

OpenAI Platform

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 Optional Defaults to false

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

conversation string or object Optional Defaults to null

The conversation that this response belongs to. Items from this conversation are prepended to `input_items` for this response request. Input items and output items from this response are automatically added to this conversation after this response completes.

> Show possible types

include array Optional

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

`web_search_call.action.sources` : Include the sources of the web search tool call.

`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.

`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 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 Optional

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 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 Optional Defaults to true

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

previous_response_id string Optional

The unique ID of the previous response to the model. Use this to create multi-turn conversations. Learn more about [conversation state](#). Cannot be used in conjunction with `conversation`.

prompt object Optional

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

> Show properties

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](#).

prompt_cache_retention string Optional

The retention policy for the prompt cache. Set to `24h` to enable extended prompt caching, which keeps cached prefixes active for longer, up to a maximum of 24 hours. [Learn more](#).

reasoning object Optional

gpt-5 and o-series models only

Configuration options for [reasoning models](#).

> Show properties

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 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.

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 Optional Defaults to true

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

stream boolean 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 Optional Defaults to null

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

> Show properties

temperature number 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

> Show properties

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.

> Show possible types

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.

We support the following categories of tools:

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](#).

MCP Tools: Integrations with third-party systems via custom MCP servers or predefined connectors such as Google Drive and SharePoint. Learn more about [MCP 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.

> Show possible types

top_logprobs integer 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 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 Optional Defaults to disabled

The truncation strategy to use for the model response.

`auto` : If the input to this Response exceeds the model's context window size, the model will truncate the response to fit the context window by dropping items from the beginning of the conversation.

`disabled` (default): If the input size 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.

Text input

Image input

File input

Web search

File search

Streaming

Functions

Realtime

Example request

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 bedtime story about a unicorn."
7   }'
```

Response



```
1 {
2   "id": "resp_67ccd2bed1ec8190b14f964abc0542670bb6a6b452d3795b",
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    {
13      "type": "message",
14      "id": "msg_67ccd2bf17f0819081ff3bb2cf6508e60bb6a6b452d3795b",
15      "status": "completed",
16      "role": "assistant",
17      "content": [
18        {
```

```
19         "type": "output_text",
20         "text": "In a peaceful grove beneath a silver moon, a unicorn named Lumina d
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,
42 "truncation": "disabled",
43 "usage": {
44     "input_tokens": 36,
45     "input_tokens_details": {
46         "cached_tokens": 0
47     },
48     "output_tokens": 87,
49     "output_tokens_details": {
50         "reasoning_tokens": 0
51     },
52     "total_tokens": 123
53 },
54 "user": null,
55 "metadata": {}
56 }
```

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.

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.

Example request

curl ↕



```
1 curl https://api.openai.com/v1/responses/resp_123 \  
2   -H "Content-Type: application/json" \  
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response



```
1 {  
2   "id": "resp_67cb71b351908190a308f3859487620d06981a8637e6bc44",  
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_67cb71b3c2b0819084d481baaaf148f206981a8637e6bc44",
15      "status": "completed",
16      "role": "assistant",
17      "content": [
18        {
19          "type": "output_text",
20          "text": "Silent circuits hum, \nThoughts emerge in data streams– \nDigital",
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,
42  "truncation": "disabled",
43  "usage": {
44    "input_tokens": 32,
45    "input_tokens_details": {
46      "cached_tokens": 0
47    },
48    "output_tokens": 18,
49    "output_tokens_details": {
50      "reasoning_tokens": 0
51    },
52    "total_tokens": 50
```

```
53 },
54 "user": null,
55 "metadata": {}
56 }
```

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

A success message.

Example request

curl ↕



```
1 curl -X DELETE https://api.openai.com/v1/responses/resp_123 \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response



```
1 {
2   "id": "resp_6786a1bec27481909a17d673315b29f6",
3   "object": "response",
4   "deleted": true
5 }
```

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/responses/resp_123/cancel \
2   -H "Content-Type: application/json" \
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

Response



```
1 {
2   "id": "resp_67cb71b351908190a308f3859487620d06981a8637e6bc44",
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_67cb71b3c2b0819084d481baaaf148f206981a8637e6bc44",
15      "status": "completed",
16      "role": "assistant",
17      "content": [
18        {
```