](#).

Within the returned Upload object, there is a nested [File](#) object that is ready to use in the rest of the platform.

You can specify the order of the Parts by passing in an ordered list of the Part IDs.

The number of bytes uploaded upon completion must match the number of bytes initially specified when creating the Upload object. No Parts may be added after an Upload is completed.

Path parameters

upload_id string Required

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/uploads/upload_id/complete
2 -d '{'
3   "part_ids": ["part_def456", "part_ghi789"]
4 }'
```

Response

⌂

```
1 {
2   "id": "upload_abc123",
3   "object": "upload",
4   "bytes": 2147483648,
5   "created_at": 1719184911,
6   "filename": "training_examples.jsonl",
7   "purpose": "fine-tune",
8   "status": "completed",
```

The ID of the Upload.

Request body

part_ids array Required

The ordered list of Part IDs.

md5 string Optional

The optional md5 checksum for the file contents to verify if the bytes uploaded matches what you expect.

```

9   "expires_at": 1719127296,
10  "file": {
11    "id": "file-xyz321",
12    "object": "file",
13    "bytes": 2147483648,
14    "created_at": 1719186911,
15    "filename": "training_examples.jsonl",
16    "purpose": "fine-tune",
17  }
18 }
```

Returns

The [Upload](#) object with status `completed` with an additional `file` property containing the created usable File object.

Cancel upload

POST https://api.openai.com/v1/uploads/{upload_id}/cancel

Cancels the Upload. No Parts may be added after an Upload is cancelled.

Path parameters

curl ⌂

Example request

curl https://api.openai.com/v1/uploads/upload_id/cancel



Response

upload_id string Required

The ID of the Upload.

Returns

The [Upload](#) object with status `cancelled`.

```

1  {
2    "id": "upload_abc123",
3    "object": "upload",
4    "bytes": 2147483648,
5    "created_at": 1719184911,
6    "filename": "training_examples.jsonl",
7    "purpose": "fine-tune",
8    "status": "cancelled",
9    "expires_at": 1719127296
10 }
```

The upload object

The Upload object can accept byte chunks in the form of Parts.

bytes integer

The intended number of bytes to be uploaded.

created_at integer

The Unix timestamp (in seconds) for when the Upload was created.

expires_at integer

The Unix timestamp (in seconds) for when the Upload will expire.

file undefined or null

The ready File object after the Upload is completed.

filename string

OBJECT The upload object

```

1  {
2    "id": "upload_abc123",
3    "object": "upload",
4    "bytes": 2147483648,
5    "created_at": 1719184911,
6    "filename": "training_examples.jsonl",
7    "purpose": "fine-tune",
8    "status": "completed",
9    "expires_at": 1719127296,
10   "file": {
11     "id": "file-xyz321",
12     "object": "file",
13     "bytes": 2147483648,
14     "created_at": 1719186911,
15     "filename": "training_examples.jsonl",
16     "purpose": "fine-tune",
```

```

17 }
18 }
```

The name of the file to be uploaded.

id string

The Upload unique identifier, which can be referenced in API endpoints.

object string

The object type, which is always "upload".

purpose string

The intended purpose of the file. [Please refer here](#) for acceptable values.

status string

The status of the Upload.

The upload part object

The upload Part represents a chunk of bytes we can add to an Upload object.

created_at integer

The Unix timestamp (in seconds) for when the Part was created.

id string

The upload Part unique identifier, which can be referenced in API endpoints.

object string
OBJECT The upload part object

```

1 {
2   "id": "part_def456",
3   "object": "upload.part",
4   "created_at": 1719186911,
5   "upload_id": "upload_abc123"
6 }
```

The object type, which is always `upload.part`.

upload_id string

The ID of the Upload object that this Part was added to.

Models

List and describe the various models available in the API. You can refer to the [Models](#) documentation to understand what models are available and the differences between them.

List models

GET `https://api.openai.com/v1/models`

Lists the currently available models, and provides basic information about each one such as the owner and availability.

Returns

A list of [model](#) objects.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/models \
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1  {
2    "object": "list",
3    "data": [
4      {
5        "id": "model-id-0",
```

```

6     "object": "model",
7     "created": 1686935002,
8     "owned_by": "organization-owner"
9   },
10  {
11    "id": "model-id-1",
12    "object": "model",
13    "created": 1686935002,
14    "owned_by": "organization-owner",
15  },
16  {
17    "id": "model-id-2",
18    "object": "model",
19    "created": 1686935002,
20    "owned_by": "openai"
21  },
22  ],
23  "object": "list"
24 }

```

Retrieve model

GET <https://api.openai.com/v1/models/{model}>

Retrieves a model instance, providing basic information about the model such as the owner and permissioning.

Path parameters

model string **Required**

Example request

gpt-5 ⚡ curl ⚡ [Copy](#)

```

1 curl https://api.openai.com/v1/models/gpt-5
2 -H "Authorization: Bearer $OPENAI_API_KEY"

```

Response

```

1 {
2   "id": "gpt-5",
3   "object": "model",

```

The ID of the model to use for this request

```
4  "created": 1686935002,  
5  "owned_by": "openai"  
6 }
```

Returns

The [model](#) object matching the specified ID.

Delete a fine-tuned model

```
DELETE https://api.openai.com/v1/models/{model}
```

Delete a fine-tuned model. You must have the Owner role in your organization to delete a model.

Path parameters

model string Required

The model to delete

Returns

Deletion status.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/models/ft:gpt-4o-mini:acemeco:suffix:abc123
2   -X DELETE \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "id": "ft:gpt-4o-mini:acemeco:suffix:abc123",
3   "object": "model",
4   "deleted": true
5 }
```

The model object

Describes an OpenAI model offering that can be used with the API.

created integer

The Unix timestamp (in seconds) when the model was created.

id string

The model identifier, which can be referenced in the API endpoints.

OBJECT The model object

⌂

```
1 {
2   "id": "gpt-5",
3   "object": "model",
4   "created": 1686935002,
5   "owned_by": "openai"
6 }
```

object string

The object type, which is always "model".

owned_by string

The organization that owns the model.

Moderations

Given text and/or image inputs, classifies if those inputs are potentially harmful across several categories.

Related guide: [Moderations](#)

Create moderation

POST <https://api.openai.com/v1/moderations>

Classifies if text and/or image inputs are potentially harmful. Learn more in the [moderation guide](#).

Request body

input string or array Required

Single string

Image and text

Example request

curl ⚡ ↗

```
1 curl https://api.openai.com/v1/moderations \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY" \
4   -d '{
5     "input": "I want to kill them."
6   }'
```

Input (or inputs) to classify. Can be a single string, an array of strings, or an array of multi-modal input objects similar to other models.

✓ Show possible types

model string Optional Defaults to omni-moderation-latest

The content moderation model you would like to use. Learn more in [the moderation guide](#), and learn about available models [here](#).

Returns

A [moderation](#) object.

The moderation object

Represents if a given text input is potentially harmful.

id string

The unique identifier for the moderation request.

model string

The model used to generate the moderation results.

Response

```

1  {
2    "id": "modr-AB8Cj0Tu2jiq12hp1AQPfeqFWaORR",
3    "model": "text-moderation-007",
4    "results": [
5      {
6        "flagged": true,
7        "categories": {
8          "sexual": false,
9          "hate": false,
10         "harassment": true,
11         "self-harm": false,
12         "sexual/minors": false,
13         "hate/threatening": false,
14         "violence/graphic": false,
15         "self-harm/intent": false,
16         "self-harm/instructions": false,
17         "harassment/threatening": true,
18         "violence": true
19       },
20       "category_scores": {
21         "sexual": 0.000011726012417057063,
22         "hate": 0.22706663608551025,
23         "harassment": 0.5215635299682617,
24         "self-harm": 2.227119921371923e-6,

```

OBJECT The moderation object

```

1  {
2    "id": "modr-0d9740456c391e43c445bf0f01094",
3    "model": "omni-moderation-latest",
4    "results": [
5      {
6        "flagged": true,
7        "categories": {

```

results array

A list of moderation objects.

▼ Show properties

Vector stores

Vector stores power semantic search for the Retrieval API and the [file_search](#) Assistants APIs.

Related guide: [File Search](#)

```
8      "harassment": true,
9      "harassment/threatening": true,
10     "sexual": false,
11     "hate": false,
12     "hate/threatening": false,
13     "illicit": false,
14     "illicit/violent": false,
15     "self-harm/intent": false,
16     "self-harm/instructions": false,
17     "self-harm": false,
18     "sexual/minors": false,
19     "violence": true,
20     "violence/graphic": true
21   },
22   "category_scores": {
23     "harassment": 0.8189693396524255,
24     "harassment/threatening": 0.8049854
25     "sexual": 1.573112165348997e-6,
26     "hate": 0.007562942636942845,
27     "hate/threatening": 0.0042088545918
28     "illicit": 0.030535955153511665,
29     "illicit/violent": 0.00892530672238
30     "self-harm/intent": 0.0002302393097
31     "self-harm/instructions": 0.0002293
32     "self-harm": 0.012598046106750154,
33     "sexual/minors": 2.212566909570261e
34     "violence": 0.9999992735124786,
35     "violence/graphic": 0.8430648711570
36   },
37   "category_applied_input_types": {
38     "harassment": [
39       "text"
40     ],
41     "harassment/threatening": [
42       "text"
43     ]
44   }
45 }
```

Create vector store

POST https://api.openai.com/v1/vector_stores

Create a vector store.

Request body

chunking_strategy object Optional

The chunking strategy used to chunk the file(s). If not set, will use the `auto` strategy.

Only applicable if `file_ids` is non-empty.

✗ Show possible types

expires_after object Optional

The expiration policy for a vector store.

✗ Show properties

file_ids array Optional

A list of [File](#) IDs that the vector store should use. Useful for tools like `file_search` that can access files.

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

```
43     ],
44     "sexual": [
45       "text".

```

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -d '{
6     "name": "Support FAQ"
7   }'
```

56 1

Response

```
1 {
2   "id": "vs_abc123",
3   "object": "vector_store",
4   "created_at": 1699061776,
5   "name": "Support FAQ",
6   "bytes": 139920,
7   "file_counts": {
8     "in_progress": 0,
9     "completed": 3,
10    "failed": 0,
11    "cancelled": 0,
12    "total": 3
13  }
14 }
```

...

```
74   ],
75   "violence": [
76     "text",
77     "image"
```

name string Optional

The name of the vector store.

Returns

A [vector store](#) object.

```

78     ],
79     "violence/graphic": [
80       "text",
81       "image"
82     ]
83   }
84 }
85 1

```

List vector stores

GET https://api.openai.com/v1/vector_stores

Returns a list of vector stores.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with

Example request

[curl](#) [Copy](#)

```

1 curl https://api.openai.com/v1/vector_stores
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -H "OpenAI-Beta: assistants=v2"

```

Response

```

1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "vs_abc123",
6       "object": "vector_store",
7       "created_at": 1699061776,
8       "name": "Support FAQ",

```

`obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

A list of vector store objects.

Retrieve vector store

```
GET https://api.openai.com/v1/vector_stores/{vector_store_id}
```

Retrieves a vector store.

Path parameters

vector_store_id string **Required**

```

9     "bytes": 139920,
10    "file_counts": {
11      "in_progress": 0,
12      "completed": 3,
13      "failed": 0,
14      "cancelled": 0,
15      "total": 3
16    }
17  },
18  [
19    {
20      "id": "vs_abc456",
21      "object": "vector_store",
22      "created_at": 1699061776,
23      "name": "Support FAQ v2",
24      "bytes": 139920,
25      "file_counts": {
26        "in_progress": 0,
27        "completed": 3,
28        "failed": 0,
29        "cancelled": 0,
30        "total": 3
31      }
32    ],
33    "first_id": "vs_abc123",
34    "last_id": "vs_abc456",

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2"
```

Response

⌚

The ID of the vector store to retrieve.

Returns

The [vector store](#) object matching the specified ID.

```

1 {
2   "id": "vs_abc123",
3   "object": "vector_store",
4   "created_at": 1699061776
5 }
```

Modify vector store

POST https://api.openai.com/v1/vector_stores/{vector_store_id}

Modifies a vector store.

Path parameters

vector_store_id string Required

The ID of the vector store to modify.

Request body

expires_after object or null Optional

The expiration policy for a vector store.

▼ Show properties

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2"
5   -d '{
6     "name": "Support FAQ"
7   }'
```

Response

```

1 {
2   "id": "vs_abc123",
3   "object": "vector_store",
4   "created_at": 1699061776,
5   "name": "Support FAQ",
6   "bytes": 139920,
7   "file_counts": {
```

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string or null Optional

The name of the vector store.

```

8     "in_progress": 0,
9     "completed": 3,
10    "failed": 0,
11    "cancelled": 0,
12    "total": 3
13  }
14 }
```

Returns

The modified [vector store](#) object.

Delete vector store

```
DELETE https://api.openai.com/v1/vector_stores/{vector_store_id}
```

Delete a vector store.

Path parameters

vector_store_id string **Required**

Example request

[curl](#) [Copy](#)

```

1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -X DELETE
```

The ID of the vector store to delete.

Returns

Deletion status

Response

```
1 {
2   id: "vs_abc123",
3   object: "vector_store.deleted",
4   deleted: true
```

Search vector store

```
POST https://api.openai.com/v1/vector_stores/{vector_store_id}/search
```

Search a vector store for relevant chunks based on a query and file attributes filter.

Path parameters

vector_store_id string Required

The ID of the vector store to search.

Request body

query string or array Required

A query string for a search

Example request

curl ⌂

```
1 curl -X POST \
2 https://api.openai.com/v1/vector_stores/vs_a
3 -H "Authorization: Bearer $OPENAI_API_KEY" \
4 -H "Content-Type: application/json" \
5 -d '{"query": "What is the return policy?",
```

Response

```
1 {
2   "object": "vector_store.search_results.page",
3   "search_query": "What is the return policy?",
4   "data": [
5     {
6       "file_id": "file_123",
7       "filename": "document.pdf",
8       "score": 0.95,
9       "attributes": {
```

filters object Optional

A filter to apply based on file attributes.

✓ Show possible types

max_num_results integer Optional Defaults to 10

The maximum number of results to return. This number should be between 1 and 50 inclusive.

ranking_options object Optional

Ranking options for search.

✓ Show properties

rewrite_query boolean Optional Defaults to false

Whether to rewrite the natural language query for vector search.

Returns

A page of search results from the vector store.

```

10      "author": "John Doe",
11      "date": "2023-01-01"
12    },
13    "content": [
14      {
15        "type": "text",
16        "text": "Relevant chunk"
17      }
18    ]
19  },
20  {
21    "file_id": "file_456",
22    "filename": "notes.txt",
23    "score": 0.89,
24    "attributes": {
25      "author": "Jane Smith",
26      "date": "2023-01-02"
27    },
28    "content": [
29      {
30        "type": "text",
31        "text": "Sample text content from"
32      }
33    ]
34  }
35 ],
36 "has_more": false,
37 "next_page": null
38 }
```

The vector store object

A vector store is a collection of processed files can be used by the

[file_search](#) tool.

OBJECT The vector store object



created_at integer

The Unix timestamp (in seconds) for when the vector store was created.

expires_after object

The expiration policy for a vector store.

▽ Show properties

expires_at integer or null

The Unix timestamp (in seconds) for when the vector store will expire.

file_counts object

▽ Show properties

id string

The identifier, which can be referenced in API endpoints.

last_active_at integer or null

The Unix timestamp (in seconds) for when the vector store was last active.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string

The name of the vector store.

```
1  {
2    "id": "vs_123",
3    "object": "vector_store",
4    "created_at": 1698107661,
5    "usage_bytes": 123456,
6    "last_active_at": 1698107661,
7    "name": "my_vector_store",
8    "status": "completed",
9    "file_counts": {
10      "in_progress": 0,
11      "completed": 100,
12      "cancelled": 0,
13      "failed": 0,
14      "total": 100
15    },
16    "last_used_at": 1698107661
```

object string

The object type, which is always `vector_store`.

status string

The status of the vector store, which can be either `expired`, `in_progress`, or `completed`. A status of `completed` indicates that the vector store is ready for use.

usage_bytes integer

The total number of bytes used by the files in the vector store.

Vector store files

Vector store files represent files inside a vector store.

Related guide: [File Search](#)

Create vector store file

```
POST https://api.openai.com/v1/vector_stores/{vector_store_id}/file  
s
```

Create a vector store file by attaching a [File](#) to a [vector store](#).

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/vector_stores  
2   -H "Authorization: Bearer $OPENAI_API_KEY"  
3   -H "Content-Type: application/json" \
```

Path parameters

vector_store_id string Required

The ID of the vector store for which to create a File.

Request body

file_id string Required

A [File](#) ID that the vector store should use. Useful for tools like [file_search](#) that can access files.

attributes map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard. Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters, booleans, or numbers.

chunking_strategy object Optional

The chunking strategy used to chunk the file(s). If not set, will use the [auto](#) strategy.

▼ Show possible types

Returns

A [vector store file](#) object.

```
4     -H "OpenAI-Beta: assistants=v2" \
5     -d '{
6       "file_id": "file-abc123"
7     }'
```

Response

```
1 {
2   "id": "file-abc123",
3   "object": "vector_store_file",
4   "created_at": 1699061776,
5   "usage_bytes": 1234,
6   "vector_store_id": "vs_abcd",
7   "status": "completed",
8   "last_error": null
9 }
```

List vector store files

```
GET https://api.openai.com/v1/vector_stores/{vector_store_id}/files
```

Returns a list of vector store files.

Path parameters

vector_store_id string Required

The ID of the vector store that the files belong to.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

filter string Optional

Filter by file status. One of `in_progress`, `completed`, `failed`, `cancelled`.

Example request

curl ⚡ Copy

```
1 curl https://api.openai.com/v1/vector_stores
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -H "OpenAI-Beta: assistants=v2"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "file-abc123",
6       "object": "vector_store.file",
7       "created_at": 1699061776,
8       "vector_store_id": "vs_abc123"
9     },
10    {
11      "id": "file-abc456",
12      "object": "vector_store.file",
13      "created_at": 1699061776,
14      "vector_store_id": "vs_abc123"
15    }
16  ],
17  "first_id": "file-abc123",
18  "last_id": "file-abc456",
19  "has_more": false
20 }
```

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

A list of [vector store file](#) objects.

Retrieve vector store file

```
GET https://api.openai.com/v1/vector_stores/{vector_store_id}/files/{file_id}
```

Retrieves a vector store file.

Path parameters

file_id string Required

The ID of the file being retrieved.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2"
```

Response

⌚

```
1 {
2   "id": "file-abc123",
```

vector_store_id string Required

The ID of the vector store that the file belongs to.

Returns

The [vector store file](#) object.

```

3   "object": "vector_store.file",
4   "created_at": 1699061776,
5   "vector_store_id": "vs_abcd",
6   "status": "completed",
7   "last_error": null
8 }
```

Retrieve vector store file content

```
GET https://api.openai.com/v1/vector_stores/{vector_store_id}/files/{file_id}/content
```

Retrieve the parsed contents of a vector store file.

Path parameters**file_id** string Required

The ID of the file within the vector store.

vector_store_id string Required

The ID of the vector store.

Returns**Example request**
curl ⌂

```

1 curl \
2 https://api.openai.com/v1/vector_stores/vs_a
3 -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```

1 {
2   "file_id": "file-abc123",
3   "filename": "example.txt",
4   "attributes": {"key": "value"},
5   "content": [
6     {"type": "text", "text": "..."},
7     ...
8
9 }
```

The parsed contents of the specified vector store file.

Update vector store file attributes

```
POST https://api.openai.com/v1/vector_stores/{vector_store_id}/file  
s/{file_id}
```

Update attributes on a vector store file.

Path parameters

file_id string Required

The ID of the file to update attributes.

vector_store_id string Required

The ID of the vector store the file belongs to.

Request body

attributes map Required

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard. Keys are strings with a maximum length of 64

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/vector_stores  
2   -H "Authorization: Bearer $OPENAI_API_KEY"  
3   -H "Content-Type: application/json" \  
4   -d '{"attributes": {"key1": "value1", "key2": "value2", "key3": "value3", "key4": "value4", "key5": "value5", "key6": "value6", "key7": "value7", "key8": "value8", "key9": "value9", "key10": "value10", "key11": "value11", "key12": "value12", "key13": "value13", "key14": "value14", "key15": "value15", "key16": "value16"}' -X POST
```

Response

```
1 {  
2   "id": "file-abc123",  
3   "object": "vector_store.file",  
4   "usage_bytes": 1234,  
5   "created_at": 1699061776,  
6   "vector_store_id": "vs_abcd",  
7   "status": "completed",  
8   "last_error": null,  
9   "chunking_strategy": {...},  
10  "attributes": {"key1": "value1", "key2": "value2", "key3": "value3", "key4": "value4", "key5": "value5", "key6": "value6", "key7": "value7", "key8": "value8", "key9": "value9", "key10": "value10", "key11": "value11", "key12": "value12", "key13": "value13", "key14": "value14", "key15": "value15", "key16": "value16"}  
11 }
```

characters. Values are strings with a maximum length of 512 characters, booleans, or numbers.

Returns

The updated [vector store file](#) object.

Delete vector store file

```
DELETE https://api.openai.com/v1/vector_stores/{vector_store_id}/files/{file_id}
```

Delete a vector store file. This will remove the file from the vector store but the file itself will not be deleted. To delete the file, use the [delete file](#) endpoint.

Path parameters

file_id string Required

The ID of the file to delete.

vector_store_id string Required

The ID of the vector store that the file belongs to.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -X DELETE
```

Response

⌚

```
1 {
2   id: "file-abc123",
3   object: "vector_store.file.deleted",
4   deleted: true
5 }
```

Returns

Deletion status

The vector store file object Beta

A list of files attached to a vector store.

attributes map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard. Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters, booleans, or numbers.

chunking_strategy object

The strategy used to chunk the file.

▽ Show possible types

created_at integer

The Unix timestamp (in seconds) for when the vector store file was created.

id string

The identifier, which can be referenced in API endpoints.

OBJECT The vector store file object

```
1  {
2    "id": "file-abc123",
3    "object": "vector_store.file",
4    "usage_bytes": 1234,
5    "created_at": 1698107661,
6    "vector_store_id": "vs_abc123",
7    "status": "completed",
8    "last_error": null,
9    "chunking_strategy": {
10      "type": "static",
11      "static": {
12        "max_chunk_size_tokens": 800,
13        "chunk_overlap_tokens": 400
14      }
15    }
16 }
```

last_error object or null

The last error associated with this vector store file. Will be `null` if there are no errors.

▼ Show properties

object string

The object type, which is always `vector_store.file`.

status string

The status of the vector store file, which can be either `in_progress`, `completed`, `cancelled`, or `failed`. The status `completed` indicates that the vector store file is ready for use.

usage_bytes integer

The total vector store usage in bytes. Note that this may be different from the original file size.

vector_store_id string

The ID of the [vector store](#) that the [File](#) is attached to.

Vector store file batches

Vector store file batches represent operations to add multiple files to a vector store. Related guide:

[File Search](#)

Create vector store file batch

POST https://api.openai.com/v1/vector_stores/{vector_store_id}/file_batches

Create a vector store file batch.

Path parameters

vector_store_id string Required

The ID of the vector store for which to create a File Batch.

Request body

file_ids array Required

A list of [File](#) IDs that the vector store should use. Useful for tools like [file_search](#) that can access files.

attributes map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard. Keys are strings with a maximum length of 64

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json"
4   -H "OpenAI-Beta: assistants=v2"
5   -d '{
6     "file_ids": ["file-abc123", "file-abc456"]
7   }'
```

Response

```
1 {
2   "id": "vsfb_abc123",
3   "object": "vector_store.file_batch",
4   "created_at": 1699061776,
5   "vector_store_id": "vs_abc123",
6   "status": "in_progress",
7   "file_counts": {
8     "in_progress": 1,
9     "completed": 1,
10    "failed": 0,
11    "cancelled": 0,
12    "total": 0,
```

13 }

characters. Values are strings with a maximum length of 512 characters, booleans, or numbers.

chunking_strategy object Optional

The chunking strategy used to chunk the file(s). If not set, will use the `auto` strategy.

▼ Show possible types

Returns

A [vector store file batch](#) object.

Retrieve vector store file batch

```
GET https://api.openai.com/v1/vector_stores/{vector_store_id}/file_batches/{batch_id}
```

Retrieves a vector store file batch.

Path parameters

batch_id string Required

The ID of the file batch being retrieved.

vector_store_id string Required

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/vector_stores  
2   -H "Authorization: Bearer $OPENAI_API_KEY"  
3   -H "Content-Type: application/json" \  
4   -H "OpenAI-Beta: assistants=v2"
```

Response

```
1 {  
2   "id": "vsfb_abc123",  
3   "object": "vector_store.file_batch",  
4   "created_at": 1699061776,
```

The ID of the vector store that the file batch belongs to.

Returns

The [vector store file batch](#) object.

```

5   "vector_store_id": "vs_abc123",
6   "status": "in_progress",
7   "file_counts": {
8     "in_progress": 1,
9     "completed": 1,
10    "failed": 0,
11    "cancelled": 0,
12    "total": 0,
13  }
14 }
```

Cancel vector store file batch

```
POST https://api.openai.com/v1/vector_stores/{vector_store_id}/file_batches/{batch_id}/cancel
```

Cancel a vector store file batch. This attempts to cancel the processing of files in this batch as soon as possible.

Path parameters

batch_id string Required

The ID of the file batch to cancel.

vector_store_id string Required

The ID of the vector store that the file batch belongs to.

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -X POST
```

Response

⌚

```

1  {
2    "id": "vsfb_abc123",
3    "object": "vector_store.file_batch",
4    "created_at": 1699061776,
5    "vector_store_id": "vs_abc123",
6    "status": "in_progress",
7    "file_counts": {
```

Returns

The modified vector store file batch object.

```

8     "in_progress": 12,
9     "completed": 3,
10    "failed": 0,
11    "cancelled": 0,
12    "total": 15,
13  }
14 }
```

List vector store files in a batch

```
GET https://api.openai.com/v1/vector_stores/{vector_store_id}/file_batches/{batch_id}/files
```

Returns a list of vector store files in a batch.

Path parameters

batch_id string Required

The ID of the file batch that the files belong to.

vector_store_id string Required

The ID of the vector store that the files belong to.

Query parameters

after string Optional

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/vector_stores
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2"
```

Response

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "id": "file-abc123",
6        "object": "vector_store.file",
7        "created_at": 1699061776,
8        "vector_store_id": "vs_abc123"
9      },
10     {
11       "id": "file-abc456",
```

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

filter string Optional

Filter by file status. One of `in_progress`, `completed`, `failed`, `cancelled`.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

A list of vector store file objects.

```

12     "object": "vector_store.file",
13     "created_at": 1699061776,
14     "vector_store_id": "vs_abc123"
15   }
16 ],
17   "first_id": "file-abc123",
18   "last_id": "file-abc456",
19   "has_more": false
20 }
```

The vector store files batch object Beta

A batch of files attached to a vector store.

created_at integer

The Unix timestamp (in seconds) for when the vector store files batch was created.

file_counts object

✓ Show properties

id string

The identifier, which can be referenced in API endpoints.

object string

The object type, which is always `vector_store.file_batch`.

status string

The status of the vector store files batch, which can be either `in_progress`, `completed`, `cancelled` or `failed`.

vector_store_id string

The ID of the vector store that the File is attached to.

OBJECT The vector store files batch object

```
1  {
2    "id": "vsfb_123",
3    "object": "vector_store.files_batch",
4    "created_at": 1698107661,
5    "vector_store_id": "vs_abc123",
6    "status": "completed",
7    "file_counts": {
8      "in_progress": 0,
9      "completed": 100,
10     "failed": 0,
11     "cancelled": 0,
12     "total": 100
13   }
14 }
```

Containers

Create and manage containers for use with the Code Interpreter tool.

Create container

POST <https://api.openai.com/v1/containers>

Create Container

Request body

name string Required

Name of the container to create.

expires_after object Optional

Container expiration time in seconds relative to the 'anchor' time.

✓ Show properties

file_ids array Optional

IDs of files to copy to the container.

Returns

The created [container](#) object.

List containers

GET <https://api.openai.com/v1/containers>

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/containers \
2   -H "Authorization: Bearer $OPENAI_API_KEY" \
3   -H "Content-Type: application/json" \
4   -d '{
5     "name": "My Container"
6   }'
```

Response

```
1 {
2   "id": "cntr_682e30645a488191b6363a0cbef",
3   "object": "container",
4   "created_at": 1747857508,
5   "status": "running",
6   "expires_after": {
7     "anchor": "last_active_at",
8     "minutes": 20
9   },
10  "last_active_at": 1747857508,
11  "name": "My Container"
12 }
```

List Containers

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

a list of [container](#) objects.

```
1 curl https://api.openai.com/v1/containers \
2 -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "cntr_682dfebaacac8198bbfe9c2",
6       "object": "container",
7       "created_at": 1747844794,
8       "status": "running",
9       "expires_after": {
10         "anchor": "last_active_at",
11         "minutes": 20
12       },
13       "last_active_at": 1747844794,
14       "name": "My Container"
15     }
16   ],
17   "first_id": "container_123",
18   "last_id": "container_123",
19   "has_more": false
20 }
```

Retrieve container

GET https://api.openai.com/v1/containers/{container_id}

Retrieve Container

Path parameters

container_id string Required

Returns

The [container](#) object.

Example request

curl ⚡ ↗

```
1 curl https://api.openai.com/v1/containers/cn
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "id": "cntr_682dfebaacac8198bbfe9c2474f",
3   "object": "container",
4   "created_at": 1747844794,
5   "status": "running",
6   "expires_after": {
7     "anchor": "last_active_at",
8     "minutes": 20
9   },
10  "last_active_at": 1747844794,
11  "name": "My Container"
12 }
```

Delete a container

DELETE https://api.openai.com/v1/containers/{container_id}

Delete Container

Path parameters

Example request

curl ⚡ ↗

```
1 curl -X DELETE https://api.openai.com/v1/containers/cn
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

container_id string Required

The ID of the container to delete.

Returns

Deletion Status

```

1 {
2   "id": "cntr_682dfebaacac8198bbfe9c2474fb"
3   "object": "container.deleted",
4   "deleted": true
5 }
```

The container object

created_at integer

Unix timestamp (in seconds) when the container was created.

expires_after object

The container will expire after this time period. The anchor is the reference point for the expiration. The minutes is the number of minutes after the anchor before the container expires.

✓ Show properties

id string

Unique identifier for the container.

name string

Name of the container.

OBJECT The container object

```

1 {
2   "id": "cntr_682dfebaacac8198bbfe9c2474fb"
3   "object": "container",
4   "created_at": 1747844794,
5   "status": "running",
6   "expires_after": {
7     "anchor": "last_active_at",
8     "minutes": 20
9   },
10  "last_active_at": 1747844794,
11  "name": "My Container"
12 }
```

object string

The type of this object.

status string

Status of the container (e.g., active, deleted).

Container Files

Create and manage container files for use with the Code Interpreter tool.

Create container file

POST https://api.openai.com/v1/containers/{container_id}/files

Create a Container File

You can send either a multipart/form-data request with the raw file content, or a JSON request with a file ID.

Path parameters

container_id string Required

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/containers/cn  
2   -H "Authorization: Bearer $OPENAI_API_KEY"  
3   -F file="@example.txt"
```

Response

```
1 {  
2   "id": "cfile_682e0e8a43c88191a7978f477a09b",  
3   "object": "container_file",  
4   "created_at": 1747848842,
```

Request body

file file Optional

The File object (not file name) to be uploaded.

```

5   "bytes": 880,
6   "container_id": "cntr_682e0e7318108198aa78
7   "path": "/mnt/data/88e12fa445d32636f190a0b
8   "source": "user"
9 }
```

file_id string Optional

Name of the file to create.

Returns

The created [container file](#) object.

List container files

GET https://api.openai.com/v1/containers/{container_id}/files

List Container files

Path parameters

container_id string Required

Query parameters

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/containers/cn
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```

1  {
2    "object": "list",
3    "data": [
4      {
```

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

a list of [container file](#) objects.

Retrieve container file

```
GET https://api.openai.com/v1/containers/{container_id}/files/{file_id}
```

Retrieve Container File

```

5   "id": "cfile_682e0e8a43c88191a7
6   "object": "container.file",
7   "created_at": 1747848842,
8   "bytes": 880,
9   "container_id": "cntr_682e0e731
10  "path": "/mnt/data/88e12fa445d3
11  "source": "user"
12  }
13 ],
14  "first_id": "cfile_682e0e8a43c88191a797
15  "has_more": false,
16  "last_id": "cfile_682e0e8a43c88191a7978
17 }
```

Example request

[curl](#) [Copy](#)

```
1 curl https://api.openai.com/v1/containers/co
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Path parameters

container_id string Required

file_id string Required

Returns

The container_file object.

Response

```
1 {
2   "id": "cfile_682e0e8a43c88191a7978f477a0",
3   "object": "container_file",
4   "created_at": 1747848842,
5   "bytes": 880,
6   "container_id": "cntr_682e0e7318108198aa",
7   "path": "/mnt/data/88e12fa445d32636f190a",
8   "source": "user"
9 }
```

Retrieve container file content

```
GET https://api.openai.com/v1/containers/{container_id}/files/{file_id}/content
```

Retrieve Container File Content

Path parameters

container_id string Required

file_id string Required

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/containers/co
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

<binary content of the file>

Returns

The contents of the container file.

Delete a container file

```
DELETE https://api.openai.com/v1/containers/{container_id}/files/{file_id}
```

Delete Container File

Path parameters

container_id string Required

file_id string Required

Returns

Deletion Status

Example request

curl ⌂ 🔗

```
1 curl -X DELETE https://api.openai.com/v1/containers/{container_id}/files/{file_id}
2   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response

```
1 {
2   "id": "cfile_682e0e8a43c88191a7978f477a0",
3   "object": "container.file.deleted",
4   "deleted": true
5 }
```

The container file object

bytes integer

Size of the file in bytes.

container_id string

The container this file belongs to.

created_at integer

Unix timestamp (in seconds) when the file was created.

id string

Unique identifier for the file.

object string

The type of this object (`container.file`).

path string

Path of the file in the container.

source string

Source of the file (e.g., `user` , `assistant`).

OBJECT The container file object

```
1 {  
2   "id": "cfile_682e0e8a43c88191a7978f477a0  
3   "object": "container.file",  
4   "created_at": 1747848842,  
5   "bytes": 880,  
6   "container_id": "cntr_682e0e7318108198aa  
7   "path": "/mnt/data/88e12fa445d32636f190a  
8   "source": "user"  
9 }
```

Communicate with a GPT-4o class model in real time using WebRTC or WebSockets. Supports text and audio inputs and outputs, along with audio transcriptions. [Learn more about the Realtime API.](#)

Session tokens

REST API endpoint to generate ephemeral session tokens for use in client-side applications.

Create session

POST <https://api.openai.com/v1/realtime/sessions>

Create an ephemeral API token for use in client-side applications with the Realtime API. Can be configured with the same session parameters as the `session.update` client event.

It responds with a session object, plus a `client_secret` key which contains a usable ephemeral API token that can be used to authenticate browser clients for the Realtime API.

Request body

client_secret object Optional

Configuration options for the generated client secret.

▼ Show properties

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/realtime/sessions
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5   "model": "gpt-4o-realtime-preview",
6   "modalities": ["audio", "text"],
7   "instructions": "You are a friendly assistant."
8 }'
```

Response

```
1 {
2   "id": "sess_001",
3   "object": "realtime.session",
4   "model": "gpt-4o-realtime-preview",
```

input_audio_format string Optional Defaults to pcm16

The format of input audio. Options are `pcm16`, `g711_ulaw`, or `g711_alaw`. For `pcm16`, input audio must be 16-bit PCM at a 24kHz sample rate, single channel (mono), and little-endian byte order.

input_audio_noise_reduction object Optional Defaults to null

Configuration for input audio noise reduction. This can be set to `null` to turn off.

Noise reduction filters audio added to the input audio buffer before it is sent to VAD and the model. Filtering the audio can improve VAD and turn detection accuracy (reducing false positives) and model performance by improving perception of the input audio.

▼ Show properties

input_audio_transcription object Optional

Configuration for input audio transcription, defaults to off and can be set to `null` to turn off once on. Input audio transcription is not native to the model, since the model consumes audio directly. Transcription runs asynchronously through [the /audio/transcriptions endpoint](#) and should be treated as guidance of input audio content rather than precisely what the model heard. The client can optionally set the language and prompt for transcription, these offer additional guidance to the transcription service.

▼ Show properties

instructions string Optional

The default system instructions (i.e. system message) prepended to model calls. This field allows the client to guide the model on desired responses. The model can be instructed on response content and format, (e.g. "be extremely succinct", "act friendly", "here are examples of good responses") and on audio behavior (e.g. "talk quickly", "inject emotion into your voice", "laugh frequently"). The instructions are not

```

5   "modalities": ["audio", "text"],
6   "instructions": "You are a friendly assis
7   "voice": "alloy",
8   "input_audio_format": "pcm16",
9   "output_audio_format": "pcm16",
10  "input_audio_transcription": {
11    "model": "whisper-1"
12  },
13  "turn_detection": null,
14  "tools": [],
15  "tool_choice": "none",
16  "temperature": 0.7,
17  "max_response_output_tokens": 200,
18  "speed": 1.1,
19  "tracing": "auto",
20  "client_secret": {
21    "value": "ek_abc123",
22    "expires_at": 1234567890
23  }
24 }
```

guaranteed to be followed by the model, but they provide guidance to the model on the desired behavior.

Note that the server sets default instructions which will be used if this field is not set and are visible in the `session.created` event at the start of the session.

max_response_output_tokens integer or "inf" Optional

Maximum number of output tokens for a single assistant response, inclusive of tool calls. Provide an integer between 1 and 4096 to limit output tokens, or `inf` for the maximum available tokens for a given model. Defaults to `inf`.

modalities Optional

The set of modalities the model can respond with. To disable audio, set this to `["text"]`.

model string Optional

The Realtime model used for this session.

output_audio_format string Optional Defaults to `pcm16`

The format of output audio. Options are `pcm16`, `g711_ulaw`, or `g711_alaw`. For `pcm16`, output audio is sampled at a rate of 24kHz.

speed number Optional Defaults to 1

The speed of the model's spoken response. 1.0 is the default speed. 0.25 is the minimum speed. 1.5 is the maximum speed. This value can only be changed in between model turns, not while a response is in progress.

temperature number Optional Defaults to 0.8

Sampling temperature for the model, limited to `[0.6, 1.2]`. For audio models a temperature of 0.8 is highly recommended for best performance.

tool_choice string Optional Defaults to auto

How the model chooses tools. Options are `auto`, `none`, `required`, or specify a function.

tools array Optional

Tools (functions) available to the model.

▼ Show properties

tracing "auto" or object Optional

Configuration options for tracing. Set to null to disable tracing. Once tracing is enabled for a session, the configuration cannot be modified.

`auto` will create a trace for the session with default values for the workflow name, group id, and metadata.

▼ Show possible types

turn_detection object Optional

Configuration for turn detection, either Server VAD or Semantic VAD. This can be set to `null` to turn off, in which case the client must manually trigger model response.

Server VAD means that the model will detect the start and end of speech based on audio volume and respond at the end of user speech. Semantic VAD is more advanced and uses a turn detection model (in conjunction with VAD) to semantically estimate whether the user has finished speaking, then dynamically sets a timeout based on this probability. For example, if user audio trails off with "uhhm", the model will score a low probability of turn end and wait longer for the user to continue speaking. This can be useful for more natural conversations, but may have a higher latency.

▼ Show properties

voice string Optional

The voice the model uses to respond. Voice cannot be changed during the session once the model has responded with audio at least once. Current voice options are `alloy`, `ash`, `ballad`, `coral`, `echo`, `sage`, `shimmer`, and `verse`.

Returns

The created Realtime session object, plus an ephemeral key

Create transcription session

POST https://api.openai.com/v1/realtime/transcription_sessions

Create an ephemeral API token for use in client-side applications with the Realtime API specifically for realtime transcriptions. Can be configured with the same session parameters as the `transcription_session.update` client event.

It responds with a session object, plus a `client_secret` key which contains a usable ephemeral API token that can be used to authenticate browser clients for the Realtime API.

Request body

`client_secret` object Optional

Configuration options for the generated client secret.

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/realtime/transcription_sessions
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -d '{}'
```

Response

```
1 {
2   "id": "sess_BBwZc7cFV3XizEyKGDCGL",
3   "object": "realtime.transcription_session",
4   "modalities": ["audio", "text"],
5   "turn_detection": {
6     "type": "server_vad",
7     "threshold": 0.5,
```

✓ Show properties

include array Optional

The set of items to include in the transcription. Current available items are:

null.

input_audio_format string Optional Defaults to pcm16

The format of input audio. Options are `pcm16`, `g711_ulaw`, or `g711_alaw`. For `pcm16`, input audio must be 16-bit PCM at a 24kHz sample rate, single channel (mono), and little-endian byte order.

input_audio_noise_reduction object Optional Defaults to null

Configuration for input audio noise reduction. This can be set to `null` to turn off.

Noise reduction filters audio added to the input audio buffer before it is sent to VAD and the model. Filtering the audio can improve VAD and turn detection accuracy (reducing false positives) and model performance by improving perception of the input audio.

✓ Show properties

input_audio_transcription object Optional

Configuration for input audio transcription. The client can optionally set the language and prompt for transcription, these offer additional guidance to the transcription service.

✓ Show properties

modalities Optional

The set of modalities the model can respond with. To disable audio, set this to `["text"]`.

turn_detection object Optional

```

8     "prefix_padding_ms": 300,
9     "silence_duration_ms": 200
10    },
11    "input_audio_format": "pcm16",
12    "input_audio_transcription": {
13      "model": "gpt-4o-transcribe",
14      "language": null,
15      "prompt": ""
16    },
17    "client_secret": null
18 }
```

Configuration for turn detection, either Server VAD or Semantic VAD. This can be set to

`null` to turn off, in which case the client must manually trigger model response.

Server VAD means that the model will detect the start and end of speech based on audio volume and respond at the end of user speech. Semantic VAD is more advanced and uses a turn detection model (in conjunction with VAD) to semantically estimate whether the user has finished speaking, then dynamically sets a timeout based on this probability. For example, if user audio trails off with "uhhm", the model will score a low probability of turn end and wait longer for the user to continue speaking. This can be useful for more natural conversations, but may have a higher latency.

✓ Show properties

Returns

The created [Realtime transcription session object](#), plus an ephemeral key

The session object

A new Realtime session configuration, with an ephemeral key. Default TTL for keys is one minute.

`client_secret` object

Ephemeral key returned by the API.

✓ Show properties

OBJECT The session object

```
1  {
2    "id": "sess_001",
3    "object": "realtime.session",
4    "model": "gpt-4o-realtime-preview",
5    "modalities": ["audio", "text"],
6    "instructions": "You are a friendly assis
7    "voice": "alloy",
```

input_audio_format string

The format of input audio. Options are `pcm16`, `g711_ulaw`, or `g711_alaw`.

input_audio_transcription object

Configuration for input audio transcription, defaults to off and can be set to `null` to turn off once on. Input audio transcription is not native to the model, since the model consumes audio directly. Transcription runs asynchronously and should be treated as rough guidance rather than the representation understood by the model.

▽ Show properties

instructions string

The default system instructions (i.e. system message) prepended to model calls. This field allows the client to guide the model on desired responses. The model can be instructed on response content and format, (e.g. "be extremely succinct", "act friendly", "here are examples of good responses") and on audio behavior (e.g. "talk quickly", "inject emotion into your voice", "laugh frequently"). The instructions are not guaranteed to be followed by the model, but they provide guidance to the model on the desired behavior.

Note that the server sets default instructions which will be used if this field is not set and are visible in the `session.created` event at the start of the session.

max_response_output_tokens integer or "inf"

Maximum number of output tokens for a single assistant response, inclusive of tool calls. Provide an integer between 1 and 4096 to limit output tokens, or `inf` for the maximum available tokens for a given model. Defaults to `inf`.

modalities

The set of modalities the model can respond with. To disable audio, set this to `["text"]`.

```

8   "input_audio_format": "pcm16",
9   "output_audio_format": "pcm16",
10  "input_audio_transcription": {
11    "model": "whisper-1"
12  },
13  "turn_detection": null,
14  "tools": [],
15  "tool_choice": "none",
16  "temperature": 0.7,
17  "speed": 1.1,
18  "tracing": "auto",
19  "max_response_output_tokens": 200,
20  "client_secret": {
21    "value": "ek_abc123",
22    "expires_at": 1234567890
23  }
24 }
```

output_audio_format string

The format of output audio. Options are `pcm16`, `g711_ulaw`, or `g711_alaw`.

speed number

The speed of the model's spoken response. 1.0 is the default speed. 0.25 is the minimum speed. 1.5 is the maximum speed. This value can only be changed in between model turns, not while a response is in progress.

temperature number

Sampling temperature for the model, limited to [0.6, 1.2]. Defaults to 0.8.

tool_choice string

How the model chooses tools. Options are `auto`, `none`, `required`, or specify a function.

tools array

Tools (functions) available to the model.

▼ Show properties

tracing "auto" or object

Configuration options for tracing. Set to null to disable tracing. Once tracing is enabled for a session, the configuration cannot be modified.

`auto` will create a trace for the session with default values for the workflow name, group id, and metadata.

▼ Show possible types

turn_detection object

Configuration for turn detection. Can be set to `null` to turn off. Server VAD means that the model will detect the start and end of speech based on audio volume and respond at the end of user speech.

✓ Show properties

voice string

The voice the model uses to respond. Voice cannot be changed during the session once the model has responded with audio at least once. Current voice options are `alloy`, `ash`, `ballad`, `coral`, `echo`, `sage`, `shimmer`, and `verse`.

The transcription session object

A new Realtime transcription session configuration.

When a session is created on the server via REST API, the session object also contains an ephemeral key. Default TTL for keys is 10 minutes. This property is not present when a session is updated via the WebSocket API.

client_secret object

Ephemeral key returned by the API. Only present when the session is created on the server via REST API.

✓ Show properties

input_audio_format string

The format of input audio. Options are `pcm16`, `g711_ulaw`, or `g711_alaw`.

OBJECT The transcription session object

```

1  {
2    "id": "sess_BBwZc7cFV3XizEyKGDCGL",
3    "object": "realtime.transcription_session",
4    "expires_at": 1742188264,
5    "modalities": ["audio", "text"],
6    "turn_detection": {
7      "type": "server_vad",
8      "threshold": 0.5,
9      "prefix_padding_ms": 300,
10     "silence_duration_ms": 200
11   },
12   "input_audio_format": "pcm16",
13   "input_audio_transcription": {
14     "model": "gpt-4o-transcribe",

```

input_audio_transcription object

Configuration of the transcription model.

▼ Show properties

```
15 "language": null,  
16 "prompt": ""  
17 },  
18 "client_secret": null  
19 }
```

modalities

The set of modalities the model can respond with. To disable audio, set this to ["text"].

turn_detection object

Configuration for turn detection. Can be set to `null` to turn off. Server VAD means that the model will detect the start and end of speech based on audio volume and respond at the end of user speech.

▼ Show properties

Client events

These are events that the OpenAI Realtime WebSocket server will accept from the client.

session.update

Send this event to update the session's default configuration. The client may send this event at any time to update any field, except for `voice`. However, note that once a session has been initialized with a particular `model`, it can't be changed to another model using `session.update`.

When the server receives a `session.update`, it will respond with a `session.updated` event showing the full, effective configuration. Only the fields that are present are updated. To clear a field like `instructions`, pass an empty string.

event_id string

Optional client-generated ID used to identify this event.

session object

Realtime session object configuration.

▽ Show properties

type string

The event type, must be `session.update`.

OBJECT `session.update`

```

1  {
2      "event_id": "event_123",
3      "type": "session.update",
4      "session": {
5          "modalities": ["text", "audio"],
6          "instructions": "You are a helpful",
7          "voice": "sage",
8          "input_audio_format": "pcm16",
9          "output_audio_format": "pcm16",
10         "input_audio_transcription": {
11             "model": "whisper-1"
12         },
13         "turn_detection": {
14             "type": "server_vad",
15             "threshold": 0.5,
16             "prefix_padding_ms": 300,
17             "silence_duration_ms": 500,
18             "create_response": true
19         },
20         "tools": [
21             {
22                 "type": "function",
23                 "name": "get_weather",
24                 "description": "Get the cur",
25                 "parameters": {
26                     "type": "object",
27                     "properties": {
28                         "location": { "type": "string" },
29                     },
30                     "required": ["location"]
31                 }
32             }
33         ],

```

input_audio_buffer.append

Send this event to append audio bytes to the input audio buffer. The audio buffer is temporary storage you can write to and later commit. In Server VAD mode, the audio buffer is used to detect speech and the server will decide when to commit. When Server VAD is disabled, you must commit the audio buffer manually.

The client may choose how much audio to place in each event up to a maximum of 15 MiB, for example streaming smaller chunks from the client may allow the VAD to be more responsive. Unlike made other client events, the server will not send a confirmation response to this event.

audio string

Base64-encoded audio bytes. This must be in the format specified by the `input_audio_format` field in the session configuration.

event_id string

Optional client-generated ID used to identify this event.

type string

The event type, must be `input_audio_buffer.append`.

```
34     "tool_choice": "auto",
35     "temperature": 0.8,
36     "max_response_output_tokens": "inf"
37     "speed": 1.1,
38     "tracing": "auto"
39 }
```

OBJECT `input_audio_buffer.append` 🔗

```
1 {
2   "event_id": "event_456",
3   "type": "input_audio_buffer.append",
4   "audio": "Base64EncodedAudioData"
5 }
```

input_audio_buffer.commit

Send this event to commit the user input audio buffer, which will create a new user message item in the conversation. This event will produce an error if the input audio buffer is empty. When in Server VAD mode, the client does not need to send this event, the server will commit the audio buffer automatically.

Committing the input audio buffer will trigger input audio transcription (if enabled in session configuration), but it will not create a response from the model. The server will respond with an `input_audio_buffer.committed` event.

event_id string

Optional client-generated ID used to identify this event.

type string

The event type, must be `input_audio_buffer.commit`.

input_audio_buffer.clear

Send this event to clear the audio bytes in the buffer. The server will respond with an `input_audio_buffer.cleared` event.

event_id string

OBJECT `input_audio_buffer.commit` 

```
1 {  
2   "event_id": "event_789",  
3   "type": "input_audio_buffer.commit"  
4 }
```

OBJECT `input_audio_buffer.clear` 

```
1 {  
2   "event_id": "event_012",
```

Optional client-generated ID used to identify this event.

type string

The event type, must be `input_audio_buffer.clear`.

3

```
"type": "input_audio_buffer.clear"
```

conversation.item.create

Add a new Item to the Conversation's context, including messages, function calls, and function call responses. This event can be used both to populate a "history" of the conversation and to add new items mid-stream, but has the current limitation that it cannot populate assistant audio messages.

If successful, the server will respond with a `conversation.item.created` event, otherwise an `error` event will be sent.

event_id string

Optional client-generated ID used to identify this event.

item object

The item to add to the conversation.

OBJECT `conversation.item.create`

```

1  {
2      "event_id": "event_345",
3      "type": "conversation.item.create",
4      "previous_item_id": null,
5      "item": {
6          "id": "msg_001",
7          "type": "message",
8          "role": "user",
9          "content": [
10             {
11                 "type": "input_text",
12                 "text": "Hello, how are you"
13             }
14         ]
}
```

previous_item_id string

The ID of the preceding item after which the new item will be inserted. If not set, the new item will be appended to the end of the conversation. If set to `root`, the new item will be added to the beginning of the conversation. If set to an existing ID, it allows an item to be inserted mid-conversation. If the ID cannot be found, an error will be returned and the item will not be added.

type string

The event type, must be `conversation.item.create`.

conversation.item.retrieve

Send this event when you want to retrieve the server's representation of a specific item in the conversation history. This is useful, for example, to inspect user audio after noise cancellation and VAD. The server will respond with a `conversation.item.retrieved` event, unless the item does not exist in the conversation history, in which case the server will respond with an error.

event_id string

Optional client-generated ID used to identify this event.

item_id stringOBJECT `conversation.item.retrieve`

```
1 {  
2   "event_id": "event_901",  
3   "type": "conversation.item.retrieve",  
4   "item_id": "msg_003"  
5 }
```

The ID of the item to retrieve.

type string

The event type, must be `conversation.item.retrieve`.

conversation.item.truncate

Send this event to truncate a previous assistant message's audio. The server will produce audio faster than realtime, so this event is useful when the user interrupts to truncate audio that has already been sent to the client but not yet played. This will synchronize the server's understanding of the audio with the client's playback.

Truncating audio will delete the server-side text transcript to ensure there is not text in the context that hasn't been heard by the user.

If successful, the server will respond with a `conversation.item.truncated` event.

audio_end_ms integer

Inclusive duration up to which audio is truncated, in milliseconds. If the `audio_end_ms` is greater than the actual audio duration, the server will respond with an error.

content_index integer

The index of the content part to truncate. Set this to 0.

OBJECT `conversation.item.truncate`

```
1 {  
2   "event_id": "event_678",  
3   "type": "conversation.item.truncate",  
4   "item_id": "msg_002",  
5   "content_index": 0,  
6   "audio_end_ms": 1500  
7 }
```

event_id string

Optional client-generated ID used to identify this event.

item_id string

The ID of the assistant message item to truncate. Only assistant message items can be truncated.

type string

The event type, must be `conversation.item.truncate`.

conversation.item.delete

Send this event when you want to remove any item from the conversation history. The server will respond with a `conversation.item.deleted` event, unless the item does not exist in the conversation history, in which case the server will respond with an error.

event_id string

Optional client-generated ID used to identify this event.

item_id string

The ID of the item to delete.

type stringOBJECT `conversation.item.delete` 

```
1 {
2   "event_id": "event_901",
3   "type": "conversation.item.delete",
4   "item_id": "msg_003"
5 }
```

The event type must be `conversation.item.delete`.

response.create

This event instructs the server to create a Response, which means triggering model inference. When in Server VAD mode, the server will create Responses automatically.

A Response will include at least one Item, and may have two, in which case the second will be a function call. These Items will be appended to the conversation history.

The server will respond with a `response.created` event, events for Items and content created, and finally a `response.done` event to indicate the Response is complete.

The `response.create` event includes inference configuration like `instructions`, and `temperature`. These fields will override the Session's configuration for this Response only.

event_id string

Optional client-generated ID used to identify this event.

OBJECT `response.create`

```
1  {
2      "event_id": "event_234",
3      "type": "response.create",
4      "response": {
5          "modalities": ["text", "audio"],
6          "instructions": "Please assist the",
7          "voice": "sage",
8          "output_audio_format": "pcm16",
9          "tools": [
10              {
11                  "type": "function",
12                  "name": "calculate_sum",
13                  "description": "Calculates",
14                  "parameters": {
15                      "type": "object",
16                      "properties": {
17                          "a": { "type": "num",
18                          "b": { "type": "num
19                          },
20                          "required": ["a", "b"]
21                      }
22                  }
23              }
24          ]
25      }
26  }
```

response object

Create a new Realtime response with these parameters

✓ Show properties

type string

The event type, must be `response.create`.

```

22
23
24
25
26
27
28
      },
      ],
      "tool_choice": "auto",
      "temperature": 0.8,
      "max_output_tokens": 1024
    }
  }
```

response.cancel

Send this event to cancel an in-progress response. The server will respond with a `response.cancelled` event or an error if there is no response to cancel.

event_id string

Optional client-generated ID used to identify this event.

OBJECT `response.cancel`

```

1 {
2   "event_id": "event_567",
3   "type": "response.cancel"
4 }
```

response_id string

A specific response ID to cancel - if not provided, will cancel an in-progress response in the default conversation.

type string

The event type, must be `response.cancel`.

transcription_session.update

Send this event to update a transcription session.

event_id string

Optional client-generated ID used to identify this event.

session object

Realtime transcription session object configuration.

▽ Show properties

type string

The event type, must be `transcription_session.update`.

OBJECT `transcription_session.update`

```
1  {
2    "type": "transcription_session.update",
3    "session": {
4      "input_audio_format": "pcm16",
5      "input_audio_transcription": {
6        "model": "gpt-4o-transcribe",
7        "prompt": "",
8        "language": ""
9      },
10     "turn_detection": {
11       "type": "server_vad",
12       "threshold": 0.5,
13       "prefix_padding_ms": 300,
14       "silence_duration_ms": 500,
15       "create_response": true,
16     },
17     "input_audio_noise_reduction": {
18       "type": "near_field"
19     },
20     "include": [
21       "item.input_audio_transcription.logpr
22     ]
23   }
24 }
```

output_audio_buffer.clear

WebRTC Only: Emit to cut off the current audio response. This will trigger the server to stop generating audio and emit a

`output_audio_buffer.cleared` event. This event should be preceded by a `response.cancel` client event to stop the generation of the current response. [Learn more.](#)

OBJECT `output_audio_buffer.clear`

```
1 {  
2   "event_id": "optional_client_event_id",  
3   "type": "output_audio_buffer.clear"  
4 }
```

event_id string

The unique ID of the client event used for error handling.

type string

The event type, must be `output_audio_buffer.clear`.

Server events

These are events emitted from the OpenAI Realtime WebSocket server to the client.

error

Returned when an error occurs, which could be a client problem or a server problem. Most errors are recoverable and the session will stay open, we recommend to implementors to monitor and log error messages by default.

error object

Details of the error.

▼ Show properties

event_id string

The unique ID of the server event.

type string

The event type, must be `error`.

OBJECT `error`

```

1  {
2      "event_id": "event_890",
3      "type": "error",
4      "error": {
5          "type": "invalid_request_error",
6          "code": "invalid_event",
7          "message": "The 'type' field is mis
8          "param": null,
9          "event_id": "event_567"
10     }
11 }
```

session.created

Returned when a Session is created. Emitted automatically when a new connection is established as the first server event. This event will contain the default Session configuration.

event_id string

The unique ID of the server event.

OBJECT `session.created`

```

1  {
2      "event_id": "event_1234",
3      "type": "session.created",
4      "session": {
5          "id": "sess_001",
6          "object": "realtime.session",
```

session object

Realtime session object configuration.

✓ Show properties

type string

The event type, must be `session.created`.

```

7   "model": "gpt-4o-realtime-preview",
8   "modalities": ["text", "audio"],
9   "instructions": "...model instructi
10  "voice": "sage",
11  "input_audio_format": "pcm16",
12  "output_audio_format": "pcm16",
13  "input_audio_transcription": null,
14  "turn_detection": {
15    "type": "server_vad",
16    "threshold": 0.5,
17    "prefix_padding_ms": 300,
18    "silence_duration_ms": 200
19  },
20  "tools": [],
21  "tool_choice": "auto",
22  "temperature": 0.8,
23  "max_response_output_tokens": "inf"
24  "speed": 1.1,
25  "tracing": "auto"
26}
27 }
```

session.updated

Returned when a session is updated with a `session.update` event, unless there is an error.

event_id string

The unique ID of the server event.

OBJECT `session.updated`

```

1  {
2    "event_id": "event_5678",
3    "type": "session.updated",
4    "session": {
5      "id": "sess_001",
```

session object

Realtime session object configuration.

▼ Show properties

type string

The event type, must be `session.updated`.

```

6   "object": "realtime.session",
7   "model": "gpt-4o-realtime-preview",
8   "modalities": ["text"],
9   "instructions": "New instructions",
10  "voice": "sage",
11  "input_audio_format": "pcm16",
12  "output_audio_format": "pcm16",
13  "input_audio_transcription": {
14    "model": "whisper-1"
15  },
16  "turn_detection": null,
17  "tools": [],
18  "tool_choice": "none",
19  "temperature": 0.7,
20  "max_response_output_tokens": 200,
21  "speed": 1.1,
22  "tracing": "auto"
23 }
24 }
```

conversation.created

Returned when a conversation is created. Emitted right after session creation.

conversation object

The conversation resource.

OBJECT `conversation.created`

```

1 {
2   "event_id": "event_9101",
3   "type": "conversation.created",
```

✓ Show properties

event_id string

The unique ID of the server event.

```

4   "conversation": {
5     "id": "conv_001",
6     "object": "realtime.conversation"
7   }
8 }
```

type string

The event type, must be `conversation.created`.

conversation.item.created

Returned when a conversation item is created. There are several scenarios that produce this event:

The server is generating a Response, which if successful will produce either one or two Items, which will be of type `message` (role `assistant`) or type `function_call`.

The input audio buffer has been committed, either by the client or the server (in `server_vad` mode). The server will take the content of the input audio buffer and add it to a new user message Item.

The client has sent a `conversation.item.create` event to add a new Item to the Conversation.

OBJECT `conversation.item.created`

```

1  {
2    "event_id": "event_1920",
3    "type": "conversation.item.created",
4    "previous_item_id": "msg_002",
5    "item": {
6      "id": "msg_003",
7      "object": "realtime.item",
8      "type": "message",
9      "status": "completed",
10     "role": "user",
11     "content": []
12   }
13 }
```

event_id

string

The unique ID of the server event.

item object

The item to add to the conversation.

✓ Show properties

previous_item_id string or null

The ID of the preceding item in the Conversation context, allows the client to understand the order of the conversation. Can be `null` if the item has no predecessor.

type string

The event type, must be `conversation.item.created`.

conversation.item.retrieved

Returned when a conversation item is retrieved with `conversation.item.retrieve`.

event_id string

The unique ID of the server event.

item object

The item to add to the conversation.

OBJECT `conversation.item.retrieved`

```
1  {
2    "event_id": "event_1920",
3    "type": "conversation.item.created",
4    "previous_item_id": "msg_002",
5    "item": {
6      "id": "msg_003",
7      "object": "realtime.item",
8      "type": "message",
```

✓ Show properties

type string

The event type, must be `conversation.item.retrieved`.

```

 9      "status": "completed",
10      "role": "user",
11      "content": [
12          {
13              "type": "input_audio",
14              "transcript": "hello how ar",
15              "audio": "base64encodedaudi"
16          }
17      ]
18  }
19 }
```

conversation.item.input_audio_transcription.completed

This event is the output of audio transcription for user audio written to the user audio buffer. Transcription begins when the input audio buffer is committed by the client or server (in `server_vad` mode). Transcription runs asynchronously with Response creation, so this event may come before or after the Response events.

Realtime API models accept audio natively, and thus input transcription is a separate process run on a separate ASR (Automatic Speech Recognition) model. The transcript may diverge somewhat from the model's interpretation, and should be treated as a rough guide.

content_index integer

OBJECT `conversation.item.input_audio_transcription.completed`

```

1  {
2      "event_id": "event_2122",
3      "type": "conversation.item.input_audio_transcription.completed",
4      "item_id": "msg_003",
5      "content_index": 0,
6      "transcript": "Hello, how are you?",
7      "usage": {
8          "type": "tokens",
9          "total_tokens": 48,
10         "input_tokens": 38,
11         "input_token_details": {
12             "text_tokens": 10,
```

The index of the content part containing the audio.

event_id string

The unique ID of the server event.

```
13     "audio_tokens": 28,  
14   },  
15   "output_tokens": 10,  
16 }  
17 }
```

item_id string

The ID of the user message item containing the audio.

logprobs array or null

The log probabilities of the transcription.

✓ Show properties

transcript string

The transcribed text.

type string

The event type, must be

```
conversation.item.input_audio_transcription.completed .
```

usage object

Usage statistics for the transcription.

✓ Show possible types

conversation.item.input_audio_transcription.delta

Returned when the text value of an input audio transcription content part is updated.

content_index integer

The index of the content part in the item's content array.

delta string

The text delta.

event_id string

The unique ID of the server event.

item_id string

The ID of the item.

logprobs array or null

The log probabilities of the transcription.

▼ Show properties

type string

The event type, must be

`conversation.item.input_audio_transcription.delta`.

OBJECT `conversation.item.input_audio_transcr...` 

```
1 {
2   "type": "conversation.item.input_audio_tra
3   "event_id": "event_001",
4   "item_id": "item_001",
5   "content_index": 0,
6   "delta": "Hello"
7 }
```

conversation.item.input_audio_transcription.failed

Returned when input audio transcription is configured, and a transcription request for a user message failed. These events are separate from other `error` events so that the client can identify the related Item.

content_index integer

The index of the content part containing the audio.

error object

Details of the transcription error.

▽ Show properties

event_id string

The unique ID of the server event.

item_id string

The ID of the user message item.

type string

The event type, must be

`conversation.item.input_audio_transcription.failed`.

OBJECT `conversation.item.input_audio_transcription.failed`

```

1  {
2      "event_id": "event_2324",
3      "type": "conversation.item.input_audio_transcription.failed",
4      "item_id": "msg_003",
5      "content_index": 0,
6      "error": {
7          "type": "transcription_error",
8          "code": "audio_unintelligible",
9          "message": "The audio could not be transcribed.",
10         "param": null
11     }
12 }
```

conversation.item.truncated

Returned when an earlier assistant audio message item is truncated by the client with a `conversation.item.truncate` event. This event is used to

OBJECT `conversation.item.truncated`

synchronize the server's understanding of the audio with the client's playback.

This action will truncate the audio and remove the server-side text transcript to ensure there is no text in the context that hasn't been heard by the user.

```
1 {  
2   "event_id": "event_2526",  
3   "type": "conversation.item.truncated",  
4   "item_id": "msg_004",  
5   "content_index": 0,  
6   "audio_end_ms": 1500
```

audio_end_ms integer

The duration up to which the audio was truncated, in milliseconds.

content_index integer

The index of the content part that was truncated.

event_id string

The unique ID of the server event.

item_id string

The ID of the assistant message item that was truncated.

type string

The event type, must be `conversation.item.truncated`.

conversation.item.deleted

Returned when an item in the conversation is deleted by the client with a `conversation.item.delete` event. This event is used to synchronize the

OBJECT `conversation.item.deleted`



server's understanding of the conversation history with the client's view.

event_id string

The unique ID of the server event.

```

1 {
2   "event_id": "event_2728",
3   "type": "conversation.item.deleted",
4   "item_id": "msg_005"
5 }
```

item_id string

The ID of the item that was deleted.

type string

The event type, must be `conversation.item.deleted`.

input_audio_buffer.committed

Returned when an input audio buffer is committed, either by the client or automatically in server VAD mode. The `item_id` property is the ID of the user message item that will be created, thus a `conversation.item.created` event will also be sent to the client.

event_id string

The unique ID of the server event.

item_id string

OBJECT `input_audio_buffer.committed`

```

1 {
2   "event_id": "event_1121",
3   "type": "input_audio_buffer.committed",
4   "previous_item_id": "msg_001",
5   "item_id": "msg_002"
6 }
```

The ID of the user message item that will be created.

previous_item_id string or null

The ID of the preceding item after which the new item will be inserted. Can be `null` if the item has no predecessor.

type string

The event type, must be `input_audio_buffer.committed`.

input_audio_buffer.cleared

Returned when the input audio buffer is cleared by the client with a `input_audio_buffer.clear` event.

event_id string

The unique ID of the server event.

type string

The event type, must be `input_audio_buffer.cleared`.

OBJECT `input_audio_buffer.cleared`

```
1 {  
2   "event_id": "event_1314",  
3   "type": "input_audio_buffer.cleared"  
4 }
```

input_audio_buffer.speech_started

Sent by the server when in `server_vad` mode to indicate that speech has been detected in the audio buffer. This can happen any time audio is added to the buffer (unless speech is already detected). The client may want to use this event to interrupt audio playback or provide visual feedback to the user.

The client should expect to receive a `input_audio_buffer.speech_stopped` event when speech stops. The `item_id` property is the ID of the user message item that will be created when speech stops and will also be included in the `input_audio_buffer.speech_stopped` event (unless the client manually commits the audio buffer during VAD activation).

audio_start_ms integer

Milliseconds from the start of all audio written to the buffer during the session when speech was first detected. This will correspond to the beginning of audio sent to the model, and thus includes the `prefix_padding_ms` configured in the Session.

event_id string

The unique ID of the server event.

item_id string

The ID of the user message item that will be created when speech stops.

type string

The event type, must be `input_audio_buffer.speech_started`.

OBJECT `input_audio_buffer.speech_started`

```
1 {  
2   "event_id": "event_1516",  
3   "type": "input_audio_buffer.speech_start"  
4   "audio_start_ms": 1000,  
5   "item_id": "msg_003"  
6 }
```

input_audio_buffer.speech_stopped

Returned in `server_vad` mode when the server detects the end of speech in the audio buffer. The server will also send an `conversation.item.created` event with the user message item that is created from the audio buffer.

audio_end_ms integer

Milliseconds since the session started when speech stopped. This will correspond to the end of audio sent to the model, and thus includes the `min_silence_duration_ms` configured in the Session.

event_id string

The unique ID of the server event.

item_id string

The ID of the user message item that will be created.

type string

The event type, must be `input_audio_buffer.speech_stopped`.

OBJECT `input_audio_buffer.speech_stopped`

```
1 {
2   "event_id": "event_1718",
3   "type": "input_audio_buffer.speech_stopped",
4   "audio_end_ms": 2000,
5   "item_id": "msg_003"
6 }
```

response.created

Returned when a new Response is created. The first event of response creation, where the response is in an initial state of `in_progress`.

event_id string

The unique ID of the server event.

response object

The response resource.

▼ Show properties

type string

The event type, must be `response.created`.

OBJECT `response.created`

```

1  {
2      "event_id": "event_2930",
3      "type": "response.created",
4      "response": {
5          "id": "resp_001",
6          "object": "realtime.response",
7          "status": "in_progress",
8          "status_details": null,
9          "output": [],
10         "usage": null
11     }
12 }
```

response.done

Returned when a Response is done streaming. Always emitted, no matter the final state. The Response object included in the `response.done` event will include all output items in the Response but will omit the raw audio data.

event_id string

The unique ID of the server event.

response object

The response resource.

OBJECT `response.done`

```

1  {
2      "event_id": "event_3132",
3      "type": "response.done",
4      "response": {
5          "id": "resp_001",
6          "object": "realtime.response",
7          "status": "completed",
8          "status_details": null,
9          "output": [
10             {
11                 "text": "The quick brown fox jumps over the lazy dog."
12             }
13         ]
14     }
15 }
```

✓ Show properties

type string

The event type, must be `response.done`.

response.output_item.added

Returned when a new Item is created during Response generation.

event_id string

The unique ID of the server event.

item

```

1      "id": "msg_006",
2      "object": "realtime.item",
3      "type": "message",
4      "status": "completed",
5      "role": "assistant",
6      "content": [
7          {
8              "type": "text",
9              "text": "Sure, how"
10         }
11     ],
12     "usage": {
13         "total_tokens": 275,
14         "input_tokens": 127,
15         "output_tokens": 148,
16         "input_token_details": {
17             "cached_tokens": 384,
18             "text_tokens": 119,
19             "audio_tokens": 8,
20             "cached_tokens_details": {
21                 "text_tokens": 128,
22                 "audio_tokens": 256
23             }
24         },
25         "output_token_details": {
26             "text_tokens": 36,
27             "audio_tokens": 112
28         }
29     }
30 }
```

OBJECT `response.output_item.added`

```

1  {
2      "event_id": "event_3334",
3      "type": "response.output_item.added",
4      "response_id": "resp_001",
```

object

The item to add to the conversation.

✓ Show properties

output_index integer

The index of the output item in the Response.

response_id string

The ID of the Response to which the item belongs.

type string

The event type, must be `response.output_item.added`.

```

5     "output_index": 0,
6     "item": {
7       "id": "msg_007",
8       "object": "realtime.item",
9       "type": "message",
10      "status": "in_progress",
11      "role": "assistant",
12      "content": []
13    }
14 }
```

response.output_item.done

Returned when an Item is done streaming. Also emitted when a Response is interrupted, incomplete, or cancelled.

event_id string

The unique ID of the server event.

item object

The item to add to the conversation.

✓ Show properties

OBJECT `response.output_item.done`

```

1  {
2    "event_id": "event_3536",
3    "type": "response.output_item.done",
4    "response_id": "resp_001",
5    "output_index": 0,
6    "item": {
7      "id": "msg_007",
8      "object": "realtime.item",
9      "type": "message",
10     "status": "completed",
11     "role": "assistant",
```

output_index integer

The index of the output item in the Response.

response_id string

The ID of the Response to which the item belongs.

type string

The event type, must be `response.output_item.done`.

```

12     "content": [
13         {
14             "type": "text",
15             "text": "Sure, I can help w
16         }
17     ]
18 }
19 }
```

response.content_part.added

Returned when a new content part is added to an assistant message item during response generation.

content_index integer

The index of the content part in the item's content array.

event_id string

The unique ID of the server event.

item_id string

The ID of the item to which the content part was added.

OBJECT `response.content_part.added`

```

1 {
2     "event_id": "event_3738",
3     "type": "response.content_part.added",
4     "response_id": "resp_001",
5     "item_id": "msg_007",
6     "output_index": 0,
7     "content_index": 0,
8     "part": {
9         "type": "text",
10        "text": ""
11    }
12 }
```

output_index integer

The index of the output item in the response.

part object

The content part that was added.

▼ Show properties

response_id string

The ID of the response.

type string

The event type, must be `response.content_part.added`.

response.content_part.done

Returned when a content part is done streaming in an assistant message item. Also emitted when a Response is interrupted, incomplete, or cancelled.

content_index integer

The index of the content part in the item's content array.

event_id string

The unique ID of the server event.

OBJECT `response.content_part.done`

```
1  {
2      "event_id": "event_3940",
3      "type": "response.content_part.done",
4      "response_id": "resp_001",
5      "item_id": "msg_007",
6      "output_index": 0,
7      "content_index": 0,
8      "part": {
9          "type": "text",
10         "text": "Sure, I can help with that"
```

item_id string

The ID of the item.

```
11      }
12 }
```

output_index integer

The index of the output item in the response.

part object

The content part that is done.

✓ Show properties

response_id string

The ID of the response.

type string

The event type, must be `response.content_part.done`.

response.text.delta

Returned when the text value of a "text" content part is updated.

content_index integer

The index of the content part in the item's content array.

OBJECT `response.text.delta`

```
1 {
2   "event_id": "event_4142",
3   "type": "response.text.delta",
```

delta string

The text delta.

event_id string

The unique ID of the server event.

item_id string

The ID of the item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

type string

The event type, must be `response.text.delta`.

```

4   "response_id": "resp_001",
5   "item_id": "msg_007",
6   "output_index": 0,
7   "content_index": 0,
8   "delta": "Sure, I can h"
9 }
```

response.text.done

Returned when the text value of a "text" content part is done streaming.

Also emitted when a Response is interrupted, incomplete, or cancelled.

content_index integerOBJECT `response.text.done`

```

1 {
2   "event_id": "event_4344",
3   "type": "response.text.done",
4   "response_id": "resp_001",
```

The index of the content part in the item's content array.

event_id string

The unique ID of the server event.

```
5 "item_id": "msg_007",
6 "output_index": 0,
7 "content_index": 0,
8 "text": "Sure, I can help with that."
9 }
```

item_id string

The ID of the item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

text string

The final text content.

type string

The event type, must be `response.text.done`.

response.audio_transcript.delta

Returned when the model-generated transcription of audio output is updated.

content_index integer

The index of the content part in the item's content array.

delta string

The transcript delta.

event_id string

The unique ID of the server event.

item_id string

The ID of the item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

type string

The event type, must be `response.audio_transcript.delta`.

OBJECT `response.audio_transcript.delta`

```
1 {
2   "event_id": "event_4546",
3   "type": "response.audio_transcript.delta"
4   "response_id": "resp_001",
5   "item_id": "msg_008",
6   "output_index": 0,
7   "content_index": 0,
8   "delta": "Hello, how can I a"
9 }
```

response.audio_transcript.done

Returned when the model-generated transcription of audio output is done streaming. Also emitted when a Response is interrupted, incomplete, or cancelled.

content_index integer

The index of the content part in the item's content array.

event_id string

The unique ID of the server event.

item_id string

The ID of the item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

transcript string

The final transcript of the audio.

type string

The event type, must be `response.audio_transcript.done`.

OBJECT `response.audio_transcript.done`

```
1 {
2   "event_id": "event_4748",
3   "type": "response.audio_transcript.done"
4   "response_id": "resp_001",
5   "item_id": "msg_008",
6   "output_index": 0,
7   "content_index": 0,
8   "transcript": "Hello, how can I assist you"
9 }
```

response.audio.delta

Returned when the model-generated audio is updated.

content_index integer

The index of the content part in the item's content array.

delta string

Base64-encoded audio data delta.

event_id string

The unique ID of the server event.

item_id string

The ID of the item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

type string

The event type, must be `response.audio.delta`.

OBJECT `response.audio.delta`

```
1 {
2   "event_id": "event_4950",
3   "type": "response.audio.delta",
4   "response_id": "resp_001",
5   "item_id": "msg_008",
6   "output_index": 0,
7   "content_index": 0,
8   "delta": "Base64EncodedAudioDelta"
9 }
```

response.audio.done

Returned when the model-generated audio is done. Also emitted when a Response is interrupted, incomplete, or cancelled.

content_index integer

The index of the content part in the item's content array.

event_id string

The unique ID of the server event.

item_id string

The ID of the item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

type string

The event type, must be `response.audio.done`.

OBJECT `response.audio.done`

```
1 {
2   "event_id": "event_5152",
3   "type": "response.audio.done",
4   "response_id": "resp_001",
5   "item_id": "msg_008",
6   "output_index": 0,
7   "content_index": 0
8 }
```

response.function_call_arguments.delta

Returned when the model-generated function call arguments are updated.

call_id string

The ID of the function call.

delta string

The arguments delta as a JSON string.

event_id string

The unique ID of the server event.

item_id string

The ID of the function call item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

type string

The event type, must be `response.function_call_arguments.delta`.

OBJECT `response.function_call_arguments.delta`

```
1 {
2   "event_id": "event_5354",
3   "type": "response.function_call_argument"
4   "response_id": "resp_002",
5   "item_id": "fc_001",
6   "output_index": 0,
7   "call_id": "call_001",
8   "delta": "{\"location\": \"San\""
9 }
```

response.function_call_arguments.done

Returned when the model-generated function call arguments are done streaming. Also emitted when a Response is interrupted, incomplete, or cancelled.

arguments string

The final arguments as a JSON string.

call_id string

The ID of the function call.

event_id string

The unique ID of the server event.

item_id string

The ID of the function call item.

output_index integer

The index of the output item in the response.

response_id string

The ID of the response.

type string

The event type, must be `response.function_call_arguments.done`.

OBJECT `response.function_call_arguments.done`

```
1 {
2   "event_id": "event_5556",
3   "type": "response.function_call_argument"
4   "response_id": "resp_002",
5   "item_id": "fc_001",
6   "output_index": 0,
7   "call_id": "call_001",
8   "arguments": "{\"location\": \"San Franc
9 }
```

transcription_session.updated

Returned when a transcription session is updated with a `transcription_session.update` event, unless there is an error.

event_id string

The unique ID of the server event.

session object

A new Realtime transcription session configuration.

When a session is created on the server via REST API, the session object also contains an ephemeral key. Default TTL for keys is 10 minutes. This property is not present when a session is updated via the WebSocket API.

✓ Show properties

type string

The event type, must be `transcription_session.updated`.

OBJECT transcription_session.updated

```
1  {
2    "event_id": "event_5678",
3    "type": "transcription_session.updated",
4    "session": {
5      "id": "sess_001",
6      "object": "realtime.transcription_sessi
7      "input_audio_format": "pcm16",
8      "input_audio_transcription": {
9        "model": "gpt-4o-transcribe",
10       "prompt": "",
11       "language": ""
12     },
13     "turn_detection": {
14       "type": "server_vad",
15       "threshold": 0.5,
16       "prefix_padding_ms": 300,
17       "silence_duration_ms": 500,
18       "create_response": true,
19       // "interrupt_response": false -- th
20     },
21     "input_audio_noise_reduction": {
22       "type": "near_field"
23     },
24     "include": [
25       "item.input_audio_transcription.avg_l
26     ],
27   }
28 }
```

rate_limits.updated

Emitted at the beginning of a Response to indicate the updated rate limits. When a Response is created some tokens will be "reserved" for the output tokens, the rate limits shown here reflect that reservation, which is then adjusted accordingly once the Response is completed.

event_id string

The unique ID of the server event.

rate_limits array

List of rate limit information.

▼ Show properties

type string

The event type, must be `rate_limits.updated`.

OBJECT `rate_limits.updated`

```
1  {
2      "event_id": "event_5758",
3      "type": "rate_limits.updated",
4      "rate_limits": [
5          {
6              "name": "requests",
7              "limit": 1000,
8              "remaining": 999,
9              "reset_seconds": 60
10         },
11         {
12             "name": "tokens",
13             "limit": 50000,
14             "remaining": 49950,
15             "reset_seconds": 60
16         }
17     ]
18 }
```

output_audio_buffer.started

WebRTC Only: Emitted when the server begins streaming audio to the client. This event is emitted after an audio content part has been added (`response.content_part.added`) to the response. [Learn more.](#)

event_id string

The unique ID of the server event.

response_id string

The unique ID of the response that produced the audio.

type string

The event type, must be `output_audio_buffer.started`.

OBJECT `output_audio_buffer.started` 🔗

```
1 {  
2   "event_id": "event_abc123",  
3   "type": "output_audio_buffer.started",  
4   "response_id": "resp_abc123"  
5 }
```

output_audio_buffer.stopped

WebRTC Only: Emitted when the output audio buffer has been completely drained on the server, and no more audio is forthcoming. This event is emitted after the full response data has been sent to the client (`response.done`). [Learn more.](#)

OBJECT `output_audio_buffer.stopped` 🔗

```
1 {  
2   "event_id": "event_abc123",  
3   "type": "output_audio_buffer.stopped",
```

4

`"response_id": "resp_abc123"`**event_id** string

The unique ID of the server event.

response_id string

The unique ID of the response that produced the audio.

type string

The event type, must be `output_audio_buffer.stopped`.

output_audio_buffer.cleared

WebRTC Only: Emitted when the output audio buffer is cleared. This happens either in VAD mode when the user has interrupted (`input_audio_buffer.speech_started`), or when the client has emitted the `output_audio_buffer.clear` event to manually cut off the current audio response. [Learn more.](#)

event_id string

The unique ID of the server event.

response_id string

The unique ID of the response that produced the audio.

type stringOBJECT `output_audio_buffer.cleared`

```
1 {  
2   "event_id": "event_abc123",  
3   "type": "output_audio_buffer.cleared",  
4   "response_id": "resp_abc123"  
5 }
```

The event type must be `output_audio_buffer.cleared`.

Chat Completions

The Chat Completions API endpoint will generate a model response from a list of messages comprising a conversation.

Related guides:

[Quickstart](#)

[Text inputs and outputs](#)

[Image inputs](#)

[Audio inputs and outputs](#)

[Structured Outputs](#)

[Function calling](#)

[Conversation state](#)

Starting a new project? We recommend trying [Responses](#) to take advantage of the latest OpenAI platform features. Compare [Chat Completions with Responses](#).

Create chat completion

POST <https://api.openai.com/v1/chat/completions>

[Default](#) [Image input](#) [Streaming](#) [Functions](#) [Logs](#)

Starting a new project? We recommend trying [Responses](#) to take advantage of the latest OpenAI platform features. Compare [Chat Completions with Responses](#).

Creates a model response for the given chat conversation. Learn more in the [text generation](#), [vision](#), and [audio](#) guides.

Parameter support can differ depending on the model used to generate the response, particularly for newer reasoning models. Parameters that are only supported for reasoning models are noted below. For the current state of unsupported parameters in reasoning models, refer to the [reasoning guide](#).

Request body

messages array Required

A list of messages comprising the conversation so far. Depending on the [model](#) you use, different message types (modalities) are supported, like [text](#), [images](#), and [audio](#).

▼ Show possible types

model string Required

Model ID used to generate the response, like `gpt-4o` or `o3`. OpenAI offers a wide range of models with different capabilities, performance characteristics, and price points. Refer to the [model guide](#) to browse and compare available models.

audio object or null Optional

Parameters for audio output. Required when audio output is requested with `modalities: ["audio"]`. [Learn more](#).

Example request

gpt-5 [curl](#) [Log](#)

```
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-5",
6     "messages": [
7         {
8             "role": "developer",
9             "content": "You are a helpful assis
10        },
11        {
12            "role": "user",
13            "content": "Hello!"
14        }
15    ]
16 }'
```

Response

```
1 {
2     "id": "chatcmpl-B9MBs8Cjcv0U2jLn4n570S5qM
3     "object": "chat.completion",
4     "created": 1741569952,
5     "model": "gpt-4.1-2025-04-14",
6     "choices": [
7         {
8             "index": 0,
9             "message": {
10                 "role": "assistant",
11                 "content": "Hello! How can I assist
12             }
13         }
14     ]
15 }
```

✓ Show properties

frequency_penalty number or null Optional Defaults to 0

Number between -2.0 and 2.0. Positive values penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim.

function_call Deprecated string or object Optional

Deprecated in favor of `tool_choice`.

Controls which (if any) function is called by the model.

`none` means the model will not call a function and instead generates a message.

`auto` means the model can pick between generating a message or calling a function.

Specifying a particular function via `{"name": "my_function"}` forces the model to call that function.

`none` is the default when no functions are present. `auto` is the default if functions are present.

✓ Show possible types

functions Deprecated array Optional

Deprecated in favor of `tools`.

A list of functions the model may generate JSON inputs for.

✓ Show properties

logit_bias map Optional Defaults to null

Modify the likelihood of specified tokens appearing in the completion.

```

12     "refusal": null,
13     "annotations": []
14   },
15   "logprobs": null,
16   "finish_reason": "stop"
17 }
18 ],
19 "usage": {
20   "prompt_tokens": 19,
21   "completion_tokens": 10,
22   "total_tokens": 29,
23   "prompt_tokens_details": {
24     "cached_tokens": 0,
25     "audio_tokens": 0
26   },
27   "completion_tokens_details": {
28     "reasoning_tokens": 0,
29     "audio_tokens": 0,
30     "accepted_prediction_tokens": 0,
31     "rejected_prediction_tokens": 0
32   }
33 },
34   "service_tier": "default"
35 }
```

Accepts a JSON object that maps tokens (specified by their token ID in the tokenizer) to an associated bias value from -100 to 100. Mathematically, the bias is added to the logits generated by the model prior to sampling. The exact effect will vary per model, but values between -1 and 1 should decrease or increase likelihood of selection; values like -100 or 100 should result in a ban or exclusive selection of the relevant token.

logprobs boolean or null Optional Defaults to false

Whether to return log probabilities of the output tokens or not. If true, returns the log probabilities of each output token returned in the `content` of `message`.

max_completion_tokens integer or null Optional

An upper bound for the number of tokens that can be generated for a completion, including visible output tokens and reasoning tokens.

max_tokens Deprecated integer or null Optional

The maximum number of tokens that can be generated in the chat completion. This value can be used to control costs for text generated via API.

This value is now deprecated in favor of `max_completion_tokens`, and is not compatible with o-series models.

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

modalities array or null Optional

Output types that you would like the model to generate. Most models are capable of generating text, which is the default:

```
["text"]
```

The `gpt-4o-audio-preview` model can also be used to generate audio. To request that this model generate both text and audio responses, you can use:

```
["text", "audio"]
```

n integer or null Optional Defaults to 1

How many chat completion choices to generate for each input message. Note that you will be charged based on the number of generated tokens across all of the choices. Keep `n` as `1` to minimize costs.

parallel_tool_calls boolean Optional Defaults to true

Whether to enable parallel function calling during tool use.

prediction object Optional

Configuration for a Predicted Output, which can greatly improve response times when large parts of the model response are known ahead of time. This is most common when you are regenerating a file with only minor changes to most of the content.

✓ Show possible types

presence_penalty number or null Optional Defaults to 0

Number between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics.

prompt_cache_key string Optional

Used by OpenAI to cache responses for similar requests to optimize your cache hit rates. Replaces the `user` field. [Learn more](#).

reasoning_effort string or null Optional Defaults to medium

Constrains effort on reasoning for [reasoning models](#). Currently supported values are `minimal`, `low`, `medium`, and `high`. Reducing reasoning effort can result in faster responses and fewer tokens used on reasoning in a response.

response_format object Optional

An object specifying the format that the model must output.

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables Structured Outputs which ensures the model will match your supplied JSON schema. Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables the older JSON mode, which ensures the message the model generates is valid JSON. Using `json_schema` is preferred for models that support it.

▼ Show possible types

safety_identifier string Optional

A stable identifier used to help detect users of your application that may be violating OpenAI's usage policies. The IDs should be a string that uniquely identifies each user. We recommend hashing their username or email address, in order to avoid sending us any identifying information. [Learn more](#).

seed integer or null Optional

This feature is in Beta. If specified, our system will make a best effort to sample deterministically, such that repeated requests with the same `seed` and parameters should return the same result. Determinism is not guaranteed, and you should refer to the `system_fingerprint` response parameter to monitor changes in the backend.

service_tier string or null Optional Defaults to auto

Specifies the processing type used for serving the request.

If set to 'auto', then the request will be processed with the service tier configured in the Project settings. Unless otherwise configured, the Project will use 'default'.

If set to 'default', then the request will be processed with the standard pricing and performance for the selected model.

If set to '[flex](#)' or '[priority](#)', then the request will be processed with the corresponding service tier. [Contact sales](#) to learn more about Priority processing.

When not set, the default behavior is 'auto'.

When the `service_tier` parameter is set, the response body will include the `service_tier` value based on the processing mode actually used to serve the request. This response value may be different from the value set in the parameter.

stop string / array / null Optional Defaults to null

Not supported with latest reasoning models `o3` and `o4-mini`.

Up to 4 sequences where the API will stop generating further tokens. The returned text will not contain the stop sequence.

store boolean or null Optional Defaults to false

Whether or not to store the output of this chat completion request for use in our [model distillation](#) or [evals](#) products.

Supports text and image inputs. Note: image inputs over 10MB will be dropped.

stream boolean or null Optional Defaults to false

If set to true, the model response data will be streamed to the client as it is generated using [server-sent events](#). See the [Streaming section below](#) for more information, along with the [streaming responses](#) guide for more information on how to handle the streaming events.

stream_options object or null Optional Defaults to null

Options for streaming response. Only set this when you set `stream: true`.

✓ Show properties

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic. We generally recommend altering this or `top_p` but not both.

text object Optional

✓ Show properties

tool_choice string or object Optional

Controls which (if any) tool is called by the model. `none` means the model will not call any tool and instead generates a message. `auto` means the model can pick between generating a message or calling one or more tools. `required` means the model must call one or more tools. Specifying a particular tool via

`{"type": "function", "function": {"name": "my_function"}}` forces the model to call that tool.

`none` is the default when no tools are present. `auto` is the default if tools are present.

✓ Show possible types

tools array Optional

A list of tools the model may call. You can provide either custom tools or function tools.

✓ Show possible types

top_logprobs integer or null Optional

An integer between 0 and 20 specifying the number of most likely tokens to return at each token position, each with an associated log probability. `logprobs` must be set

to `true` if this parameter is used.

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or `temperature` but not both.

user Deprecated string Optional

This field is being replaced by `safety_identifier` and `prompt_cache_key`. Use `prompt_cache_key` instead to maintain caching optimizations. A stable identifier for your end-users. Used to boost cache hit rates by better bucketing similar requests and to help OpenAI detect and prevent abuse. [Learn more](#).

web_search_options object Optional

This tool searches the web for relevant results to use in a response. Learn more about the [web search tool](#).

✓ Show properties

Returns

Returns a [chat completion](#) object, or a streamed sequence of [chat completion chunk](#) objects if the request is streamed.

Get chat completion

GET https://api.openai.com/v1/chat/completions/{completion_id}

Get a stored chat completion. Only Chat Completions that have been created with the `store` parameter set to `true` will be returned.

Path parameters

`completion_id` string Required

The ID of the chat completion to retrieve.

Returns

The [ChatCompletion](#) object matching the specified ID.

Example request

curl ⚡ ↗

```
1 curl https://api.openai.com/v1/chat/completions/{completion_id}
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "chat.completion",
3   "id": "chatcmpl-abc123",
4   "model": "gpt-4o-2024-08-06",
5   "created": 1738960610,
6   "request_id": "req_ded8ab984ec4bf840f3756",
7   "tool_choice": null,
8   "usage": {
9     "total_tokens": 31,
10    "completion_tokens": 18,
11    "prompt_tokens": 13
12  },
13  "seed": 4944116822809979520,
14  "top_p": 1.0,
15  "temperature": 1.0,
16  "presence_penalty": 0.0,
17  "frequency_penalty": 0.0,
18  "system_fingerprint": "fp_50cad350e4",
19  "input_user": null,
20  "service_tier": "default",
21  "tools": null,
22  "metadata": {},
23  "choices": [
```

Get chat messages

```
GET https://api.openai.com/v1/chat/completions/{completion_id}/messages
```

Get the messages in a stored chat completion. Only Chat Completions that have been created with the `store` parameter set to `true` will be returned.

Path parameters

completion_id string **Required**

The ID of the chat completion to retrieve messages from.

Query parameters

after string **Optional**

Identifier for the last message from the previous pagination request.

limit integer **Optional** Defaults to 20

Number of messages to retrieve.

order string **Optional** Defaults to asc

```
24   {
25     "index": 0,
26     "message": {
27       "content": "Mind of circuits hum",
28       "role": "assistant",
29       "tool_calls": null,
30       "function_call": null
31     }
32   }
```

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/chat/completions/{completion_id}
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json"
```

curl ⌂

Response

```
1   {
2     "object": "list",
3     "data": [
4       {
5         "id": "chatmpl-AyPNinnUqUDYo9SAdA52N",
6         "role": "user",
7         "content": "write a haiku about ai",
8         "name": null,
9         "content_parts": null
10      }
11    ],
12    "first_id": "chatmpl-AyPNinnUqUDYo9SAdA52N",
13    "last_id": "chatmpl-AyPNinnUqUDYo9SAdA52N",
14    "has_more": false
15  }
```

Sort order for messages by timestamp. Use `asc` for ascending order or `desc` for descending order. Defaults to `asc`.

Returns

A list of [messages](#) for the specified chat completion.

List Chat Completions

GET <https://api.openai.com/v1/chat/completions>

List stored Chat Completions. Only Chat Completions that have been stored with the `store` parameter set to `true` will be returned.

Query parameters

after string Optional

Identifier for the last chat completion from the previous pagination request.

limit integer Optional Defaults to 20

Number of Chat Completions to retrieve.

metadata map Optional

A list of metadata keys to filter the Chat Completions by. Example:

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/chat/completions  
2 -H "Authorization: Bearer $OPENAI_API_KEY"  
3 -H "Content-Type: application/json"
```

Response

```
1 {  
2   "object": "list",  
3   "data": [  
4     {  
5       "object": "chat.completion",  
6       "id": "chatcmpl-AyPNinnUqUDYo9SAdA52N",  
7       "model": "gpt-4.1-2025-04-14",  
8       "created": 1738960610,  
9       "request_id": "req_ded8ab984ec4bf840f"
```

```
metadata[key1]=value1&metadata[key2]=value2
```

model string Optional

The model used to generate the Chat Completions.

order string Optional Defaults to asc

Sort order for Chat Completions by timestamp. Use `asc` for ascending order or `desc` for descending order. Defaults to `asc`.

Returns

A list of [Chat Completions](#) matching the specified filters.

Update chat completion

```
POST https://api.openai.com/v1/chat/completions/{completion_id}
```

Modify a stored chat completion. Only Chat Completions that have been created with the `store` parameter set to `true` can be modified.

Currently, the only supported modification is to update the `metadata` field.

Path parameters**completion_id** string Required

```

10      "tool_choice": null,
11      "usage": {
12          "total_tokens": 31,
13          "completion_tokens": 18,
14          "prompt_tokens": 13
15      },
16      "seed": 4944116822809979520,
17      "top_p": 1.0,
18      "temperature": 1.0,
19      "presence_penalty": 0.0,
20      "frequency_penalty": 0.0,
21      "system_fingerprint": "fp_50cad350e4",
22      "input_user": null,
23      "service_tier": "default",
24      "tools": null,
25      "metadata": {},
26      "choices": [
27          {
28              "index": 0,
29              "message": {
30                  "content": "Mind of circuits hu",
31                  "role": "assistant",
32                  "tool_calls": null,
33                  "function_call": null
34              }
35          }
36      ]
37  }
```

Example request

[curl](#) [Copy](#)

```

1 curl -X POST https://api.openai.com/v1/chat/
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -d '{"metadata": {"foo": "bar"}}'
5
6  ]
```

Response

[Copy](#)

The ID of the chat completion to update.

Request body

metadata map Required

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

Returns

The [ChatCompletion](#) object matching the specified ID.

Delete chat completion

```
DELETE https://api.openai.com/v1/chat/completions/{completion_id}
```

Delete a stored chat completion. Only Chat Completions that have been created with the `store` parameter set to `true` can be deleted.

Path parameters

```

1  {
2    "object": "chat.completion",
3    "id": "chatcmpl-AyPNinnUqUDYo9SAdA52NobMf",
4    "model": "gpt-4o-2024-08-06",
5    "created": 1738960610,
6    "request_id": "req_ded8ab984ec4bf840f3756",
7    "tool_choice": null,
8    "usage": {
9      "total_tokens": 31,
10     "completion_tokens": 18,
11     "prompt_tokens": 13
12   },
13   "seed": 4944116822809979520,
14   "top_p": 1.0,
15   "temperature": 1.0,
16   "presence_penalty": 0.0,
17   "frequency_penalty": 0.0,
18   "system_fingerprint": "fp_50cad350e4",
19   "input_user": null,
20   "service_tier": "default",
21   "tools": null,
22   "metadata": {
23     "foo": "bar"
24   },
25   "choices": [
26     {
27       "index": 0,

```

Example request

curl ⌂

```

1 curl -X DELETE https://api.openai.com/v1/cha
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json"
4
5   "finish_reason": "stop"

```

completion_id string Required

The ID of the chat completion to delete.

Returns

A deletion confirmation object.

Response

```

1 {
2   "object": "chat.completion.deleted",
3   "id": "chatcmpl-AyPNinnUqUDYo9SAdA52NobMfl",
4   "deleted": true

```

The chat completion object

Represents a chat completion response returned by model, based on the provided input.

choices array

A list of chat completion choices. Can be more than one if is greater than 1.

▼ Show properties

created integer

The Unix timestamp (in seconds) of when the chat completion was created.

id string

A unique identifier for the chat completion.

model string**OBJECT** The chat completion object

```

1 {
2   "id": "chatcmpl-B9MHDb5lfkBeAs8l4bebGdF0J",
3   "object": "chat.completion",
4   "created": 1741570283,
5   "model": "gpt-4o-2024-08-06",
6   "choices": [
7     {
8       "index": 0,
9       "message": {
10         "role": "assistant",
11         "content": "The image shows a woode",
12         "refusal": null,
13         "annotations": []
14     },
15     "logprobs": null,
16     "finish_reason": "stop"

```

The model used for the chat completion.

object string

The object type, which is always `chat.completion`.

service_tier string or null

Specifies the processing type used for serving the request.

If set to 'auto', then the request will be processed with the service tier configured in the Project settings. Unless otherwise configured, the Project will use 'default'.

If set to 'default', then the request will be processed with the standard pricing and performance for the selected model.

If set to '[flex](#)' or 'priority', then the request will be processed with the corresponding service tier. [Contact sales](#) to learn more about Priority processing.

When not set, the default behavior is 'auto'.

When the `service_tier` parameter is set, the response body will include the `service_tier` value based on the processing mode actually used to serve the request. This response value may be different from the value set in the parameter.

system_fingerprint string

This fingerprint represents the backend configuration that the model runs with.

Can be used in conjunction with the `seed` request parameter to understand when backend changes have been made that might impact determinism.

usage object

Usage statistics for the completion request.

▼ Show properties

```

17    }
18  ],
19  "usage": {
20    "prompt_tokens": 1117,
21    "completion_tokens": 46,
22    "total_tokens": 1163,
23    "prompt_tokens_details": {
24      "cached_tokens": 0,
25      "audio_tokens": 0
26    },
27    "completion_tokens_details": {
28      "reasoning_tokens": 0,
29      "audio_tokens": 0,
30      "accepted_prediction_tokens": 0,
31      "rejected_prediction_tokens": 0
32    }
33  },
34  "service_tier": "default",
35  "system_fingerprint": "fp_fc9f1d7035"
36 }
```

The chat completion list object

An object representing a list of Chat Completions.

data array

An array of chat completion objects.

▽ Show properties

first_id string

The identifier of the first chat completion in the data array.

has_more boolean

Indicates whether there are more Chat Completions available.

last_id string

The identifier of the last chat completion in the data array.

object string

The type of this object. It is always set to "list".

OBJECT The chat completion list object

```
1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "chat.completion",
6        "id": "chatcmpl-AyPNinnUqUDYo9SAdA52N",
7        "model": "gpt-4o-2024-08-06",
8        "created": 1738960610,
9        "request_id": "req_ded8ab984ec4bf840f",
10       "tool_choice": null,
11       "usage": {
12         "total_tokens": 31,
13         "completion_tokens": 18,
14         "prompt_tokens": 13
15       },
16       "seed": 4944116822809979520,
17       "top_p": 1.0,
18       "temperature": 1.0,
19       "presence_penalty": 0.0,
20       "frequency_penalty": 0.0,
21       "system_fingerprint": "fp_50cad350e4",
22       "input_user": null,
23       "service_tier": "default",
24       "tools": null,
25       "metadata": {},
26       "choices": [
```

The chat completion message list object

An object representing a list of chat completion messages.

data array

An array of chat completion message objects.

▽ Show properties

first_id string

The identifier of the first chat message in the data array.

has_more boolean

Indicates whether there are more chat messages available.

last_id string

The identifier of the last chat message in the data array.

object string

The type of this object. It is always set to "list".

```

27     {
28         "index": 0,
29         "message": {
30             "content": "Mind of circuits hu",
31             "role": "assistant",
32             "tool_calls": null,
33             "function_call": null
34         },
35         "finish_reason": "stop",
36     }
  
```

OBJECT The chat completion message list obj...

```

1  {
2      "object": "list",
3      "data": [
4          {
5              "id": "chatmpl-AyPNinnUqUDYo9SAdA52N",
6              "role": "user",
7              "content": "write a haiku about ai",
8              "name": null,
9              "content_parts": null
10         }
11     ],
12     "first_id": "chatmpl-AyPNinnUqUDYo9SAdA5",
13     "last_id": "chatmpl-AyPNinnUqUDYo9SAdA52",
14     "has_more": false
15 }
  
```

Streaming

Stream Chat Completions in real time. Receive chunks of completions returned from the model using server-sent events. [Learn more.](#)

The chat completion chunk object

Represents a streamed chunk of a chat completion response returned by the model, based on the provided input. [Learn more.](#)

choices array

A list of chat completion choices. Can contain more than one elements if `n` is greater than 1. Can also be empty for the last chunk if you set

```
stream_options: {"include_usage": true}
```

▼ Show properties

created integer

The Unix timestamp (in seconds) of when the chat completion was created. Each chunk has the same timestamp.

id string

A unique identifier for the chat completion. Each chunk has the same ID.

model string

The model to generate the completion.

OBJECT The chat completion chunk object

```
1 {"id":"chatmpl-123","object":"chat.completi
2
3 {"id":"chatmpl-123","object":"chat.completi
4
5 ....
6
7 {"id":"chatmpl-123","object":"chat.completi
```

object string

The object type, which is always `chat.completion.chunk`.

service_tier string or null

Specifies the processing type used for serving the request.

If set to 'auto', then the request will be processed with the service tier configured in the Project settings. Unless otherwise configured, the Project will use 'default'.

If set to 'default', then the request will be processed with the standard pricing and performance for the selected model.

If set to '[flex](#)' or 'priority', then the request will be processed with the corresponding service tier. [Contact sales](#) to learn more about Priority processing.

When not set, the default behavior is 'auto'.

When the `service_tier` parameter is set, the response body will include the `service_tier` value based on the processing mode actually used to serve the request. This response value may be different from the value set in the parameter.

system_fingerprint string

This fingerprint represents the backend configuration that the model runs with. Can be used in conjunction with the `seed` request parameter to understand when backend changes have been made that might impact determinism.

usage object or null

Usage statistics for the completion request.

▼ Show properties

Assistants Beta

Build assistants that can call models and use tools to perform tasks.

[Get started with the Assistants API](#)

Create assistant Beta

POST <https://api.openai.com/v1/assistants>

Create an assistant with a model and instructions.

Request body

model string Required

ID of the model to use. You can use the [List models](#) API to see all of your available models, or see our [Model overview](#) for descriptions of them.

description string or null Optional

The description of the assistant. The maximum length is 512 characters.

instructions string or null Optional

The system instructions that the assistant uses. The maximum length is 256,000 characters.

metadata map Optional

[Code Interpreter](#) [Files](#)

Example request

curl [?](#)

```
1 curl "https://api.openai.com/v1/assistants"  
2   -H "Content-Type: application/json" \  
3   -H "Authorization: Bearer $OPENAI_API_KEY" \  
4   -H "OpenAI-Beta: assistants=v2" \  
5   -d '{  
6     "instructions": "You are a personal mat",  
7     "name": "Math Tutor",  
8     "tools": [{"type": "code_interpreter"}]  
9     "model": "gpt-4o"  
10   }'
```

Response

?

```
1 {  
2   "id": "asst_abc123",  
3   "object": "assistant",  
4   "created_at": 1698984975,
```

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

name string or null Optional

The name of the assistant. The maximum length is 256 characters.

reasoning_effort string or null Optional Defaults to medium

Constrains effort on reasoning for [reasoning models](#). Currently supported values are `minimal`, `low`, `medium`, and `high`. Reducing reasoning effort can result in faster responses and fewer tokens used on reasoning in a response.

response_format "auto" or object Optional

Specifies the format that the model must output. Compatible with [GPT-4o](#), [GPT-4 Turbo](#), and all GPT-3.5 Turbo models since [gpt-3.5-turbo-1106](#).

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables Structured Outputs which ensures the model will match your supplied JSON schema. Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables JSON mode, which ensures the message the model generates is valid JSON.

Important: when using JSON mode, you **must** also instruct the model to produce JSON yourself via a system or user message. Without this, the model may generate an unending stream of whitespace until the generation reaches the token limit, resulting in a long-running and seemingly "stuck" request. Also note that the message content may be partially cut off if `finish_reason="length"`, which indicates the generation exceeded `max_tokens` or the conversation exceeded the max context length.

```

5   "name": "Math Tutor",
6   "description": null,
7   "model": "gpt-4o",
8   "instructions": "You are a personal math
9   "tools": [
10    {
11      "type": "code_interpreter"
12    }
13  ],
14  "metadata": {},
15  "top_p": 1.0,
16  "temperature": 1.0,
17  "response_format": "auto"
18 }
```

✓ Show possible types

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic.

tool_resources object or null Optional

A set of resources that are used by the assistant's tools. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

✓ Show properties

tools array Optional Defaults to []

A list of tool enabled on the assistant. There can be a maximum of 128 tools per assistant. Tools can be of types `code_interpreter`, `file_search`, or `function`.

✓ Show possible types

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or temperature but not both.

Returns

An [assistant](#) object.

List assistants Beta

GET <https://api.openai.com/v1/assistants>

Returns a list of assistants.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

limit integer Optional Defaults to 20

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/assistants?o
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
4   -H "OpenAI-Beta: assistants=v2"
```

Response

⌂

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "asst_abc123",
6       "object": "assistant",
7       "created_at": 1698982736,
8       "name": "Coding Tutor",
9       "description": null,
10      "model": "gpt-4o",
11      "instructions": "You are a helpful as
12      "tools": [],
13      "tool_resources": {},
14      "metadata": {},
15      "top_p": 1.0,
```

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

A list of `assistant` objects.

Retrieve assistant Beta

GET `https://api.openai.com/v1/assistants/{assistant_id}`

Retrieves an assistant.

Path parameters

assistant_id string Required

The ID of the assistant to retrieve.

Returns

```

16     "temperature": 1.0,
17     "response_format": "auto"
18   },
19   {
20     "id": "asst_abc456",
21     "object": "assistant",
22     "created_at": 1698982718,
23     "name": "My Assistant",
24     "description": null,
25     "model": "gpt-4o",
26     "instructions": "You are a helpful as
27     "tools": [],
28     "tool_resources": {},
29     "metadata": {},
30     "top_p": 1.0,
31     "temperature": 1.0,
32     "response_format": "auto"
33   },
34   {
35     "id": "asst_abc789",
36     "object": "assistant",
37     "created_at": 1698982742

```

Example request

curl ◁ ↗

```

1 curl https://api.openai.com/v1/assistants/as
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
4   -H "OpenAI-Beta: assistants=v2"

```

Response

```

1   {
2     "id": "asst_abc123",
3     "object": "assistant",
4     "created_at": 1699009709,

```

The [assistant](#) object matching the specified ID.

```

5  "name": "HR Helper",
6  "description": null,
7  "model": "gpt-4o",
8  "instructions": "You are an HR bot, and y
9  "tools": [
10   {
11     "type": "file_search"
12   }
13 ],
14 "metadata": {},
15 "top_p": 1.0,
16 "temperature": 1.0,
17 "response_format": "auto"
18 }
```

Modify assistant Beta

`POST https://api.openai.com/v1/assistants/{assistant_id}`

Modifies an assistant.

Path parameters

assistant_id string Required

The ID of the assistant to modify.

Request body

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/assistants/as
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY" \
4 -H "OpenAI-Beta: assistants=v2" \
5 -d '{
6   "instructions": "You are an HR bot, an
7   "tools": [{"type": "file_search"}],
8   "model": "gpt-4o"
9 }'
```

Response

description string or null Optional

The description of the assistant. The maximum length is 512 characters.

instructions string or null Optional

The system instructions that the assistant uses. The maximum length is 256,000 characters.

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string Optional

ID of the model to use. You can use the [List models](#) API to see all of your available models, or see our [Model overview](#) for descriptions of them.

name string or null Optional

The name of the assistant. The maximum length is 256 characters.

reasoning_effort string or null Optional Defaults to medium

Constrains effort on reasoning for [reasoning models](#). Currently supported values are `minimal`, `low`, `medium`, and `high`. Reducing reasoning effort can result in faster responses and fewer tokens used on reasoning in a response.

response_format "auto" or object Optional

Specifies the format that the model must output. Compatible with [GPT-4o](#), [GPT-4 Turbo](#), and all GPT-3.5 Turbo models since [gpt-3.5-turbo-1106](#).

```

1  {
2    "id": "asst_123",
3    "object": "assistant",
4    "created_at": 1699009709,
5    "name": "HR Helper",
6    "description": null,
7    "model": "gpt-4o",
8    "instructions": "You are an HR bot, and y
9    "tools": [
10   {
11     "type": "file_search"
12   }
13 ],
14 "tool_resources": {
15   "file_search": {
16     "vector_store_ids": []
17   }
18 },
19 "metadata": {},
20 "top_p": 1.0,
21 "temperature": 1.0,
22 "response_format": "auto"
23 }
```

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables

Structured Outputs which ensures the model will match your supplied JSON schema.

Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables JSON mode, which ensures the message the model generates is valid JSON.

Important: when using JSON mode, you **must** also instruct the model to produce JSON yourself via a system or user message. Without this, the model may generate an unending stream of whitespace until the generation reaches the token limit, resulting in a long-running and seemingly "stuck" request. Also note that the message content may be partially cut off if `finish_reason="length"`, which indicates the generation exceeded `max_tokens` or the conversation exceeded the max context length.

▼ Show possible types

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic.

tool_resources object or null Optional

A set of resources that are used by the assistant's tools. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

▼ Show properties

tools array Optional Defaults to []

A list of tool enabled on the assistant. There can be a maximum of 128 tools per assistant. Tools can be of types `code_interpreter`, `file_search`, or `function`.

✓ Show possible types

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or temperature but not both.

Returns

The modified [assistant](#) object.

Delete assistant Beta

```
DELETE https://api.openai.com/v1/assistants/{assistant_id}
```

Delete an assistant.

Path parameters

assistant_id string Required

The ID of the assistant to delete.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/assistants/as
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
4   -H "OpenAI-Beta: assistants=v2" \
5   -X DELETE
```

Response

Returns

Deletion status

```
1 {
2   "id": "asst_abc123",
3   "object": "assistant.deleted",
```

The assistant object Beta

Represents an `assistant` that can call the model and use tools.

created_at integer

The Unix timestamp (in seconds) for when the assistant was created.

description string or null

The description of the assistant. The maximum length is 512 characters.

id string

The identifier, which can be referenced in API endpoints.

instructions string or null

The system instructions that the assistant uses. The maximum length is 256,000 characters.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

OBJECT The assistant object

```
1 {
2   "id": "asst_abc123",
3   "object": "assistant",
4   "created_at": 1698984975,
5   "name": "Math Tutor",
6   "description": null,
7   "model": "gpt-4o",
8   "instructions": "You are a personal math
9   "tools": [
10    {
11      "type": "code_interpreter"
12    }
13  ],
14   "metadata": {},
15   "top_p": 1.0,
16   "temperature": 1.0,
17   "response_format": "auto"
18 }
```

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string

ID of the model to use. You can use the [List models](#) API to see all of your available models, or see our [Model overview](#) for descriptions of them.

name string or null

The name of the assistant. The maximum length is 256 characters.

object string

The object type, which is always `assistant`.

response_format "auto" or object

Specifies the format that the model must output. Compatible with [GPT-4o](#), [GPT-4 Turbo](#), and all GPT-3.5 Turbo models since `gpt-3.5-turbo-1106`.

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables

Structured Outputs which ensures the model will match your supplied JSON schema.

Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables JSON mode, which ensures the message the model generates is valid JSON.

Important: when using JSON mode, you **must** also instruct the model to produce JSON yourself via a system or user message. Without this, the model may generate an unending stream of whitespace until the generation reaches the token limit, resulting in a long-running and seemingly "stuck" request. Also note that the message content may be partially cut off if `finish_reason="length"`, which indicates the generation exceeded `max_tokens` or the conversation exceeded the max context length.

✓ Show possible types

temperature number or null

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic.

tool_resources object or null

A set of resources that are used by the assistant's tools. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

✓ Show properties

tools array

A list of tool enabled on the assistant. There can be a maximum of 128 tools per assistant. Tools can be of types `code_interpreter`, `file_search`, or `function`.

✓ Show possible types

top_p number or null

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or temperature but not both.

Threads Beta

Create threads that assistants can interact with.

Related guide: [Assistants](#)

Create thread Beta

POST <https://api.openai.com/v1/threads>

Create a thread.

Request body

messages array Optional

A list of [messages](#) to start the thread with.

>Show properties

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

tool_resources object or null Optional

Empty [Messages](#)

Example request

curl [copy](#)

```
1 curl https://api.openai.com/v1/threads \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -d ''
```

Response [copy](#)

```
1 {
2   "id": "thread_abc123",
3   "object": "thread",
4   "created_at": 1699012949,
5   "metadata": {},
6   "tool_resources": {}
7 }
```

A set of resources that are made available to the assistant's tools in this thread. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

✓ Show properties

Returns

A `thread` object.

Retrieve thread Beta

GET `https://api.openai.com/v1/threads/{thread_id}`

Retrieves a thread.

Path parameters

thread_id string Required

The ID of the thread to retrieve.

Returns

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/threa
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
4   -H "OpenAI-Beta: assistants=v2"
```

Response

```
1 {
2   "id": "thread_abc123",
3   "object": "thread",
4   "created_at": 1699014083,
5   "metadata": {},
```

The `thread` object matching the specified ID.

```

6   "tool_resources": {
7     "code_interpreter": {
8       "file_ids": []
9     }
10  }
11 }
```

Modify thread Beta

POST https://api.openai.com/v1/threads/{thread_id}

Modifies a thread.

Path parameters

thread_id string Required

The ID of the thread to modify. Only the `metadata` can be modified.

Request body

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/threads/thre
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -H "OpenAI-Beta: assistants=v2" \
5 -d '{
6   "metadata": {
7     "modified": "true",
8     "user": "abc123"
9   }
10 }'
```

Response

⌂

```

1 {
2   "id": "thread_abc123",
3   "object": "thread",
4   "created_at": 1699014083,
5   "metadata": {
6     "modified": "true",
```

tool_resources object or null Optional

A set of resources that are made available to the assistant's tools in this thread. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

▼ Show properties

```

7     "user": "abc123"
8   },
9   "tool_resources": {}
10 }
```

Returns

The modified `thread` object matching the specified ID.

Delete thread Beta

`DELETE https://api.openai.com/v1/threads/{thread_id}`

Delete a thread.

Path parameters**thread_id** string Required

The ID of the thread to delete.

Returns**Example request**

curl ⌂

```

1 curl https://api.openai.com/v1/threads/threa
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -H "OpenAI-Beta: assistants=v2" \
5 -X DELETE
```

Response

```

1 {
2   "id": "thread_abc123",
```

Deletion status

```
3   "object": "thread.deleted",
4   "deleted": true
5 }
```

The thread object Beta

Represents a thread that contains [messages](#).

created_at integer

The Unix timestamp (in seconds) for when the thread was created.

id string

The identifier, which can be referenced in API endpoints.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

object string

The object type, which is always `thread`.

tool_resources object or null

OBJECT The thread object

```
1 {
2   "id": "thread_abc123",
3   "object": "thread",
4   "created_at": 1698107661,
5   "metadata": {}
6 }
```

A set of resources that are made available to the assistant's tools in this thread. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

✓ Show properties

Messages Beta

Create messages within threads

Related guide: [Assistants](#)

Create message Beta

POST https://api.openai.com/v1/threads/{thread_id}/messages

Create a message.

Path parameters

thread_id string Required

The ID of the [thread](#) to create a message for.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/thread_id/messages
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -H "OpenAI-Beta: assistants=v2" \
5 -d '{
6   "role": "user",
7   "content": "How does AI work? Explain"
8 }'
```

Request body

content string or array Required

>Show possible types

role string Required

The role of the entity that is creating the message. Allowed values include:

`user` : Indicates the message is sent by an actual user and should be used in most cases to represent user-generated messages.

`assistant` : Indicates the message is generated by the assistant. Use this value to insert messages from the assistant into the conversation.

attachments array or null Optional

A list of files attached to the message, and the tools they should be added to.

Show properties

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

Response

```
1  {
2    "id": "msg_abc123",
3    "object": "thread.message",
4    "created_at": 1713226573,
5    "assistant_id": null,
6    "thread_id": "thread_abc123",
7    "run_id": null,
8    "role": "user",
9    "content": [
10      {
11        "type": "text",
12        "text": {
13          "value": "How does AI work? Explain",
14          "annotations": []
15        }
16      }
17    ],
18    "attachments": [],
19    "metadata": {}
20 }
```

Returns

A [message](#) object.

List messages Beta

GET https://api.openai.com/v1/threads/{thread_id}/messages

Returns a list of messages for a given thread.

Path parameters

thread_id string Required

The ID of the thread the messages belong to.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/thread_abc123/messages
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -H "OpenAI-Beta: assistants=v2"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "msg_abc123",
6       "object": "thread.message",
7       "created_at": 1699016383,
8       "assistant_id": null,
9       "thread_id": "thread_abc123",
10      "run_id": null,
11      "role": "user",
12      "content": [
13        {
14          "type": "text",
15          "text": {
16            "value": "How does AI work? Exp",
17            "annotations": []
18          }
19        }
20      ]
21    }
22  ]
23}
```

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

run_id string Optional

Filter messages by the run ID that generated them.

Returns

A list of `message` objects.

Retrieve message Beta

```
GET https://api.openai.com/v1/threads/{thread_id}/messages/{message_id}
```

Retrieve a message.

Path parameters

```

19         }
20     ],
21     "attachments": [],
22     "metadata": {}
23   },
24   {
25     "id": "msg_abc456",
26     "object": "thread.message",
27     "created_at": 1699016383,
28     "assistant_id": null,
29     "thread_id": "thread_abc123",
30     "run_id": null,
31     "role": "user",
32     "content": [
33       {
34         "type": "text",
35         "text": {
36           "value": "Hello, what is AI?",
37           "annotations": []
38         }
39       }
40     ],
41     "attachments": [],
42     "metadata": {}
43   }
44 ],
45 "first_id": "msg_abc123",

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/threads/thread_id
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -H "OpenAI-Beta: assistants=v2"

```

message_id string Required

The ID of the message to retrieve.

thread_id string Required

The ID of the thread to which this message belongs.

Returns

The message object matching the specified ID.

Response

```

1  {
2    "id": "msg_abc123",
3    "object": "thread.message",
4    "created_at": 1699017614,
5    "assistant_id": null,
6    "thread_id": "thread_abc123",
7    "run_id": null,
8    "role": "user",
9    "content": [
10      {
11        "type": "text",
12        "text": {
13          "value": "How does AI work? Explain",
14          "annotations": []
15        }
16      }
17    ],
18    "attachments": [],
19    "metadata": {}
20  ]

```

Modify message Beta

```
POST https://api.openai.com/v1/threads/{thread_id}/messages/{message_id}
```

Modifies a message.

Example request
curl ↳ Copy

```

1 curl https://api.openai.com/v1/threads/thread_id/messages/message_id \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY" \
4   -H "OpenAI-Beta: assistants=v2" \

```

Path parameters

message_id string Required

The ID of the message to modify.

thread_id string Required

The ID of the thread to which this message belongs.

Request body

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

Returns

The modified [message](#) object.

Delete message

Beta

```

5   -d '{
6     "metadata": {
7       "modified": "true",
8       "user": "abc123"
9     }
10   }'

```

Response

```

1  {
2    "id": "msg_abc123",
3    "object": "thread.message",
4    "created_at": 1699017614,
5    "assistant_id": null,
6    "thread_id": "thread_abc123",
7    "run_id": null,
8    "role": "user",
9    "content": [
10      {
11        "type": "text",
12        "text": {
13          "value": "How does AI work? Explain",
14          "annotations": []
15        }
16      }
17    ],
18    "file_ids": [],
19    "metadata": {
20      "modified": "true",
21      "user": "abc123"
22    }
23  }

```

```
DELETE https://api.openai.com/v1/threads/{thread_id}/messages/{message_id}
```

Deletes a message.

Path parameters

message_id string Required

The ID of the message to delete.

thread_id string Required

The ID of the thread to which this message belongs.

Returns

Deletion status

Example request

curl ⌂

```
1 curl -X DELETE https://api.openai.com/v1/threads/{thread_id}/messages/{message_id} \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY" \
4   -H "OpenAI-Beta: assistants=v2"
```

Response

⌚

```
1 {
2   "id": "msg_abc123",
3   "object": "thread.message.deleted",
4   "deleted": true
5 }
```

The message object Beta

Represents a message within a [thread](#).

assistant_id string or null

If applicable, the ID of the [assistant](#) that authored this message.

OBJECT The message object

⌚

```
1 {
2   "id": "msg_abc123",
3   "object": "thread.message",
4   "created_at": 1698983503,
```

attachments array or null

A list of files attached to the message, and the tools they were added to.

✓ Show properties

completed_at integer or null

The Unix timestamp (in seconds) for when the message was completed.

content array

The content of the message in array of text and/or images.

✓ Show possible types

created_at integer

The Unix timestamp (in seconds) for when the message was created.

id string

The identifier, which can be referenced in API endpoints.

incomplete_at integer or null

The Unix timestamp (in seconds) for when the message was marked as incomplete.

incomplete_details object or null

On an incomplete message, details about why the message is incomplete.

✓ Show properties

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

```
5 "thread_id": "thread_abc123",
6 "role": "assistant",
7 "content": [
8   {
9     "type": "text",
10    "text": {
11      "value": "Hi! How can I help you to
12      "annotations": []
13    }
14  }
15 ],
16 "assistant_id": "asst_abc123",
17 "run_id": "run_abc123",
18 "attachments": [],
19 "metadata": {}
20 }
```

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

object string

The object type, which is always `thread.message`.

role string

The entity that produced the message. One of `user` or `assistant`.

run_id string or null

The ID of the `run` associated with the creation of this message. Value is `null` when messages are created manually using the create message or create thread endpoints.

status string

The status of the message, which can be either `in_progress`, `incomplete`, or `completed`.

thread_id string

The `thread` ID that this message belongs to.

Runs Beta

Represents an execution run on a thread.

Related guide: [Assistants](#)

Create run Beta

POST https://api.openai.com/v1/threads/{thread_id}/runs

Default

Streaming

Streaming with Functions

Create a run.

Path parameters

thread_id string Required

The ID of the thread to run.

Query parameters

include[] array Optional

A list of additional fields to include in the response. Currently the only supported value is `step_details.tool_calls[*].file_search.results[*].content` to fetch the file search result content.

See the [file search tool documentation](#) for more information.

Request body

assistant_id string Required

Example request

curl ▾ [Copy](#)

```
1 curl https://api.openai.com/v1/threads/thread_abc123/runs
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -d '{
6     "assistant_id": "asst_abc123"
7   }'
```

Response

[Copy](#)

```
1 {
2   "id": "run_abc123",
3   "object": "thread.run",
4   "created_at": 1699063290,
5   "assistant_id": "asst_abc123",
6   "thread_id": "thread_abc123",
7   "status": "queued",
8   "started_at": 1699063290,
9   "expires_at": null,
10  "cancelled_at": null,
11  "failed_at": null,
12  "completed_at": 1699063291,
```

The ID of the [assistant](#) to use to execute this run.

additional_instructions string or null Optional

Appends additional instructions at the end of the instructions for the run. This is useful for modifying the behavior on a per-run basis without overriding other instructions.

additional_messages array or null Optional

Adds additional messages to the thread before creating the run.

✓ Show properties

instructions string or null Optional

Overrides the [instructions](#) of the assistant. This is useful for modifying the behavior on a per-run basis.

max_completion_tokens integer or null Optional

The maximum number of completion tokens that may be used over the course of the run. The run will make a best effort to use only the number of completion tokens specified, across multiple turns of the run. If the run exceeds the number of completion tokens specified, the run will end with status [incomplete](#). See [incomplete_details](#) for more info.

max_prompt_tokens integer or null Optional

The maximum number of prompt tokens that may be used over the course of the run. The run will make a best effort to use only the number of prompt tokens specified, across multiple turns of the run. If the run exceeds the number of prompt tokens specified, the run will end with status [incomplete](#). See [incomplete_details](#) for more info.

metadata map Optional

```

13 "last_error": null,
14 "model": "gpt-4o",
15 "instructions": null,
16 "incomplete_details": null,
17 "tools": [
18   {
19     "type": "code_interpreter"
20   }
21 ],
22 "metadata": {},
23 "usage": null,
24 "temperature": 1.0,
25 "top_p": 1.0,
26 "max_prompt_tokens": 1000,
27 "max_completion_tokens": 1000,
28 "truncation_strategy": {
29   "type": "auto",
30   "last_messages": null
31 },
32 "response_format": "auto",
33 "tool_choice": "auto",
34 "parallel_tool_calls": true
35 }
```

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string Optional

The ID of the [Model](#) to be used to execute this run. If a value is provided here, it will override the model associated with the assistant. If not, the model associated with the assistant will be used.

parallel_tool_calls boolean Optional Defaults to true

Whether to enable [parallel function calling](#) during tool use.

reasoning_effort string or null Optional Defaults to medium

Constrains effort on reasoning for [reasoning models](#). Currently supported values are `minimal`, `low`, `medium`, and `high`. Reducing reasoning effort can result in faster responses and fewer tokens used on reasoning in a response.

response_format "auto" or object Optional

Specifies the format that the model must output. Compatible with [GPT-4o](#), [GPT-4 Turbo](#), and all GPT-3.5 Turbo models since [gpt-3.5-turbo-1106](#).

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables Structured Outputs which ensures the model will match your supplied JSON schema. Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables JSON mode, which ensures the message the model generates is valid JSON.

Important: when using JSON mode, you **must** also instruct the model to produce JSON yourself via a system or user message. Without this, the model may generate an

unending stream of whitespace until the generation reaches the token limit, resulting in a long-running and seemingly "stuck" request. Also note that the message content may be partially cut off if `finish_reason="length"`, which indicates the generation exceeded `max_tokens` or the conversation exceeded the max context length.

✓ Show possible types

stream boolean or null Optional

If `true`, returns a stream of events that happen during the Run as server-sent events, terminating when the Run enters a terminal state with a `data: [DONE]` message.

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic.

tool_choice string or object Optional

Controls which (if any) tool is called by the model. `none` means the model will not call any tools and instead generates a message. `auto` is the default value and means the model can pick between generating a message or calling one or more tools. `required` means the model must call one or more tools before responding to the user. Specifying a particular tool like `{"type": "file_search"}` or `{"type": "function", "function": {"name": "my_function"}}` forces the model to call that tool.

✓ Show possible types

tools array or null Optional

Override the tools the assistant can use for this run. This is useful for modifying the behavior on a per-run basis.

✓ Show possible types

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or temperature but not both.

truncation_strategy object or null Optional

Controls for how a thread will be truncated prior to the run. Use this to control the initial context window of the run.

✓ Show properties

Returns

A [run](#) object.

Create thread and run Beta

POST <https://api.openai.com/v1/threads/runs>

Create a thread and run it in one request.

Request body

[Default](#) [Streaming](#) [Streaming with Functions](#)

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/runs
2 -H "Authorization: Bearer $OPENAI_API_KEY"
```

assistant_id string Required

The ID of the [assistant](#) to use to execute this run.

instructions string or null Optional

Override the default system message of the assistant. This is useful for modifying the behavior on a per-run basis.

max_completion_tokens integer or null Optional

The maximum number of completion tokens that may be used over the course of the run. The run will make a best effort to use only the number of completion tokens specified, across multiple turns of the run. If the run exceeds the number of completion tokens specified, the run will end with status `incomplete`. See `incomplete_details` for more info.

max_prompt_tokens integer or null Optional

The maximum number of prompt tokens that may be used over the course of the run. The run will make a best effort to use only the number of prompt tokens specified, across multiple turns of the run. If the run exceeds the number of prompt tokens specified, the run will end with status `incomplete`. See `incomplete_details` for more info.

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string Optional

```

3   -H "Content-Type: application/json" \
4   -H "OpenAI-Beta: assistants=v2" \
5   -d '{
6     "assistant_id": "asst_abc123",
7     "thread": {
8       "messages": [
9         {"role": "user", "content": "Expl
10        ]
11      }
12    }'

```

Response

```

1  {
2    "id": "run_abc123",
3    "object": "thread.run",
4    "created_at": 1699076792,
5    "assistant_id": "asst_abc123",
6    "thread_id": "thread_abc123",
7    "status": "queued",
8    "started_at": null,
9    "expires_at": 1699077392,
10   "cancelled_at": null,
11   "failed_at": null,
12   "completed_at": null,
13   "required_action": null,
14   "last_error": null,
15   "model": "gpt-4o",
16   "instructions": "You are a helpful assist
17   "tools": [],
18   "tool_resources": {},
19   "metadata": {},
20   "temperature": 1.0,
21   "top_p": 1.0,
22   "max_completion_tokens": null,

```

The ID of the [Model](#) to be used to execute this run. If a value is provided here, it will override the model associated with the assistant. If not, the model associated with the assistant will be used.

parallel_tool_calls boolean Optional Defaults to true

Whether to enable [parallel function calling](#) during tool use.

response_format "auto" or object Optional

Specifies the format that the model must output. Compatible with [GPT-4o](#), [GPT-4 Turbo](#), and all GPT-3.5 Turbo models since [gpt-3.5-turbo-1106](#).

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables Structured Outputs which ensures the model will match your supplied JSON schema. Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables JSON mode, which ensures the message the model generates is valid JSON.

Important: when using JSON mode, you **must** also instruct the model to produce JSON yourself via a system or user message. Without this, the model may generate an unending stream of whitespace until the generation reaches the token limit, resulting in a long-running and seemingly "stuck" request. Also note that the message content may be partially cut off if `finish_reason="length"`, which indicates the generation exceeded `max_tokens` or the conversation exceeded the max context length.

▼ Show possible types

stream boolean or null Optional

If `true`, returns a stream of events that happen during the Run as server-sent events, terminating when the Run enters a terminal state with a `data: [DONE]` message.

```

23   "max_prompt_tokens": null,
24   "truncation_strategy": {
25     "type": "auto",
26     "last_messages": null
27   },
28   "incomplete_details": null,
29   "usage": null,
30   "response_format": "auto",
31   "tool_choice": "auto",
32   "parallel_tool_calls": true
33 }
```

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic.

thread object Optional

Options to create a new thread. If no thread is provided when running a request, an empty thread will be created.

▼ Show properties

tool_choice string or object Optional

Controls which (if any) tool is called by the model. `none` means the model will not call any tools and instead generates a message. `auto` is the default value and means the model can pick between generating a message or calling one or more tools.

`required` means the model must call one or more tools before responding to the user. Specifying a particular tool like `{"type": "file_search"}` or `{"type": "function", "function": {"name": "my_function"}}` forces the model to call that tool.

▼ Show possible types

tool_resources object or null Optional

A set of resources that are used by the assistant's tools. The resources are specific to the type of tool. For example, the `code_interpreter` tool requires a list of file IDs, while the `file_search` tool requires a list of vector store IDs.

▼ Show properties

tools array or null Optional

Override the tools the assistant can use for this run. This is useful for modifying the behavior on a per-run basis.

✓ Show possible types

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or temperature but not both.

truncation_strategy object or null Optional

Controls for how a thread will be truncated prior to the run. Use this to control the initial context window of the run.

✓ Show properties

Returns

A [run](#) object.

List runs Beta

GET `https://api.openai.com/v1/threads/{thread_id}/runs`

Returns a list of runs belonging to a thread.

Path parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/thread_id/runs
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3
4
```

thread_id string Required

The ID of the thread the run belongs to.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

Response

```
1  {
2    "object": "list",
3    "data": [
4      {
5        "id": "run_abc123",
6        "object": "thread.run",
7        "created_at": 1699075072,
8        "assistant_id": "asst_abc123",
9        "thread_id": "thread_abc123",
10       "status": "completed",
11       "started_at": 1699075072,
12       "expires_at": null,
13       "cancelled_at": null,
14       "failed_at": null,
15       "completed_at": 1699075073,
16       "last_error": null,
17       "model": "gpt-4o",
18       "instructions": null,
19       "incomplete_details": null,
20       "tools": [
21         {
22           "type": "code_interpreter"
23         }
24       ],
25       "tool_resources": {
26         "code_interpreter": {
27           "file_ids": [
28             "file-abc123",
29             "file-abc456"
30           ]
31         }
32       },
33     }
34   }
```

A list of [run](#) objects.

Retrieve run Beta

GET https://api.openai.com/v1/threads/{thread_id}/runs/{run_id}

Retrieves a run.

Path parameters

run_id string Required

The ID of the run to retrieve.

thread_id string Required

The ID of the [thread](#) that was run.

Returns

The [run](#) object matching the specified ID.

```

33     "metadata": {},
34     "usage": {
35       "prompt_tokens": 123,
36       "completion_tokens": 456,
37       "total_tokens": 579
38     },
39     "temperature": 1.0,
40     "top_p": 1.0,
41     "max_prompt_tokens": 1000,
42     "max_completion_tokens": 1000,

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/threads/threa
2   -H "Authorization: Bearer $OPENAI_API_KEY"
3   -H "OpenAI-Beta: assistants=v2"
49     "parallel_tool_calls": true

```

Response

```

1   {
2     "id": "run_abc123",
3     "object": "thread.run",
4     "created_at": 1699075072,
5     "assistant_id": "asst_abc123",
6     "thread_id": "thread_abc123",
7     "status": "completed",
8     "started_at": 1699075072,
9     "expires_at": null,
10    "cancelled_at": null,
11    "failed_at": null,
12    "completed_at": 1699075073,
13    "last_error": null,
14    "model": "gpt-4o",
15    "instructions": null,
16    "incomplete_details": null,

```

Modify run Beta

POST https://api.openai.com/v1/threads/{thread_id}/runs/{run_id}

Modifies a run.

Path parameters

run_id string Required

The ID of the run to modify.

thread_id string Required

The ID of the thread that was run.

Request body

metadata map Optional

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

```

17 "tools": [
18   {
19     "type": "code_interpreter"
20   }
21 ],
22 "metadata": {},
23 "usage": {
24   "prompt_tokens": 123,
25   "completion_tokens": 456,
26   "total_tokens": 579
27 },
28 "temperature": 1.0,
29 "top_p": 1.0

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/threads/thread_abc123/runs/run_abc123
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -H "OpenAI-Beta: assistants=v2" \
5 -d '{
6   "metadata": {
7     "user_id": "user_abc123"
8   }
9 }'
--> {"id": "run_abc123", "object": "thread.run", "status": "completed", "started_at": 1699075072, "assistant_id": "asst_abc123", "thread_id": "thread_abc123", "created_at": 1699075072, "total_tokens": 579, "prompt_tokens": 123, "completion_tokens": 456, "temperature": 1.0, "top_p": 1.0}

```

Response

```

1 {
2   "id": "run_abc123",
3   "object": "thread.run",
4   "status": "completed",
5   "assistant_id": "asst_abc123",
6   "thread_id": "thread_abc123",
7   "created_at": 1699075072,
8   "total_tokens": 579,
9   "prompt_tokens": 123,
10  "completion_tokens": 456,
11  "temperature": 1.0,
12  "top_p": 1.0

```

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

Returns

The modified `run` object matching the specified ID.

Submit tool outputs to run Beta

```
POST https://api.openai.com/v1/threads/{thread_id}/runs/{run_id}/submit_tool_outputs
```

When a run has the `status: "requires_action"` and `required_action.type` is `submit_tool_outputs`, this endpoint can be used to submit the outputs from the tool calls once they're all completed. All outputs must be submitted in a single request.

Path parameters

run_id string Required

The ID of the run that requires the tool output submission.

thread_id string Required

```

9   "expires_at": null,
10  "cancelled_at": null,
11  "failed_at": null,
12  "completed_at": 1699075073,
13  "last_error": null,
14  "model": "gpt-4o",
15  "instructions": null,
16  "incomplete_details": null,
17  "tools": [
18    {
19      "type": "code_interpreter"
20    }
21  ],
22  "tool_resources": {
23    "code_interpreter": {
24      "file_ids": [
25        "file-abc123",
Default  Streaming bc456"
27      ]
28    }
29  }
30 }
```

Example request

curl ⚡

```

1 curl https://api.openai.com/v1/threads/thre
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -H "OpenAI-Beta: assistants=v2" \
5 -d '{
6   "tool_outputs": [
7     {
8       "tool_call_id": "call_001",
9       "output": "70 degrees and sunny."
10      }
11    ]
12  }'
13   "type": "auto"
14 }
```

The ID of the [thread](#) to which this run belongs.

Request body

tool_outputs array Required

A list of tools for which the outputs are being submitted.

>Show properties

stream boolean or null Optional

If `true`, returns a stream of events that happen during the Run as server-sent events, terminating when the Run enters a terminal state with a `data: [DONE]` message.

Returns

The modified [run](#) object matching the specified ID.

Cancel a run Beta

```
POST https://api.openai.com/v1/threads/{thread_id}/runs/{run_id}/cancel
```

Cancels a run that is `in_progress`.

Response

```

1  {
2    "id": "run_123",
3    "object": "thread.run",
4    "created_at": 1699075592,
5    "assistant_id": "asst_123",
6    "thread_id": "thread_123",
7    "status": "queued",
8    "started_at": 1699075592,
9    "expires_at": 1699076192,
10   "cancelled_at": null,
11   "failed_at": null,
12   "completed_at": null,
13   "last_error": null,
14   "model": "gpt-4o",
15   "instructions": null,
16   "tools": [
17     {
18       "type": "function",
19       "function": {
20         "name": "get_current_weather",
21         "description": "Get the current weather in a city",
22         "parameters": {
23           "type": "object",
24           "properties": {
25             "location": {
26               "type": "string",
27               "description": "The city and state"
28             }
29           }
30         }
31       }
32     }
33   ]
34 }
```

Example request

curl ⋮ 🔗

```
1 curl https://api.openai.com/v1/threads/thread_123/runs/run_123/cancel
2 -H "Authorization: Bearer $OPENAI_API_KEY"
```

Path parameters

run_id string Required

The ID of the run to cancel.

thread_id string Required

The ID of the thread to which this run belongs.

Returns

The modified [run](#) object matching the specified ID.

The run object Beta

Represents an execution run on a [thread](#).

```
3   -H "OpenAI-Beta: assistants=v2" \
35 }
```

Response

```
1  {
2    "id": "run_abc123",
3    "object": "thread.run",
4    "created_at": 1699076126,
5    "assistant_id": "asst_abc123",
6    "thread_id": "thread_abc123",
7    "status": "cancelling",
8    "started_at": 1699076126,
9    "expires_at": 1699076726,
10   "cancelled_at": null,
11   "failed_at": null,
12   "completed_at": null,
13   "last_error": null,
14   "model": "gpt-4o",
15   "instructions": "You summarize books.",
16   "tools": [
17     {
18       "type": "file_search"
19     }
20   ],
21   "tool_resources": {
22     "file_search": {
23       "vector_store_ids": ["vs_123"]
24     }
25   },
26   "metadata": {},
27   "usage": null,
28   "temperature": 1.0,
```

OBJECT The run object

assistant_id string

The ID of the [assistant](#) used for execution of this run.

cancelled_at integer or null

The Unix timestamp (in seconds) for when the run was cancelled.

completed_at integer or null

The Unix timestamp (in seconds) for when the run was completed.

created_at integer

The Unix timestamp (in seconds) for when the run was created.

expires_at integer or null

The Unix timestamp (in seconds) for when the run will expire.

failed_at integer or null

The Unix timestamp (in seconds) for when the run failed.

id string

The identifier, which can be referenced in API endpoints.

incomplete_details object or null

Details on why the run is incomplete. Will be `null` if the run is not incomplete.

▼ Show properties

instructions string

The instructions that the [assistant](#) used for this run.

last_error

https://platform.openai.com/docs/api-reference/webhook_events/batch/expired

```
1  {
2    "id": "run_abc123",
3    "object": "thread.run",
4    "created_at": 1698107661,
5    "assistant_id": "asst_abc123",
6    "thread_id": "thread_abc123",
7    "status": "completed",
8    "started_at": 1699073476,
9    "expires_at": null,
10   "cancelled_at": null,
11   "failed_at": null,
12   "completed_at": 1699073498,
13   "last_error": null,
14   "model": "gpt-4o",
15   "instructions": null,
16   "tools": [{"type": "file_search"}, {"type": "metadata": {}, "incomplete_details": null, "usage": { "prompt_tokens": 123, "completion_tokens": 456, "total_tokens": 579 }, "temperature": 1.0, "top_p": 1.0, "max_prompt_tokens": 1000, "max_completion_tokens": 1000, "truncation_strategy": { "type": "auto", "last_messages": null }, "response_format": "auto", "tool_choice": "auto", "parallel_tool_calls": true }]
```

object or null

The last error associated with this run. Will be `null` if there are no errors.

✓ Show properties

max_completion_tokens integer or null

The maximum number of completion tokens specified to have been used over the course of the run.

max_prompt_tokens integer or null

The maximum number of prompt tokens specified to have been used over the course of the run.

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

model string

The model that the [assistant](#) used for this run.

object string

The object type, which is always `thread.run`.

parallel_tool_calls boolean

Whether to enable [parallel function calling](#) during tool use.

required_action object or null

Details on the action required to continue the run. Will be `null` if no action is required.

▼ Show properties

response_format "auto" or object

Specifies the format that the model must output. Compatible with [GPT-4o](#), [GPT-4 Turbo](#), and all GPT-3.5 Turbo models since `gpt-3.5-turbo-1106`.

Setting to `{ "type": "json_schema", "json_schema": {...} }` enables Structured Outputs which ensures the model will match your supplied JSON schema. Learn more in the [Structured Outputs guide](#).

Setting to `{ "type": "json_object" }` enables JSON mode, which ensures the message the model generates is valid JSON.

Important: when using JSON mode, you **must** also instruct the model to produce JSON yourself via a system or user message. Without this, the model may generate an unending stream of whitespace until the generation reaches the token limit, resulting in a long-running and seemingly "stuck" request. Also note that the message content may be partially cut off if `finish_reason="length"`, which indicates the generation exceeded `max_tokens` or the conversation exceeded the max context length.

▼ Show possible types

started_at integer or null

The Unix timestamp (in seconds) for when the run was started.

status string

The status of the run, which can be either `queued`, `in_progress`, `requires_action`, `canceling`, `cancelled`, `failed`, `completed`, `incomplete`, or `expired`.

temperature number or null

The sampling temperature used for this run. If not set, defaults to 1.

thread_id string

The ID of the thread that was executed on as a part of this run.

tool_choice string or object

Controls which (if any) tool is called by the model. `none` means the model will not call any tools and instead generates a message. `auto` is the default value and means the model can pick between generating a message or calling one or more tools.

`required` means the model must call one or more tools before responding to the user. Specifying a particular tool like `{"type": "file_search"}` or `{"type": "function", "function": {"name": "my_function"}}` forces the model to call that tool.

▼ Show possible types

tools array

The list of tools that the assistant used for this run.

▼ Show possible types

top_p number or null

The nucleus sampling value used for this run. If not set, defaults to 1.

truncation_strategy object or null

Controls for how a thread will be truncated prior to the run. Use this to control the initial context window of the run.

▼ Show properties

usage

object or null

Usage statistics related to the run. This value will be `null` if the run is not in a terminal state (i.e. `in_progress`, `queued`, etc.).

▼ Show properties

Run steps Beta

Represents the steps (model and tool calls) taken during the run.

Related guide: [Assistants](#)

List run steps Beta

```
GET https://api.openai.com/v1/threads/{thread_id}/runs/{run_id}/steps
```

Returns a list of run steps belonging to a run.

Path parameters

run_id string Required

The ID of the run the run steps belong to.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/thread_id/run_id/steps
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -H "OpenAI-Beta: assistants=v2"
```

Response

⌂

```
1 {
2   "object": "list",
```

thread_id string Required

The ID of the thread the run and run steps belong to.

Query parameters**after** string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

include[] array Optional

A list of additional fields to include in the response. Currently the only supported value is `step_details.tool_calls[*].file_search.results[*].content` to fetch the file search result content.

See the [file search tool documentation](#) for more information.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

```

3   "data": [
4     {
5       "id": "step_abc123",
6       "object": "thread.run.step",
7       "created_at": 1699063291,
8       "run_id": "run_abc123",
9       "assistant_id": "asst_abc123",
10      "thread_id": "thread_abc123",
11      "type": "message_creation",
12      "status": "completed",
13      "cancelled_at": null,
14      "completed_at": 1699063291,
15      "expired_at": null,
16      "failed_at": null,
17      "last_error": null,
18      "step_details": {
19        "type": "message_creation",
20        "message_creation": {
21          "message_id": "msg_abc123"
22        }
23      },
24      "usage": {
25        "prompt_tokens": 123,
26        "completion_tokens": 456,
27        "total_tokens": 579
28      }
29    },
30    "first_id": "step_abc123",
31    "last_id": "step_abc456",
32    "has_more": false
33  }
34 }
```

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

A list of `run_step` objects.

Retrieve run step Beta

```
GET https://api.openai.com/v1/threads/{thread_id}/runs/{run_id}/steps/{step_id}
```

Retrieves a run step.

Path parameters

run_id string Required

The ID of the run to which the run step belongs.

step_id string Required

The ID of the run step to retrieve.

thread_id string Required

The ID of the thread to which the run and run step belongs.

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/threads/thread_abc123/runs/run_abc123/steps/step_abc123
2 -H "Authorization: Bearer $OPENAI_API_KEY"
3 -H "Content-Type: application/json" \
4 -H "OpenAI-Beta: assistants=v2"
```

Response

```
1 {
2   "id": "step_abc123",
3   "object": "thread.run.step",
4   "created_at": 1699063291,
5   "run_id": "run_abc123",
6   "assistant_id": "asst_abc123",
7   "thread_id": "thread_abc123",
8   "type": "message_creation",
```

Query parameters

include[] array Optional

A list of additional fields to include in the response. Currently the only supported value is `step_details.tool_calls[*].file_search.results[*].content` to fetch the file search result content.

See the [file search tool documentation](#) for more information.

Returns

The [run step](#) object matching the specified ID.

```

9  "status": "completed",
10 "cancelled_at": null,
11 "completed_at": 1699063291,
12 "expired_at": null,
13 "failed_at": null,
14 "last_error": null,
15 "step_details": {
16   "type": "message_creation",
17   "message_creation": {
18     "message_id": "msg_abc123"
19   }
20 },
21 "usage": {
22   "prompt_tokens": 123,
23   "completion_tokens": 456,
24   "total_tokens": 579
25 }
26 }
```

The run step object Beta

Represents a step in execution of a run.

assistant_id string

The ID of the [assistant](#) associated with the run step.

cancelled_at integer or null

The Unix timestamp (in seconds) for when the run step was cancelled.

completed_at integer or null

OBJECT The run step object

```

1  {
2   "id": "step_abc123",
3   "object": "thread.run.step",
4   "created_at": 1699063291,
5   "run_id": "run_abc123",
6   "assistant_id": "asst_abc123",
7   "thread_id": "thread_abc123",
8   "type": "message_creation",
```

The Unix timestamp (in seconds) for when the run step completed.

created_at integer

The Unix timestamp (in seconds) for when the run step was created.

expired_at integer or null

The Unix timestamp (in seconds) for when the run step expired. A step is considered expired if the parent run is expired.

failed_at integer or null

The Unix timestamp (in seconds) for when the run step failed.

id string

The identifier of the run step, which can be referenced in API endpoints.

last_error object or null

The last error associated with this run step. Will be `null` if there are no errors.

▼ Show properties

metadata map

Set of 16 key-value pairs that can be attached to an object. This can be useful for storing additional information about the object in a structured format, and querying for objects via API or the dashboard.

Keys are strings with a maximum length of 64 characters. Values are strings with a maximum length of 512 characters.

object string

The object type, which is always `thread.run.step`.

```

9   "status": "completed",
10  "cancelled_at": null,
11  "completed_at": 1699063291,
12  "expired_at": null,
13  "failed_at": null,
14  "last_error": null,
15  "step_details": {
16    "type": "message_creation",
17    "message_creation": {
18      "message_id": "msg_abc123"
19    }
20  },
21  "usage": {
22    "prompt_tokens": 123,
23    "completion_tokens": 456,
24    "total_tokens": 579
25  }
26 }
```

run_id string

The ID of the [run](#) that this run step is a part of.

status string

The status of the run step, which can be either `in_progress`, `cancelled`,
`failed`, `completed`, or `expired`.

step_details object

The details of the run step.

▼ Show possible types

thread_id string

The ID of the [thread](#) that was run.

type string

The type of run step, which can be either `message_creation` or `tool_calls`.

usage object or null

Usage statistics related to the run step. This value will be `null` while the run step's status is `in_progress`.

▼ Show properties

Streaming Beta

Stream the result of executing a Run or resuming a Run after submitting tool outputs. You can stream events from the [Create Thread and Run](#), [Create Run](#), and [Submit Tool Outputs](#) endpoints by passing `"stream": true`. The response will be a [Server-Sent events](#) stream. Our Node and Python SDKs provide helpful utilities to make streaming easy. Reference the [Assistants API quickstart](#) to learn more.

The message delta object Beta

Represents a message delta i.e. any changed fields on a message during streaming.

delta object

The delta containing the fields that have changed on the Message.

✓ Show properties

id string

The identifier of the message, which can be referenced in API endpoints.

object string

The object type, which is always `thread.message.delta`.

OBJECT The message delta object

```
1  {
2    "id": "msg_123",
3    "object": "thread.message.delta",
4    "delta": {
5      "content": [
6        {
7          "index": 0,
8          "type": "text",
9          "text": { "value": "Hello", "annota
10        }
11      ]
12    }
13 }
```

The run step delta object Beta

Represents a run step delta i.e. any changed fields on a run step during streaming.

delta object

The delta containing the fields that have changed on the run step.

✓ Show properties

id string

The identifier of the run step, which can be referenced in API endpoints.

object string

The object type, which is always `thread.run.step.delta`.

OBJECT The run step delta object

```

1  {
2    "id": "step_123",
3    "object": "thread.run.step.delta",
4    "delta": {
5      "step_details": {
6        "type": "tool_calls",
7        "tool_calls": [
8          {
9            "index": 0,
10           "id": "call_123",
11           "type": "code_interpreter",
12           "code_interpreter": { "input": "" }
13         }
14       ]
15     }
16   }
17 }
```

Assistant stream events Beta

Represents an event emitted when streaming a Run.

Each event in a server-sent events stream has an `event` and `data` property:

```

event: thread.created
data: {"id": "thread_123", "object": "thread", ...}
```

We emit events whenever a new object is created, transitions to a new state, or is being streamed in parts (deltas). For example, we emit

`thread.run.created` when a new run is created, `thread.run.completed` when a run completes, and so on. When an Assistant chooses to create a message during a run, we emit a `thread.message.created` event, a `thread.message.in_progress` event, many `thread.message.delta` events, and finally a `thread.message.completed` event.

We may add additional events over time, so we recommend handling unknown events gracefully in your code. See the [Assistants API quickstart](#) to learn how to integrate the Assistants API with streaming.

done `data` is `[DONE]`

Occurs when a stream ends.

error `data` is an `error`

Occurs when an `error` occurs. This can happen due to an internal server error or a timeout.

thread.created `data` is a `thread`

Occurs when a new `thread` is created.

thread.message.completed `data` is a `message`

Occurs when a `message` is completed.

thread.message.created `data` is a `message`

Occurs when a `message` is created.

thread.message.delta `data` is a [message delta](#)

Occurs when parts of a [Message](#) are being streamed.

thread.message.in_progress `data` is a [message](#)

Occurs when a [message](#) moves to an [in_progress](#) state.

thread.message.incomplete `data` is a [message](#)

Occurs when a [message](#) ends before it is completed.

thread.run.cancelled `data` is a [run](#)

Occurs when a [run](#) is cancelled.

thread.run.cancelling `data` is a [run](#)

Occurs when a [run](#) moves to a [cancelling](#) status.

thread.run.completed `data` is a [run](#)

Occurs when a [run](#) is completed.

thread.run.created `data` is a [run](#)

Occurs when a new [run](#) is created.

thread.run.expired `data` is a [run](#)

Occurs when a [run](#) expires.

thread.run.failed `data` is a [run](#)

Occurs when a [run](#) fails.

thread.run.in_progress `data` is a [run](#)

Occurs when a [run](#) moves to an [in_progress](#) status.

thread.run.incomplete `data` is a [run](#)

Occurs when a [run](#) ends with status [incomplete](#).

thread.run.queued `data` is a [run](#)

Occurs when a [run](#) moves to a [queued](#) status.

thread.run.requires_action `data` is a [run](#)

Occurs when a [run](#) moves to a [requires_action](#) status.

thread.run.step.cancelled `data` is a [run step](#)

Occurs when a [run step](#) is cancelled.

thread.run.step.completed `data` is a [run step](#)

Occurs when a [run step](#) is completed.

thread.run.step.created `data` is a [run step](#)

Occurs when a [run step](#) is created.

thread.run.step.delta `data` is a [run step delta](#)

Occurs when parts of a [run step](#) are being streamed.

thread.run.step.expired `data` is a [run step](#)

Occurs when a [run step](#) expires.

thread.run.step.failed `data` is a [run step](#)

Occurs when a [run step](#) fails.

thread.run.step.in_progress `data` is a [run step](#)

Occurs when a [run step](#) moves to an `in_progress` state.

Administration

Programmatically manage your organization. The Audit Logs endpoint provides a log of all actions taken in the organization for security and monitoring purposes. To access these endpoints please generate an Admin API Key through the [API Platform Organization overview](#). Admin API keys cannot be used for non-administration endpoints. For best practices on setting up your organization, please refer to [this guide](#)

Admin API Keys

Admin API keys enable Organization Owners to programmatically manage various aspects of their organization, including users, projects, and API keys. These keys provide administrative capabilities, such as creating, updating, and deleting users; managing projects; and overseeing API key lifecycles.

Key Features of Admin API Keys:

User Management: Invite new users, update roles, and remove users from the organization.

Project Management: Create, update, archive projects, and manage user assignments within projects.

API Key Oversight: List, retrieve, and delete API keys associated with projects.

Only Organization Owners have the authority to create and utilize Admin API keys. To manage these keys, Organization Owners can navigate to the Admin Keys section of their API Platform dashboard.

For direct access to the Admin Keys management page, Organization Owners can use the following link:

<https://platform.openai.com/settings/organization/admin-keys>

It's crucial to handle Admin API keys with care due to their elevated permissions. Adhering to best practices, such as regular key rotation and assigning appropriate permissions, enhances security and ensures proper governance within the organization.

List all organization and project API keys.

GET https://api.openai.com/v1/organization/admin_api_keys

List organization API keys

Query parameters

after string or null Optional

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/  
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"  
3   -H "Content-Type: application/json"
```

Response

⌚

limit integer Optional Defaults to 20

order string Optional Defaults to asc

Returns

A list of admin and project API key objects.

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "organization.admin_api_key",
6        "id": "key_abc",
7        "name": "Main Admin Key",
8        "redacted_value": "sk-admin...def",
9        "created_at": 1711471533,
10       "last_used_at": 1711471534,
11       "owner": {
12         "type": "service_account",
13         "object": "organization.service_acc
14         "id": "sa_456",
15         "name": "My Service Account",
16         "created_at": 1711471533,
17         "role": "member"
18       }
19     }
20   ],
21   "first_id": "key_abc",
22   "last_id": "key_abc",
23   "has_more": false

```

Create admin API key

POST https://api.openai.com/v1/organization/admin_api_keys

Create an organization admin API key

Request body

Example request

curl ⌂

```

1 curl -X POST https://api.openai.com/v1/orga
2 -H "Authorization: Bearer $OPENAI_ADMIN_KE
3 -H "Content-Type: application/json" \
4 -d '{'

```

name string Required

Returns

The created [AdminApiKey](#) object.

```
5 "name": "New Admin Key"
```

Response

```
1 {
2   "object": "organization.admin_api_key",
3   "id": "key_xyz",
4   "name": "New Admin Key",
5   "redacted_value": "sk-admin...xyz",
6   "created_at": 1711471533,
7   "last_used_at": 1711471534,
8   "owner": {
9     "type": "user",
10    "object": "organization.user",
11    "id": "user_123",
12    "name": "John Doe",
13    "created_at": 1711471533,
14    "role": "owner"
15  },
16  "value": "sk-admin-1234abcd"
17 }
```

Retrieve admin API key

```
GET https://api.openai.com/v1/organization/admin_api_keys/{key_id}
```

Retrieve a single organization API key

Path parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

key_id string Required

Returns

The requested [AdminApiKey](#) object.

Response

```
1  {
2    "object": "organization.admin_api_key",
3    "id": "key_abc",
4    "name": "Main Admin Key",
5    "redacted_value": "sk-admin...xyz",
6    "created_at": 1711471533,
7    "last_used_at": 1711471534,
8    "owner": {
9      "type": "user",
10     "object": "organization.user",
11     "id": "user_123",
12     "name": "John Doe",
13     "created_at": 1711471533,
14     "role": "owner"
15   }
16 }
```

Delete admin API key

```
DELETE https://api.openai.com/v1/organization/admin_api_keys/{key_id}
```

Delete an organization admin API key

Path parameters

Example request

curl ⌂

```
1 curl -X DELETE https://api.openai.com/v1/org
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

key_id string Required

Returns

A confirmation object indicating the key was deleted.

```

1 {
2   "id": "key_abc",
3   "object": "organization.admin_api_key.delete"
4   "deleted": true

```

The admin API key object

Represents an individual Admin API key in an org.

created_at integer

The Unix timestamp (in seconds) of when the API key was created

id string

The identifier, which can be referenced in API endpoints

last_used_at integer or null

The Unix timestamp (in seconds) of when the API key was last used

name string

The name of the API key

object string

OBJECT The admin API key object

```

1 {
2   "object": "organization.admin_api_key",
3   "id": "key_abc",
4   "name": "Main Admin Key",
5   "redacted_value": "sk-admin...xyz",
6   "created_at": 1711471533,
7   "last_used_at": 1711471534,
8   "owner": {
9     "type": "user",
10    "object": "organization.user",
11    "id": "user_123",
12    "name": "John Doe",
13    "created_at": 1711471533,
14    "role": "owner"
15  }
16 }

```

The object type, which is always `organization.admin_api_key`

owner object

▼ Show properties

redacted_value string

The redacted value of the API key

value string

The value of the API key. Only shown on create.

Invites

Invite and manage invitations for an organization.

List invites

GET `https://api.openai.com/v1/organization/invites`

Returns a list of invites in the organization.

Query parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

Returns

A list of [Invite](#) objects.

Response

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "organization.invite",
6        "id": "invite-abc",
7        "email": "user@example.com",
8        "role": "owner",
9        "status": "accepted",
10       "invited_at": 1711471533,
11       "expires_at": 1711471533,
12       "accepted_at": 1711471533
13     }
14   ],
15   "first_id": "invite-abc",
16   "last_id": "invite-abc",
17   "has_more": false
18 }
```

Create invite

POST <https://api.openai.com/v1/organization/invites>

Create an invite for a user to the organization. The invite must be accepted by the user before they have access to the organization.

Request body**Example request**

curl ⌂

```

1 curl -X POST https://api.openai.com/v1/orga
2   -H "Authorization: Bearer $OPENAI_ADMIN_K
3   -H "Content-Type: application/json" \
4   -d '{
5     "email": "anotheruser@example.com",
6     "role": "reader",
```

email string Required

Send an email to this address

role string Required

owner or reader

projects array Optional

An array of projects to which membership is granted at the same time the org invite is accepted. If omitted, the user will be invited to the default project for compatibility with legacy behavior.

▼ Show properties

Returns

The created [Invite](#) object.

Retrieve invite

```
GET https://api.openai.com/v1/organization/invites/{invite_id}
```

Retrieves an invite.

Path parameters

```

7     "projects": [
8       {
9         "id": "project-xyz",
10        "role": "member"
11      },
12      {
13        "id": "project-abc",
14        "role": "owner"
15      }
16    ]
17 }
```

Response

```

1  {
2    "object": "organization.invite",
3    "id": "invite-def",
4    "email": "anotheruser@example.com",
5    "role": "reader",
6    "status": "pending",
7    "invited_at": 1711471533,
8    "expires_at": 1711471533,
9    "accepted_at": null,
10   "projects": [
11     {
12       "id": "project-xyz",
13       "role": "member"
14     },
15     {
16       "id": "project-abc",
17       "role": "owner"
18     }
19   ]
20 }
```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

invite_id string Required

The ID of the invite to retrieve.

Returns

The [Invite](#) object matching the specified ID.

Response

```

1  {
2      "object": "organization.invite",
3      "id": "invite-abc",
4      "email": "user@example.com",
5      "role": "owner",
6      "status": "accepted",
7      "invited_at": 1711471533,
8      "expires_at": 1711471533,
9      "accepted_at": 1711471533
10 }
```

Delete invite

```
DELETE https://api.openai.com/v1/organization/invites/{invite_id}
```

Delete an invite. If the invite has already been accepted, it cannot be deleted.

Path parameters**invite_id** string Required

The ID of the invite to delete.

Returns**Example request**
curl ⋮ □

```

1 curl -X DELETE https://api.openai.com/v1/org
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```

1 {
2     "object": "organization.invite.deleted",
3     "id": "invite-abc",
4     "deleted": true
5 }
```

Confirmation that the invite has been deleted

The invite object

Represents an individual `invite` to the organization.

accepted_at integer

The Unix timestamp (in seconds) of when the invite was accepted.

email string

The email address of the individual to whom the invite was sent

expires_at integer

The Unix timestamp (in seconds) of when the invite expires.

id string

The identifier, which can be referenced in API endpoints

invited_at integer

The Unix timestamp (in seconds) of when the invite was sent.

object string

The object type, which is always `organization.invite`

projects array

OBJECT The invite object

```
1  {
2    "object": "organization.invite",
3    "id": "invite-abc",
4    "email": "user@example.com",
5    "role": "owner",
6    "status": "accepted",
7    "invited_at": 1711471533,
8    "expires_at": 1711471533,
9    "accepted_at": 1711471533,
10   "projects": [
11     {
12       "id": "project-xyz",
13       "role": "member"
14     }
15   ]
16 }
```

The projects that were granted membership upon acceptance of the invite.

✓ Show properties

role string

owner or reader

status string

accepted, expired, or pending

Users

Manage users and their role in an organization.

List users

GET <https://api.openai.com/v1/organization/users>

Lists all of the users in the organization.

Query parameters

after string Optional

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/  
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"  
3   -H "Content-Type: application/json"
```

Response ⌂

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

emails array Optional

Filter by the email address of users.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

Returns

A list of [User](#) objects.

```

1  {
2      "object": "list",
3      "data": [
4          {
5              "object": "organization.user",
6              "id": "user_abc",
7              "name": "First Last",
8              "email": "user@example.com",
9              "role": "owner",
10             "added_at": 1711471533
11         }
12     ],
13     "first_id": "user-abc",
14     "last_id": "user-xyz",
15     "has_more": false

```

Modify user

POST https://api.openai.com/v1/organization/users/{user_id}

Modifies a user's role in the organization.

Path parameters

Example request

[curl](#) [Copy](#)

```

1 curl -X POST https://api.openai.com/v1/orga
2     -H "Authorization: Bearer $OPENAI_ADMIN_KE
3     -H "Content-Type: application/json" \
4     -d '{
5
6

```

user_id string Required

The ID of the user.

Request body**role** string Required

owner or reader

ReturnsThe updated [User](#) object.**Response**

```

1 {
2   "object": "organization.user",
3   "id": "user_abc",
4   "name": "First Last",
5   "email": "user@example.com",
6   "role": "owner",
7   "added_at": 1711471533
8 }
```

Retrieve user

GET https://api.openai.com/v1/organization/users/{user_id}

Retrieves a user by their identifier.

Path parameters**user_id** string Required

The ID of the user.

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```

1 {
2   "object": "organization.user",
```

Returns

The [User](#) object matching the specified ID.

```

3   "id": "user_abc",
4   "name": "First Last",
5   "email": "user@example.com",
6   "role": "owner",
7   "added_at": 1711471533
8 }
```

Delete user

```
DELETE https://api.openai.com/v1/organization/users/{user_id}
```

Deletes a user from the organization.

Path parameters

user_id string Required

The ID of the user.

Returns

Confirmation of the deleted user

Example request

curl ⌂

```

1 curl -X DELETE https://api.openai.com/v1/org
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```

1 {
2   "object": "organization.user.deleted",
3   "id": "user_abc",
4   "deleted": true
5 }
```

The user object

Represents an individual `user` within an organization.

`added_at` integer

The Unix timestamp (in seconds) of when the user was added.

`email` string

The email address of the user

`id` string

The identifier, which can be referenced in API endpoints

`name` string

The name of the user

`object` string

The object type, which is always `organization.user`

`role` string

`owner` or `reader`

OBJECT The user object

```
1 {  
2   "object": "organization.user",  
3   "id": "user_abc",  
4   "name": "First Last",  
5   "email": "user@example.com",  
6   "role": "owner",  
7   "added_at": 1711471533  
8 }
```

Projects

Manage the projects within an organization includes creation, updating, and archiving or projects. The Default project cannot be archived.

List projects

GET <https://api.openai.com/v1/organization/projects>

Returns a list of projects.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

include_archived boolean Optional Defaults to false

If `true` returns all projects including those that have been `archived`. Archived projects are not included by default.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

Returns

https://platform.openai.com/docs/api-reference/webhook_events/batch/expired

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "proj_abc",
6       "object": "organization.project",
7       "name": "Project example",
8       "created_at": 1711471533,
9       "archived_at": null,
10      "status": "active"
11    }
12  ],
13  "first_id": "proj-abc",
14  "last_id": "proj-xyz",
15  "has_more": false
16 }
```

A list of [Project](#) objects.

Create project

POST <https://api.openai.com/v1/organization/projects>

Create a new project in the organization. Projects can be created and archived, but cannot be deleted.

Request body

name string Required

The friendly name of the project, this name appears in reports.

Returns

The created [Project](#) object.

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/organ
2   -H "Authorization: Bearer $OPENAI_ADMIN_KE
3   -H "Content-Type: application/json" \
4   -d '{
5     "name": "Project ABC"
6   }'
```

Response

⌚

```
1 {
2   "id": "proj_abc",
3   "object": "organization.project",
4   "name": "Project ABC",
5   "created_at": 1711471533,
6   "archived_at": null,
7   "status": "active"
8 }
```

Retrieve project

```
GET https://api.openai.com/v1/organization/projects/{project_id}
```

Retrieves a project.

Path parameters

project_id string Required

The ID of the project.

Returns

The [Project](#) object matching the specified ID.

Example request

curl ⚡ 🔗

```
1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```
1 {
2   "id": "proj_abc",
3   "object": "organization.project",
4   "name": "Project example",
5   "created_at": 1711471533,
6   "archived_at": null,
7   "status": "active"
8 }
```

Modify project

```
POST https://api.openai.com/v1/organization/projects/{project_id}
```

Modifies a project in the organization.

Path parameters

Example request

curl ⚡ 🔗

```
1 curl -X POST https://api.openai.com/v1/organization/projects/{project_id}
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json" \
4   -d '{'
5
6 }
```

`"name": "Project DEF"`**project_id** string Required

The ID of the project.

Request body

name string Required

The updated name of the project, this name appears in reports.

Returns

The updated [Project](#) object.

Archive project

```
POST https://api.openai.com/v1/organization/projects/{project_id}/archive
```

Archives a project in the organization. Archived projects cannot be used or updated.

Path parameters

project_id string Required

Example request

`curl ⌂`

```
1 curl -X POST https://api.openai.com/v1/organization/projects/{project_id}/archive  
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"  
3 -H "Content-Type: application/json"
```

Response

`⌂`

```
1 {  
2   "id": "proj_abc",
```

The ID of the project.

Returns

The archived [Project](#) object.

```

3   "object": "organization.project",
4   "name": "Project DEF",
5   "created_at": 1711471533,
6   "archived_at": 1711471533,
7   "status": "archived"
8 }
```

The project object

Represents an individual project.

archived_at integer or null

The Unix timestamp (in seconds) of when the project was archived or `null`.

created_at integer

The Unix timestamp (in seconds) of when the project was created.

id string

The identifier, which can be referenced in API endpoints

name string

The name of the project. This appears in reporting.

object string

The object type, which is always `organization.project`

OBJECT The project object

```

1 {
2   "id": "proj_abc",
3   "object": "organization.project",
4   "name": "Project example",
5   "created_at": 1711471533,
6   "archived_at": null,
7   "status": "active"
8 }
```

status string

active or archived

Project users

Manage users within a project, including adding, updating roles, and removing users.

List project users

GET https://api.openai.com/v1/organization/projects/{project_id}/users

Returns a list of users in the project.

Path parameters

project_id string Required

The ID of the project.

Query parameters

Example request

curl ↗

```
1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "object": "organization.project",
6       "id": "user_abc",
7       "name": "First Last",
```

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

```

8     "email": "user@example.com",
9     "role": "owner",
10    "added_at": 1711471533
11   }
12   ],
13   "first_id": "user-abc",
14   "last_id": "user-xyz",
15   "has_more": false
16 }
```

Returns

A list of [ProjectUser](#) objects.

Create project user

```
POST https://api.openai.com/v1/organization/projects/{project_id}/users
```

Adds a user to the project. Users must already be members of the organization to be added to a project.

Path parameters

Example request

[curl](#) [Copy](#)

```

1 curl -X POST https://api.openai.com/v1/organization/projects/{project_id}/users \
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY" \
3   -H "Content-Type: application/json" \
4   -d '{
5     "user_id": "user_abc",
6     "role": "member"
7   }'
```

project_id string Required

The ID of the project.

Request body

role string Required

owner or member

user_id string Required

The ID of the user.

Response

```
1 {
2   "object": "organization.project.user",
3   "id": "user_abc",
4   "email": "user@example.com",
5   "role": "owner",
6   "added_at": 1711471533
7 }
```

Returns

The created [ProjectUser](#) object.

Retrieve project user

```
GET https://api.openai.com/v1/organization/projects/{project_id}/users/{user_id}
```

Retrieves a user in the project.

Path parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

project_id string Required

The ID of the project.

user_id string Required

The ID of the user.

Returns

The [ProjectUser](#) object matching the specified ID.

Response

```

1 {
2   "object": "organization.project.user",
3   "id": "user_abc",
4   "name": "First Last",
5   "email": "user@example.com",
6   "role": "owner",
7   "added_at": 1711471533
8 }
```

Modify project user

POST https://api.openai.com/v1/organization/projects/{project_id}/users/{user_id}

Modifies a user's role in the project.

Path parameters

project_id string Required

The ID of the project.

user_id string Required

Example request

curl ⌂

```

1 curl -X POST https://api.openai.com/v1/organ
2   -H "Authorization: Bearer $OPENAI_ADMIN_KE
3   -H "Content-Type: application/json" \
4   -d '{
5     "role": "owner"
6   }'
```

Response

```

1 {
2   "object": "organization.project.user",
```

The ID of the user.

Request body

role string Required

owner or member

```

3   "id": "user_abc",
4   "name": "First Last",
5   "email": "user@example.com",
6   "role": "owner",
7   "added_at": 1711471533
8 }
```

Returns

The updated [ProjectUser](#) object.

Delete project user

```
DELETE https://api.openai.com/v1/organization/projects/{project_id}/users/{user_id}
```

Deletes a user from the project.

Path parameters

project_id string Required

The ID of the project.

Example request

curl ⌂

```

1 curl -X DELETE https://api.openai.com/v1/org
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```

1 {
2   "object": "organization.project.user.del",
3   "id": "user_abc",
```

user_id string Required

The ID of the user.

```

4   "deleted": true
5

```

Returns

Confirmation that project has been deleted or an error in case of an archived project, which has no users

The project user object

Represents an individual user in a project.

added_at integer

The Unix timestamp (in seconds) of when the project was added.

email string

The email address of the user

id string

The identifier, which can be referenced in API endpoints

name string

The name of the user

OBJECT The project user object

```

1 {
2   "object": "organization.project.user",
3   "id": "user_abc",
4   "name": "First Last",
5   "email": "user@example.com",
6   "role": "owner",
7   "added_at": 1711471533
8 }

```

object string

The object type, which is always `organization.project.user`

role string

`owner` or `member`

Project service accounts

Manage service accounts within a project. A service account is a bot user that is not associated with a user.

If a user leaves an organization, their keys and membership in projects will no longer work. Service accounts do not have this limitation. However, service accounts can also be deleted from a project.

List project service accounts

```
GET https://api.openai.com/v1/organization/projects/{project_id}/se  
rvice_accounts
```

Returns a list of service accounts in the project.

Path parameters

project_id string Required

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/  
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"  
3   -H "Content-Type: application/json"
```

Response

⌂

The ID of the project.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

```

1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "organization.project",
6        "id": "svc_acct_abc",
7        "name": "Service Account",
8        "role": "owner",
9        "created_at": 1711471533
10       }
11     ],
12     "first_id": "svc_acct_abc",
13     "last_id": "svc_acct_xyz",
14     "has_more": false

```

Returns

A list of [ProjectServiceAccount](#) objects.

Create project service account

```
POST https://api.openai.com/v1/organization/projects/{project_id}/service_accounts
```

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/organization/projects/{project_id}/service_accounts
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
```

Creates a new service account in the project. This also returns an unredacted API key for the service account.

Path parameters

project_id string Required

The ID of the project.

Request body

name string Required

The name of the service account being created.

Returns

The created [ProjectServiceAccount](#) object.

```

3   -H "Content-Type: application/json" \
4   -d '{
5     "name": "Production App"
6   }'
```

Response

```

1  {
2    "object": "organization.project.service_account",
3    "id": "svc_acct_abc",
4    "name": "Production App",
5    "role": "member",
6    "created_at": 1711471533,
7    "api_key": {
8      "object": "organization.project.service_account.api_key",
9      "value": "sk-abcdefghijklmnop123",
10     "name": "Secret Key",
11     "created_at": 1711471533,
12     "id": "key_abc"
13   }
14 }
```

Retrieve project service account

```
GET https://api.openai.com/v1/organization/projects/{project_id}/service_accounts/{service_account_id}
```

Retrieves a service account in the project.

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
```

Path parameters

project_id string Required

The ID of the project.

service_account_id string Required

The ID of the service account.

Returns

The [ProjectServiceAccount](#) object matching the specified ID.

curl "Content-Type: application/json"

Response

```

1 {
2   "object": "organization.project.service_"
3   "id": "svc_acct_abc",
4   "name": "Service Account",
5   "role": "owner",
6   "created_at": 1711471533
7 }
```

Delete project service account

```
DELETE https://api.openai.com/v1/organization/projects/{project_id}/service_accounts/{service_account_id}
```

Deletes a service account from the project.

Path parameters

project_id string Required

The ID of the project.

Example request

curl ◀

```

1 curl -X DELETE https://api.openai.com/v1/org
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

```

1 {
2   "object": "organization.project.service_"
3   "id": "svc_acct_abc",
```

service_account_id string Required

The ID of the service account.

```
4 "deleted": true
5 }
```

Returns

Confirmation of service account being deleted, or an error in case of an archived project, which has no service accounts

The project service account object

Represents an individual service account in a project.

created_at integer

The Unix timestamp (in seconds) of when the service account was created

id string

The identifier, which can be referenced in API endpoints

name string

The name of the service account

object string

The object type, which is always `organization.project.service_account`

OBJECT The project service account object

```
1 {
2   "object": "organization.project.service_
3   "id": "svc_acct_abc",
4   "name": "Service Account",
5   "role": "owner",
6   "created_at": 1711471533
7 }
```

role string

owner or member

Project API keys

Manage API keys for a given project. Supports listing and deleting keys for users. This API does not allow issuing keys for users, as users need to authorize themselves to generate keys.

List project API keys

```
GET https://api.openai.com/v1/organization/projects/{project_id}/ap  
i_keys
```

Returns a list of API keys in the project.

Path parameters

project_id string Required

The ID of the project.

Query parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/  
2   -H "Authorization: Bearer $OPENAI_ADMIN_KE  
3   -H "Content-Type: application/json"
```

Response

```
1 {  
2   "object": "list",  
3   "data": [  
4     {  
5       "object": "organization.project"  
6       "redacted_value": "sk-abc...def"
```

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

Returns

A list of [ProjectApiKey](#) objects.

```

7   "name": "My API Key",
8   "created_at": 1711471533,
9   "last_used_at": 1711471534,
10  "id": "key_abc",
11  "owner": {
12    "type": "user",
13    "user": {
14      "object": "organization"
15      "id": "user_abc",
16      "name": "First Last",
17      "email": "user@example.",
18      "role": "owner",
19      "added_at": 1711471533
20    }
21  }
22  }
23  ],
24  "first_id": "key_abc",
25  "last_id": "key_xyz",
26  "has_more": false
27 }
```

Retrieve project API key

```
GET https://api.openai.com/v1/organization/projects/{project_id}/api_keys/{key_id}
```

Retrieves an API key in the project.

Path parameters**key_id** string Required**Example request**
[curl](#) [Copy](#)

```
1 curl https://api.openai.com/v1/organization/
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response
[Copy](#)

The ID of the API key.

project_id string Required

The ID of the project.

Returns

The [ProjectApiKey](#) object matching the specified ID.

```

1  {
2      "object": "organization.project.api_key",
3      "redacted_value": "sk-abc...def",
4      "name": "My API Key",
5      "created_at": 1711471533,
6      "last_used_at": 1711471534,
7      "id": "key_abc",
8      "owner": {
9          "type": "user",
10         "user": {
11             "object": "organization.project",
12             "id": "user_abc",
13             "name": "First Last",
14             "email": "user@example.com",
15             "role": "owner",
16             "added_at": 1711471533
17         }
18     }

```

Delete project API key

```
DELETE https://api.openai.com/v1/organization/projects/{project_id}/api_keys/{key_id}
```

Deletes an API key from the project.

Path parameters

key_id string Required

Example request

curl ⌂

```

1 curl -X DELETE https://api.openai.com/v1/org
2   -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3   -H "Content-Type: application/json"
```

Response

⌚

The ID of the API key.

project_id string Required

The ID of the project.

```

1  {
2    "object": "organization.project.api_key",
3    "id": "key_abc",
4    "deleted": true

```

Returns

Confirmation of the key's deletion or an error if the key belonged to a service account

The project API key object

Represents an individual API key in a project.

created_at integer

The Unix timestamp (in seconds) of when the API key was created

id string

The identifier, which can be referenced in API endpoints

last_used_at integer

The Unix timestamp (in seconds) of when the API key was last used.

name string

The name of the API key

OBJECT The project API key object

```

1  {
2    "object": "organization.project.api_key",
3    "redacted_value": "sk-abc...def",
4    "name": "My API Key",
5    "created_at": 1711471533,
6    "last_used_at": 1711471534,
7    "id": "key_abc",
8    "owner": {
9      "type": "user",
10     "user": {
11       "object": "organization.project",
12       "id": "user_abc",
13       "name": "First Last",
14       "email": "user@example.com",

```

object string

The object type, which is always `organization.project.api_key`

owner object

▼ Show properties

redacted_value string

The redacted value of the API key

```
15      "role": "owner",  
16      "created_at": 1711471533  
17    }  
18  }  
19 }
```

Project rate limits

Manage rate limits per model for projects. Rate limits may be configured to be equal to or lower than the organization's rate limits.

List project rate limits

```
GET https://api.openai.com/v1/organization/projects/{project_id}/ra  
te_limits
```

Returns the rate limits per model for a project.

Path parameters

```
1 curl https://api.openai.com/v1/organization/  
2   -H "Authorization: Bearer $OPENAI_ADMIN_KE  
3   -H "Content-Type: application/json"
```

project_id string Required

The ID of the project.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, beginning with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

limit integer Optional Defaults to 100

A limit on the number of objects to be returned. The default is 100.

Returns

A list of [ProjectRateLimit](#) objects.

Response

```
1  {
2      "object": "list",
3      "data": [
4          {
5              "object": "project.rate_limit",
6              "id": "rl-ada",
7              "model": "ada",
8              "max_requests_per_1_minute": 600,
9              "max_tokens_per_1_minute": 150000
10             "max_images_per_1_minute": 10
11         }
12     ],
13     "first_id": "rl-ada",
14     "last_id": "rl-ada",
15     "has_more": false
```

Modify project rate limit

```
POST https://api.openai.com/v1/organization/projects/{project_id}/rate_limits/{rate_limit_id}
```

Updates a project rate limit.

Path parameters

project_id string Required

The ID of the project.

rate_limit_id string Required

The ID of the rate limit.

Request body

batch_1_day_max_input_tokens integer Optional

The maximum batch input tokens per day. Only relevant for certain models.

max_audio_megabytes_per_1_minute integer Optional

The maximum audio megabytes per minute. Only relevant for certain models.

max_images_per_1_minute integer Optional

The maximum images per minute. Only relevant for certain models.

max_requests_per_1_day integer Optional

Example request

curl ⚡ ↗

```
1 curl -X POST https://api.openai.com/v1/organization/projects/{project_id}/rate_limits/{rate_limit_id} \
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY" \
3 -H "Content-Type: application/json" \
4 -d '{
5   "max_requests_per_1_minute": 500
6 }'
```

Response

↗

```
1 {
2   "object": "project.rate_limit",
3   "id": "rl-ada",
4   "model": "ada",
5   "max_requests_per_1_minute": 600,
6   "max_tokens_per_1_minute": 150000,
7   "max_images_per_1_minute": 10
8 }
```

The maximum requests per day. Only relevant for certain models.

max_requests_per_1_minute integer Optional

The maximum requests per minute.

max_tokens_per_1_minute integer Optional

The maximum tokens per minute.

Returns

The updated [ProjectRateLimit](#) object.

The project rate limit object

Represents a project rate limit config.

batch_1_day_max_input_tokens integer

The maximum batch input tokens per day. Only present for relevant models.

id string

The identifier, which can be referenced in API endpoints.

max_audio_megabytes_per_1_minute integer

The maximum audio megabytes per minute. Only present for relevant models.

OBJECT The project rate limit object

```
1 {  
2   "object": "project.rate_limit",  
3   "id": "rl_ada",  
4   "model": "ada",  
5   "max_requests_per_1_minute": 600,  
6   "max_tokens_per_1_minute": 150000,  
7   "max_images_per_1_minute": 10  
8 }
```

max_images_per_1_minute integer

The maximum images per minute. Only present for relevant models.

max_requests_per_1_day integer

The maximum requests per day. Only present for relevant models.

max_requests_per_1_minute integer

The maximum requests per minute.

max_tokens_per_1_minute integer

The maximum tokens per minute.

model string

The model this rate limit applies to.

object string

The object type, which is always `project.rate_limit`

Audit logs

Logs of user actions and configuration changes within this organization. To log events, an Organization Owner must activate logging in the [Data Controls Settings](#). Once activated, for security reasons, logging cannot be deactivated.

List audit logs

GET https://api.openai.com/v1/organization/audit_logs

List user actions and configuration changes within this organization.

Query parameters

actor_emails[] array Optional

Return only events performed by users with these emails.

actor_ids[] array Optional

Return only events performed by these actors. Can be a user ID, a service account ID, or an api key tracking ID.

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

before string Optional

A cursor for use in pagination. `before` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, starting with `obj_foo`, your subsequent call can include `before=obj_foo` in order to fetch the previous page of the list.

effective_at object Optional

Return only events whose `effective_at` (Unix seconds) is in this range.

Example request

curl ⚡

```
1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "id": "audit_log-xxx_yyyyymmdd",
6       "type": "project.archived",
7       "effective_at": 1722461446,
8       "actor": {
9         "type": "api_key",
10        "api_key": {
11          "type": "user",
12          "user": {
13            "id": "user-xxx",
14            "email": "user@example.com"
15          }
16        }
17      },
18      "project.archived": {
19        "id": "proj_abc"
20      },
21    },
22    {
23      "id": "audit_log-yyy__20240101"
```

✓ Show properties

event_types[] array Optional

Return only events with a `type` in one of these values. For example,

`project.created`. For all options, see the documentation for the [audit log object](#).

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

project_ids[] array Optional

Return only events for these projects.

resource_ids[] array Optional

Return only events performed on these targets. For example, a project ID updated.

Returns

A list of paginated [Audit Log](#) objects.

The audit log object

A log of a user action or configuration change within this organization.

```

24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55

```



OBJECT The audit log object

actor object

The actor who performed the audit logged action.

✓ Show properties

api_key.created object

The details for events with this [type](#).

✓ Show properties

api_key.deleted object

The details for events with this [type](#).

✓ Show properties

api_key.updated object

The details for events with this [type](#).

✓ Show properties

certificate.created object

The details for events with this [type](#).

✓ Show properties

certificate.deleted object

The details for events with this [type](#).

✓ Show properties

certificate.updated object

The details for events with this [type](#).

✓ Show properties

```
1  {
2      "id": "req_xxx_20240101",
3      "type": "api_key.created",
4      "effective_at": 1720804090,
5      "actor": {
6          "type": "session",
7          "session": {
8              "user": {
9                  "id": "user-xxx",
10                 "email": "user@example.com"
11             },
12             "ip_address": "127.0.0.1",
13             "user_agent": "Mozilla/5.0 (Win
14         }
15     },
16     "api_key.created": {
17         "id": "key_xxxx",
18         "data": {
19             "scopes": ["resource.operation"
20         }
21     }
}
```

certificates.activated object

The details for events with this [type](#).

✓ Show properties

certificates.deactivated object

The details for events with this [type](#).

✓ Show properties

checkpoint_permission.created object

The project and fine-tuned model checkpoint that the checkpoint permission was created for.

✓ Show properties

checkpoint_permission.deleted object

The details for events with this [type](#).

✓ Show properties

effective_at integer

The Unix timestamp (in seconds) of the event.

id string

The ID of this log.

invite.accepted object

The details for events with this [type](#).

✓ Show properties

invite.deleted object

The details for events with this [type](#).

✓ Show properties

invite.sent object

The details for events with this [type](#).

✓ Show properties

login.failed object

The details for events with this [type](#).

✓ Show properties

logout.failed object

The details for events with this [type](#).

✓ Show properties

organization.updated object

The details for events with this [type](#).

✓ Show properties

project object

The project that the action was scoped to. Absent for actions not scoped to projects.

Note that any admin actions taken via Admin API keys are associated with the default project.

✓ Show properties

project.archived object

The details for events with this [type](#).

✓ Show properties

project.created object

The details for events with this [type](#).

✓ Show properties

project.updated object

The details for events with this [type](#).

✓ Show properties

rate_limit.deleted object

The details for events with this [type](#).

✓ Show properties

rate_limit.updated object

The details for events with this [type](#).

✓ Show properties

service_account.created object

The details for events with this [type](#).

✓ Show properties

service_account.deleted object

The details for events with this [type](#).

✓ Show properties

service_account.updated object

The details for events with this [type](#).

✓ Show properties

type string

The event type.

user.added object

The details for events with this `type`.

✓ Show properties

user.deleted object

The details for events with this `type`.

✓ Show properties

user.updated object

The details for events with this `type`.

✓ Show properties

Usage

The **Usage API** provides detailed insights into your activity across the OpenAI API. It also includes a separate [Costs endpoint](#), which offers visibility into your spend, breaking down consumption by invoice line items and project IDs.

While the Usage API delivers granular usage data, it may not always reconcile perfectly with the Costs due to minor differences in how usage and spend are recorded. For financial purposes, we recommend using

the [Costs endpoint](#) or the [Costs tab](#) in the Usage Dashboard, which will reconcile back to your billing invoice.

Completions

GET <https://api.openai.com/v1/organization/usage/completions>

Get completions usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

api_key_ids array Optional

Return only usage for these API keys.

batch boolean Optional

If `true`, return batch jobs only. If `false`, return non-batch jobs only. By default, return both.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently `1m`, `1h` and `1d` are supported, default to `1d`.

end_time integer Optional

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization"
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "page",
3   "data": [
4     {
5       "object": "bucket",
6       "start_time": 1730419200,
7       "end_time": 1730505600,
8       "results": [
9         {
10          "object": "organization",
11          "input_tokens": 1000,
12          "output_tokens": 500,
13          "input_cached_tokens": 0,
14          "input_audio_tokens": 0,
15          "output_audio_tokens": 0,
16          "num_model_requests": 5,
17          "project_id": null,
```

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include `project_id`, `user_id`, `api_key_id`, `model`, `batch` or any combination of them.

limit integer Optional

Specifies the number of buckets to return.

`bucket_width=1d` : default: 7, max: 31

`bucket_width=1h` : default: 24, max: 168

`bucket_width=1m` : default: 60, max: 1440

models array Optional

Return only usage for these models.

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only usage for these projects.

user_ids array Optional

Return only usage for these users.

Returns

```

18
19
20
21
22
23
24
25
26
27
28 }

],
"has_more": true,
"next_page": "page_AAAAGdGxdEiJdKOAAAA

```

A list of paginated, time bucketed [Completions usage](#) objects.

Completions usage object

The aggregated completions usage details of the specific time bucket.

api_key_id string or null

When `group_by=api_key_id` , this field provides the API key ID of the grouped usage result.

batch boolean or null

When `group_by=batch` , this field tells whether the grouped usage result is batch or not.

input_audio_tokens integer

The aggregated number of audio input tokens used, including cached tokens.

input_cached_tokens integer

The aggregated number of text input tokens that has been cached from previous requests. For customers subscribe to scale tier, this includes scale tier tokens.

input_tokens integer

The aggregated number of text input tokens used, including cached tokens. For customers subscribe to scale tier, this includes scale tier tokens.

OBJECT Completions usage object

```
1  {
2    "object": "organization.usage.completion",
3    "input_tokens": 5000,
4    "output_tokens": 1000,
5    "input_cached_tokens": 4000,
6    "input_audio_tokens": 300,
7    "output_audio_tokens": 200,
8    "num_model_requests": 5,
9    "project_id": "proj_abc",
10   "user_id": "user-abc",
11   "api_key_id": "key_abc",
12   "model": "gpt-4o-mini-2024-07-18",
13   "batch": false
14 }
```

model string or null

When `group_by=model` , this field provides the model name of the grouped usage result.

num_model_requests integer

The count of requests made to the model.

object string**output_audio_tokens** integer

The aggregated number of audio output tokens used.

output_tokens integer

The aggregated number of text output tokens used. For customers subscribe to scale tier, this includes scale tier tokens.

project_id string or null

When `group_by=project_id` , this field provides the project ID of the grouped usage result.

user_id string or null

When `group_by=user_id` , this field provides the user ID of the grouped usage result.

Embeddings

GET <https://api.openai.com/v1/organization/usage/embeddings>

Get embeddings usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

api_key_ids array Optional

Return only usage for these API keys.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently 1m, 1h and 1d are supported, default to 1d.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include project_id, user_id, api_key_id, model or any combination of them.

limit integer Optional

Specifies the number of buckets to return.

bucket_width=1d : default: 7, max: 31

bucket_width=1h : default: 24, max: 168

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization"
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

⌚

```
1 {
2   "object": "page",
3   "data": [
4     {
5       "object": "bucket",
6       "start_time": 1730419200,
7       "end_time": 1730505600,
8       "results": [
9         {
10           "object": "organization",
11           "input_tokens": 16,
12           "num_model_requests": 2,
13           "project_id": null,
14           "user_id": null,
15           "api_key_id": null,
16           "model": null
17         }
18       ]
19     }
20   ],
21   "has_more": false,
22   "next_page": null
23 }
```

`bucket_width=1m` : default: 60, max: 1440

models array Optional

Return only usage for these models.

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only usage for these projects.

user_ids array Optional

Return only usage for these users.

Returns

A list of paginated, time bucketed Embeddings usage objects.

Embeddings usage object

The aggregated embeddings usage details of the specific time bucket.

api_key_id string or null

OBJECT Embeddings usage object

```
1 {  
2   "object": "organization.usage.embeddings"
```

When `group_by=api_key_id` , this field provides the API key ID of the grouped usage result.

input_tokens integer

The aggregated number of input tokens used.

model string or null

When `group_by=model` , this field provides the model name of the grouped usage result.

num_model_requests integer

The count of requests made to the model.

object string**project_id** string or null

When `group_by=project_id` , this field provides the project ID of the grouped usage result.

user_id string or null

When `group_by=user_id` , this field provides the user ID of the grouped usage result.

```
3 "input_tokens": 20,  
4 "num_model_requests": 2,  
5 "project_id": "proj_abc",  
6 "user_id": "user-abc",  
7 "api_key_id": "key_abc",  
8 "model": "text-embedding-ada-002-v2"  
9 }
```

Moderations

GET <https://api.openai.com/v1/organization/usage/moderations>

Get moderations usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

api_key_ids array Optional

Return only usage for these API keys.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently 1m, 1h and 1d are supported, default to 1d.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include project_id, user_id, api_key_id, model or any combination of them.

limit integer Optional

Specifies the number of buckets to return.

bucket_width=1d : default: 7, max: 31

bucket_width=1h : default: 24, max: 168

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "page",
3   "data": [
4     {
5       "object": "bucket",
6       "start_time": 1730419200,
7       "end_time": 1730505600,
8       "results": [
9         {
10           "object": "organization",
11           "input_tokens": 16,
12           "num_model_requests": 2,
13           "project_id": null,
14           "user_id": null,
15           "api_key_id": null,
16           "model": null
17         }
18       ]
19     }
20   ],
21   "has_more": false,
22   "next_page": null
23 }
```

`bucket_width=1m` : default: 60, max: 1440

models array Optional

Return only usage for these models.

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only usage for these projects.

user_ids array Optional

Return only usage for these users.

Returns

A list of paginated, time bucketed [Moderations usage](#) objects.

Moderations usage object

The aggregated moderations usage details of the specific time bucket.

api_key_id string or null

OBJECT Moderations usage object

```
1 {
2   "object": "organization.usage.moderation"
```

When `group_by=api_key_id` , this field provides the API key ID of the grouped usage result.

input_tokens integer

The aggregated number of input tokens used.

model string or null

When `group_by=model` , this field provides the model name of the grouped usage result.

num_model_requests integer

The count of requests made to the model.

object string**project_id** string or null

When `group_by=project_id` , this field provides the project ID of the grouped usage result.

user_id string or null

When `group_by=user_id` , this field provides the user ID of the grouped usage result.

```
3 "input_tokens": 20,  
4 "num_model_requests": 2,  
5 "project_id": "proj_abc",  
6 "user_id": "user-abc",  
7 "api_key_id": "key_abc",  
8 "model": "text-moderation"  
9 }
```

Images

GET <https://api.openai.com/v1/organization/usage/images>

Get images usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

api_key_ids array Optional

Return only usage for these API keys.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently 1m, 1h and 1d are supported, default to 1d.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include project_id, user_id, api_key_id, model, size, source or any combination of them.

limit integer Optional

Specifies the number of buckets to return.

bucket_width=1d : default: 7, max: 31

bucket_width=1h : default: 24, max: 168

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "page",
3   "data": [
4     {
5       "object": "bucket",
6       "start_time": 1730419200,
7       "end_time": 1730505600,
8       "results": [
9         {
10           "object": "organization",
11           "images": 2,
12           "num_model_requests": 2,
13           "size": null,
14           "source": null,
15           "project_id": null,
16           "user_id": null,
17           "api_key_id": null,
18           "model": null
19         }
20       ]
21     }
22   ],
23   "has_more": false,
24   "next_page": null
25 }
```

`bucket_width=1m` : default: 60, max: 1440

models array Optional

Return only usage for these models.

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only usage for these projects.

sizes array Optional

Return only usages for these image sizes. Possible values are `256x256` , `512x512` , `1024x1024` , `1792x1792` , `1024x1792` or any combination of them.

sources array Optional

Return only usages for these sources. Possible values are `image.generation` , `image.edit` , `image.variation` or any combination of them.

user_ids array Optional

Return only usage for these users.

Returns

A list of paginated, time bucketed [Images usage](#) objects.

Images usage object

The aggregated images usage details of the specific time bucket.

api_key_id string or null

When `group_by=api_key_id` , this field provides the API key ID of the grouped usage result.

images integer

The number of images processed.

model string or null

When `group_by=model` , this field provides the model name of the grouped usage result.

num_model_requests integer

The count of requests made to the model.

object string

project_id string or null

When `group_by=project_id` , this field provides the project ID of the grouped usage result.

size string or null

OBJECT Images usage object

```
1  {
2      "object": "organization.usage.images.re
3      "images": 2,
4      "num_model_requests": 2,
5      "size": "1024x1024",
6      "source": "image.generation",
7      "project_id": "proj_abc",
8      "user_id": "user-abc",
9      "api_key_id": "key_abc",
10     "model": "dall-e-3"
11 }
```

When `group_by=size`, this field provides the image size of the grouped usage result.

source string or null

When `group_by=source`, this field provides the source of the grouped usage result, possible values are `image.generation`, `image.edit`, `image.variation`.

user_id string or null

When `group_by=user_id`, this field provides the user ID of the grouped usage result.

Audio speeches

GET https://api.openai.com/v1/organization/usage/audio_speeches

Get audio speeches usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

api_key_ids array Optional

Return only usage for these API keys.

Example request

[curl](#) [Copy](#)

```
1 curl "https://api.openai.com/v1/organization  
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"  
3 -H "Content-Type: application/json"
```

Response

[Copy](#)

```
1 {  
2   "object": "page",  
3   "data": [  
4     {  
5       "object": "bucket",  
6       "start_time": 1730419200,  
7     }  
8   ]  
9 }
```

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently 1m, 1h and 1d are supported, default to 1d.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include project_id, user_id, api_key_id, model or any combination of them.

limit integer Optional

Specifies the number of buckets to return.

bucket_width=1d : default: 7, max: 31

bucket_width=1h : default: 24, max: 168

bucket_width=1m : default: 60, max: 1440

models array Optional

Return only usage for these models.

page string Optional

A cursor for use in pagination. Corresponding to the next_page field from the previous response.

project_ids array Optional

Return only usage for these projects.

```

7   "end_time": 1730505600,
8   "results": [
9     {
10       "object": "organization"
11       "characters": 45,
12       "num_model_requests": 1
13       "project_id": null,
14       "user_id": null,
15       "api_key_id": null,
16       "model": null
17     }
18   ]
19 }
20 ],
21 "has_more": false,
22 "next_page": null
23 }
```

user_ids array Optional

Return only usage for these users.

Returns

A list of paginated, time bucketed [Audio speeches usage](#) objects.

Audio speeches usage object

The aggregated audio speeches usage details of the specific time bucket.

api_key_id string or null

When `group_by=api_key_id` , this field provides the API key ID of the grouped usage result.

characters integer

The number of characters processed.

model string or null

When `group_by=model` , this field provides the model name of the grouped usage result.

num_model_requests integer

OBJECT Audio speeches usage object

```
1 {  
2   "object": "organization.usage.audio_spee  
3   "characters": 45,  
4   "num_model_requests": 1,  
5   "project_id": "proj_abc",  
6   "user_id": "user-abc",  
7   "api_key_id": "key_abc",  
8   "model": "tts-1"  
9 }
```

The count of requests made to the model.

object string

project_id string or null

When `group_by=project_id`, this field provides the project ID of the grouped usage result.

user_id string or null

When `group_by=user_id`, this field provides the user ID of the grouped usage result.

Audio transcriptions

```
GET https://api.openai.com/v1/organization/usage/audio_transcription
```

Get audio transcriptions usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

api_key_ids array Optional

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization"
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "page",
3   "data": [
4     {
5       "object": "bucket",
```

Return only usage for these API keys.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently 1m, 1h and 1d are supported, default to 1d.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include project_id, user_id, api_key_id, model or any combination of them.

limit integer Optional

Specifies the number of buckets to return.

bucket_width=1d : default: 7, max: 31

bucket_width=1h : default: 24, max: 168

bucket_width=1m : default: 60, max: 1440

models array Optional

Return only usage for these models.

page string Optional

A cursor for use in pagination. Corresponding to the next_page field from the previous response.

project_ids array Optional

```

6      "start_time": 1730419200,
7      "end_time": 1730505600,
8      "results": [
9        {
10          "object": "organization",
11          "seconds": 20,
12          "num_model_requests": 1
13          "project_id": null,
14          "user_id": null,
15          "api_key_id": null,
16          "model": null
17        }
18      ],
19    },
20  ],
21  "has_more": false,
22  "next_page": null
23 }
```

Return only usage for these projects.

user_ids array Optional

Return only usage for these users.

Returns

A list of paginated, time bucketed [Audio transcriptions usage](#) objects.

Audio transcriptions usage object

The aggregated audio transcriptions usage details of the specific time bucket.

api_key_id string or null

When `group_by=api_key_id`, this field provides the API key ID of the grouped usage result.

model string or null

When `group_by=model`, this field provides the model name of the grouped usage result.

num_model_requests integer

The count of requests made to the model.

OBJECT Audio transcriptions usage object

```
1 {
2   "object": "organization.usage.audio_transcription",
3   "seconds": 10,
4   "num_model_requests": 1,
5   "project_id": "proj_abc",
6   "user_id": "user-abc",
7   "api_key_id": "key_abc",
8   "model": "tts-1"
9 }
```

object string

project_id string or null

When `group_by=project_id`, this field provides the project ID of the grouped usage result.

seconds integer

The number of seconds processed.

user_id string or null

When `group_by=user_id`, this field provides the user ID of the grouped usage result.

Vector stores

GET https://api.openai.com/v1/organization/usage/vector_stores

Get vector stores usage details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

bucket_width string Optional Defaults to 1d

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization  
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"  
3 -H "Content-Type: application/json"
```

Response

```
1 {  
2   "object": "page",  
3   "data": [
```

Width of each time bucket in response. Currently `1m`, `1h` and `1d` are supported, default to `1d`.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include `project_id`.

limit integer Optional

Specifies the number of buckets to return.

`bucket_width=1d` : default: 7, max: 31

`bucket_width=1h` : default: 24, max: 168

`bucket_width=1m` : default: 60, max: 1440

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only usage for these projects.

Returns

A list of paginated, time bucketed [Vector stores](#) usage objects.

```

4      {
5        "object": "bucket",
6        "start_time": 1730419200,
7        "end_time": 1730505600,
8        "results": [
9          {
10            "object": "organization",
11            "usage_bytes": 1024,
12            "project_id": null
13          }
14        ]
15      },
16      "has_more": false,
17      "next_page": null
18    }
19  }
```

Vector stores usage object

The aggregated vector stores usage details of the specific time bucket.

object string

project_id string or null

When `group_by=project_id` , this field provides the project ID of the grouped usage result.

usage_bytes integer

The vector stores usage in bytes.

OBJECT Vector stores usage object

```
1 {
2   "object": "organization.usage.vector_sto
3   "usage_bytes": 1024,
4   "project_id": "proj_abc"
5 }
```

Code interpreter sessions

```
GET https://api.openai.com/v1/organization/usage/code_interpreter_sessions
```

Get code interpreter sessions usage details for the organization.

Query parameters

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently `1m`, `1h` and `1d` are supported, default to `1d`.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the usage data by the specified fields. Support fields include `project_id`.

limit integer Optional

Specifies the number of buckets to return.

`bucket_width=1d` : default: 7, max: 31

`bucket_width=1h` : default: 24, max: 168

`bucket_width=1m` : default: 60, max: 1440

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only usage for these projects.

```

1  {
2    "object": "page",
3    "data": [
4      {
5        "object": "bucket",
6        "start_time": 1730419200,
7        "end_time": 1730505600,
8        "results": [
9          {
10            "object": "organization",
11            "num_sessions": 1,
12            "project_id": null
13          }
14        ]
15      }
16    ],
17    "has_more": false,
18    "next_page": null
19  }

```

Returns

A list of paginated, time bucketed [Code interpreter sessions usage](#) objects.

Code interpreter sessions usage object

The aggregated code interpreter sessions usage details of the specific time bucket.

num_sessions integer

The number of code interpreter sessions.

object string

project_id string or null

When `group_by=project_id` , this field provides the project ID of the grouped usage result.

OBJECT Code interpreter sessions usage object 

```
1 {
2   "object": "organization.usage.code_interpreter_sessions_usage",
3   "num_sessions": 1,
4   "project_id": "proj_abc"
5 }
```

Costs

GET <https://api.openai.com/v1/organization/costs>

Example request 

curl 

Get costs details for the organization.

Query parameters

start_time integer Required

Start time (Unix seconds) of the query time range, inclusive.

bucket_width string Optional Defaults to 1d

Width of each time bucket in response. Currently only `1d` is supported, default to `1d`.

end_time integer Optional

End time (Unix seconds) of the query time range, exclusive.

group_by array Optional

Group the costs by the specified fields. Support fields include `project_id`, `line_item` and any combination of them.

limit integer Optional Defaults to 7

A limit on the number of buckets to be returned. Limit can range between 1 and 180, and the default is 7.

page string Optional

A cursor for use in pagination. Corresponding to the `next_page` field from the previous response.

project_ids array Optional

Return only costs for these projects.

```
1 curl "https://api.openai.com/v1/organization"
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json"
```

Response

```
1 {
2   "object": "page",
3   "data": [
4     {
5       "object": "bucket",
6       "start_time": 1730419200,
7       "end_time": 1730505600,
8       "results": [
9         {
10           "object": "organization",
11           "amount": {
12             "value": 0.06,
13             "currency": "usd"
14           },
15           "line_item": null,
16           "project_id": null
17         }
18       ]
19     }
20   ],
21   "has_more": false,
22   "next_page": null
23 }
```

Returns

A list of paginated, time bucketed [Costs](#) objects.

Costs object

The aggregated costs details of the specific time bucket.

amount object

The monetary value in its associated currency.

▽ Show properties

line_item string or null

When `group_by=line_item`, this field provides the line item of the grouped costs result.

object string

project_id string or null

When `group_by=project_id`, this field provides the project ID of the grouped costs result.

OBJECT Costs object

```
1 {
2   "object": "organization.costs.result",
3   "amount": {
4     "value": 0.06,
5     "currency": "usd"
6   },
7   "line_item": "Image models",
8   "project_id": "proj_abc"
9 }
```

Certificates Beta

Manage Mutual TLS certificates across your organization and projects.

[Learn more about Mutual TLS.](#)

Upload certificate

POST <https://api.openai.com/v1/organization/certificates>

Upload a certificate to the organization. This does **not** automatically activate the certificate.

Organizations can upload up to 50 certificates.

Request body

content string Required

The certificate content in PEM format

name string Optional

An optional name for the certificate

Returns

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/organization/certificates
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5   "name": "My Example Certificate",
6   "certificate": "-----BEGIN CERTIFICATE-----\n-----END CERTIFICATE-----"
7 }'
```

Response

```
1 {
2   "object": "certificate",
3   "id": "cert_abc",
4   "name": "My Example Certificate",
5   "created_at": 1234567,
6   "certificate_details": {
7     "valid_at": 12345667,
8     "expires_at": 12345678
```

A single [Certificate](#) object.

Get certificate

```
GET https://api.openai.com/v1/organization/certificates/{certificate_id}
```

Get a certificate that has been uploaded to the organization.

You can get a certificate regardless of whether it is active or not.

Path parameters

certificate_id string Required

Unique ID of the certificate to retrieve.

Query parameters

include array Optional

A list of additional fields to include in the response. Currently the only supported value is `content` to fetch the PEM content of the certificate.

Returns

Example request

curl ⌂

```
1 curl "https://api.openai.com/v1/organization
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
```

Response

⌚

```
1 {
2   "object": "certificate",
3   "id": "cert_abc",
4   "name": "My Example Certificate",
5   "created_at": 1234567,
6   "certificate_details": {
7     "valid_at": 1234567,
8     "expires_at": 12345678,
9     "content": "-----BEGIN CERTIFICATE-----"
10    }
11 }
```

A single [Certificate](#) object.

Modify certificate

```
POST https://api.openai.com/v1/organization/certificates/{certificate_id}
```

Modify a certificate. Note that only the name can be modified.

Request body

name string Required

The updated name for the certificate

Returns

The updated [Certificate](#) object.

Example request

curl ⌂

```
1 curl -X POST https://api.openai.com/v1/organization/certificates/{certificate_id} \
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY" \
3 -H "Content-Type: application/json" \
4 -d '{
5   "name": "Renamed Certificate"
6 }'
```

Response

```
1 {
2   "object": "certificate",
3   "id": "cert_abc",
4   "name": "Renamed Certificate",
5   "created_at": 1234567,
6   "certificate_details": {
7     "valid_at": 12345667,
8     "expires_at": 12345678
9   }
10 }
```

Delete certificate

```
DELETE https://api.openai.com/v1/organization/certificates/{certificate_id}
```

Delete a certificate from the organization.

The certificate must be inactive for the organization and all projects.

Returns

A confirmation object indicating the certificate was deleted.

Example request

curl ⌂

```
1 curl -X DELETE https://api.openai.com/v1/organization/certificates/{certificate_id}
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
```

Response

⌚

```
1 {
2   "object": "certificate.deleted",
3   "id": "cert_abc"
4 }
```

List organization certificates

```
GET https://api.openai.com/v1/organization/certificates
```

List uploaded certificates for this organization.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/certificates
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
```

Response

⌚

```
1 {
2   "object": "list",
3   "data": [
4     {
5       "object": "organization.certificate",
6       "id": "cert_abc"
7     }
8   ]
9 }
```

`obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

```

6     "id": "cert_abc",
7     "name": "My Example Certificate",
8     "active": true,
9     "created_at": 1234567,
10    "certificate_details": {
11      "valid_at": 12345667,
12      "expires_at": 12345678
13    }
14  },
15 ],
16  "first_id": "cert_abc",
17  "last_id": "cert_abc",
18  "has_more": false
19 }
```

Returns

A list of [Certificate](#) objects.

List project certificates

```
GET https://api.openai.com/v1/organization/projects/{project_id}/certificates
```

List certificates for this project.

Path parameters

Example request

curl ⌂

```
1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
```

Response

project_id string Required

The ID of the project.

Query parameters

after string Optional

A cursor for use in pagination. `after` is an object ID that defines your place in the list. For instance, if you make a list request and receive 100 objects, ending with `obj_foo`, your subsequent call can include `after=obj_foo` in order to fetch the next page of the list.

limit integer Optional Defaults to 20

A limit on the number of objects to be returned. Limit can range between 1 and 100, and the default is 20.

order string Optional Defaults to desc

Sort order by the `created_at` timestamp of the objects. `asc` for ascending order and `desc` for descending order.

Returns

A list of [Certificate](#) objects.

```
1  {
2    "object": "list",
3    "data": [
4      {
5        "object": "organization.project.certificate",
6        "id": "cert_abc",
7        "name": "My Example Certificate",
8        "active": true,
9        "created_at": 1234567,
10       "certificate_details": {
11         "valid_at": 12345667,
12         "expires_at": 12345678
13       }
14     },
15   ],
16   "first_id": "cert_abc",
17   "last_id": "cert_abc",
18   "has_more": false
```

Activate certificates for organization

POST <https://api.openai.com/v1/organization/certificates/activate>

Activate certificates at the organization level.

You can atomically and idempotently activate up to 10 certificates at a time.

Request body

certificate_ids array Required

Returns

A list of [Certificate](#) objects that were activated.

Example request

curl ⚡ [Copy](#)

```
1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5   "data": ["cert_abc", "cert_def"]
6 }'
```

Response

```
1 {
2   "object": "organization.certificate.activate",
3   "data": [
4     {
5       "object": "organization.certificate",
6       "id": "cert_abc",
7       "name": "My Example Certificate",
8       "active": true,
9       "created_at": 1234567,
10      "certificate_details": {
11        "valid_at": 12345667,
12        "expires_at": 12345678
13      }
14    },
15    {
16      "object": "organization.certificate",
17      "id": "cert_def",
18      "name": "My Example Certificate 2",
19      "active": true,
20      "created_at": 1234567,
```

Deactivate certificates for organization

POST <https://api.openai.com/v1/organization/certificates/deactivate>

Deactivate certificates at the organization level.

You can atomically and idempotently deactivate up to 10 certificates at a time.

Request body

`certificate_ids` array Required

Returns

A list of [Certificate](#) objects that were deactivated.

```

21     "certificate_details": {
22         "valid_at": 12345667,
23         "expires_at": 12345678
24     }
25 },
26 ],
27 }
```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5     "data": ["cert_abc", "cert_def"]
6 }'
```

Response

```

1 {
2     "object": "organization.certificate.deact
3     "data": [
4         {
5             "object": "organization.certificate",
6             "id": "cert_abc",
7             "name": "My Example Certificate",
8             "active": false,
9             "created_at": 1234567,
10            "certificate_details": {
11                "valid_at": 12345667,
12                "expires_at": 12345678
13            }
14        },
15        {
16            "object": "organization.certificate",
```

Activate certificates for project

```
POST https://api.openai.com/v1/organization/projects/{project_id}/certificates/activate
```

Activate certificates at the project level.

You can atomically and idempotently activate up to 10 certificates at a time.

Path parameters

project_id string Required

The ID of the project.

Request body

certificate_ids array Required

Returns

```

17     "id": "cert_def",
18     "name": "My Example Certificate 2",
19     "active": false,
20     "created_at": 1234567,
21     "certificate_details": {
22       "valid_at": 12345667,
23       "expires_at": 12345678
24     }
25   },

```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5   "data": ["cert_abc", "cert_def"]
6 }'

```

Response

```

1   {
2     "object": "organization.project.certificate",
3     "data": [
4       {
5         "object": "organization.project.certificate",
6         "id": "cert_abc",
7         "name": "My Example Certificate",
8         "active": true,
9         "created_at": 1234567,
10        "certificate_details": {
11          "valid_at": 12345667,
12          "expires_at": 12345678
13        }
14      },

```

A list of [Certificate](#) objects that were activated.

Deactivate certificates for project

```
POST https://api.openai.com/v1/organization/projects/{project_id}/certificates/deactivate
```

Deactivate certificates at the project level. You can atomically and idempotently deactivate up to 10 certificates at a time.

Path parameters

project_id string Required

The ID of the project.

Request body

certificate_ids array Required

Returns

```

15   {
16     "object": "organization.project.certificate",
17     "id": "cert_def",
18     "name": "My Example Certificate 2",
19     "active": true,
20     "created_at": 1234567,
21     "certificate_details": {
22       "valid_at": 12345667,
23       "expires_at": 12345678
24     }
25   },
26 }
```

Example request

curl ⌂

```

1 curl https://api.openai.com/v1/organization/
2 -H "Authorization: Bearer $OPENAI_ADMIN_KEY"
3 -H "Content-Type: application/json" \
4 -d '{
5   "data": ["cert_abc", "cert_def"]
6 }'
```

Response

```

1   {
2     "object": "organization.project.certificate",
3     "data": [
4       {
5         "object": "organization.project.certificate",
6         "id": "cert_abc",
7         "name": "My Example Certificate",
8         "active": false,
9         "created_at": 1234567,
10        "certificate_details": {
11          "valid_at": 12345667,
12          "expires_at": 12345678
13        }
14      }
15    ]
16  }
```

A list of [Certificate](#) objects that were deactivated.

The certificate object

Represents an individual [certificate](#) uploaded to the organization.

active boolean

Whether the certificate is currently active at the specified scope. Not returned when getting details for a specific certificate.

certificate_details object

✓ Show properties

created_at integer

The Unix timestamp (in seconds) of when the certificate was uploaded.

id string

The identifier, which can be referenced in API endpoints

name string

The name of the certificate.

```

13     }
14   },
15   {
16     "object": "organization.project.certi
17     "id": "cert_def",
18     "name": "My Example Certificate 2",
19     "active": false,
20     "created_at": 1234567,
21     "certificate_details": {
22       "valid_at": 12345667,
23       "expires_at": 12345678
24     }
25   },
26 ],

```

OBJECT The certificate object

```

1  {
2   "object": "certificate",
3   "id": "cert_abc",
4   "name": "My Certificate",
5   "created_at": 1234567,
6   "certificate_details": {
7     "valid_at": 1234567,
8     "expires_at": 12345678,
9     "content": "-----BEGIN CERTIFICATE-----
10  }
11 }

```

object string

The object type.

If creating, updating, or getting a specific certificate, the object type is `certificate`.

If listing, activating, or deactivating certificates for the organization, the object type is `organization.certificate`.

If listing, activating, or deactivating certificates for a project, the object type is `organization.project.certificate`.

Completions Legacy

Given a prompt, the model will return one or more predicted completions along with the probabilities of alternative tokens at each position. Most developer should use our [Chat Completions API](#) to leverage our best and newest models.

Create completion Legacy

POST <https://api.openai.com/v1/completions>

No streaming

Streaming

Creates a completion for the provided prompt and parameters.

Example req...

gpt-3.5-turbo-instruct ◊

curl ◊



Request body

model string Required

ID of the model to use. You can use the [List models](#) API to see all of your available models, or see our [Model overview](#) for descriptions of them.

prompt string or array Required

The prompt(s) to generate completions for, encoded as a string, array of strings, array of tokens, or array of token arrays.

Note that <|endoftext|> is the document separator that the model sees during training, so if a prompt is not specified the model will generate as if from the beginning of a new document.

best_of integer or null Optional Defaults to 1

Generates `best_of` completions server-side and returns the "best" (the one with the highest log probability per token). Results cannot be streamed.

When used with `n`, `best_of` controls the number of candidate completions and `n` specifies how many to return – `best_of` must be greater than `n`.

Note: Because this parameter generates many completions, it can quickly consume your token quota. Use carefully and ensure that you have reasonable settings for `max_tokens` and `stop`.

echo boolean or null Optional Defaults to false

Echo back the prompt in addition to the completion

frequency_penalty number or null Optional Defaults to 0

Number between -2.0 and 2.0. Positive values penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim.

```
1 curl https://api.openai.com/v1/completions \
2 -H "Content-Type: application/json" \
3 -H "Authorization: Bearer $OPENAI_API_KEY" \
4 -d '{
5   "model": "gpt-3.5-turbo-instruct",
6   "prompt": "Say this is a test",
7   "max_tokens": 7,
8   "temperature": 0}
```

Response

```
1 {
2   "id": "cmpl-uqkvlQyYK7bGYrRHQ0eXlWi7",
3   "object": "text_completion",
4   "created": 1589478378,
5   "model": "gpt-3.5-turbo-instruct",
6   "system_fingerprint": "fp_44709d6fcb",
7   "choices": [
8     {
9       "text": "\n\nThis is indeed a test",
10      "index": 0,
11      "logprobs": null,
12      "finish_reason": "length"
13    }
14  ],
15  "usage": {
16    "prompt_tokens": 5,
17    "completion_tokens": 7,
18    "total_tokens": 12
19  }
20 }
```

[See more information about frequency and presence penalties.](#)

logit_bias map Optional Defaults to null

Modify the likelihood of specified tokens appearing in the completion.

Accepts a JSON object that maps tokens (specified by their token ID in the GPT tokenizer) to an associated bias value from -100 to 100. You can use this [tokenizer tool](#) to convert text to token IDs. Mathematically, the bias is added to the logits generated by the model prior to sampling. The exact effect will vary per model, but values between -1 and 1 should decrease or increase likelihood of selection; values like -100 or 100 should result in a ban or exclusive selection of the relevant token.

As an example, you can pass `{"50256": -100}` to prevent the `<|endoftext|>` token from being generated.

logprobs integer or null Optional Defaults to null

Include the log probabilities on the `logprobs` most likely output tokens, as well the chosen tokens. For example, if `logprobs` is 5, the API will return a list of the 5 most likely tokens. The API will always return the `logprob` of the sampled token, so there may be up to `logprobs+1` elements in the response.

The maximum value for `logprobs` is 5.

max_tokens integer or null Optional Defaults to 16

The maximum number of [tokens](#) that can be generated in the completion.

The token count of your prompt plus `max_tokens` cannot exceed the model's context length. [Example Python code](#) for counting tokens.

n integer or null Optional Defaults to 1

How many completions to generate for each prompt.

Note: Because this parameter generates many completions, it can quickly consume your token quota. Use carefully and ensure that you have reasonable settings for `max_tokens` and `stop`.

presence_penalty number or null Optional Defaults to 0

Number between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics.

[See more information about frequency and presence penalties.](#)

seed integer or null Optional

If specified, our system will make a best effort to sample deterministically, such that repeated requests with the same `seed` and parameters should return the same result.

Determinism is not guaranteed, and you should refer to the `system_fingerprint` response parameter to monitor changes in the backend.

stop string / array / null Optional Defaults to null

Not supported with latest reasoning models `o3` and `o4-mini`.

Up to 4 sequences where the API will stop generating further tokens. The returned text will not contain the stop sequence.

stream boolean or null Optional Defaults to false

Whether to stream back partial progress. If set, tokens will be sent as data-only [server-sent events](#) as they become available, with the stream terminated by a `data: [DONE]` message. [Example Python code](#).

stream_options object or null Optional Defaults to null

Options for streaming response. Only set this when you set `stream: true`.

✓ Show properties

suffix string or null Optional Defaults to null

The suffix that comes after a completion of inserted text.

This parameter is only supported for `gpt-3.5-turbo-instruct`.

temperature number or null Optional Defaults to 1

What sampling temperature to use, between 0 and 2. Higher values like 0.8 will make the output more random, while lower values like 0.2 will make it more focused and deterministic.

We generally recommend altering this or `top_p` but not both.

top_p number or null Optional Defaults to 1

An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with `top_p` probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

We generally recommend altering this or `temperature` but not both.

user string Optional

A unique identifier representing your end-user, which can help OpenAI to monitor and detect abuse. [Learn more](#).

Returns

Returns a [completion](#) object, or a sequence of completion objects if the request is streamed.

The completion object Legacy

Represents a completion response from the API. Note: both the streamed and non-streamed response objects share the same shape (unlike the chat endpoint).

choices array

The list of completion choices the model generated for the input prompt.

▼ Show properties

created integer

The Unix timestamp (in seconds) of when the completion was created.

id string

A unique identifier for the completion.

model string

The model used for completion.

object string

The object type, which is always "text_completion"

system_fingerprint string

This fingerprint represents the backend configuration that the model runs with.

OBJECT The completion object

```
1  {
2    "id": "cmpl-uqkvlQyYK7bGYrRHQ0eXlWi7",
3    "object": "text_completion",
4    "created": 1589478378,
5    "model": "gpt-4-turbo",
6    "choices": [
7      {
8        "text": "\n\nThis is indeed a test",
9        "index": 0,
10       "logprobs": null,
11       "finish_reason": "length"
12     }
13   ],
14   "usage": {
15     "prompt_tokens": 5,
16     "completion_tokens": 7,
17     "total_tokens": 12
18   }
19 }
```

Can be used in conjunction with the `seed` request parameter to understand when backend changes have been made that might impact determinism.

usage object

Usage statistics for the completion request.

✓ Show properties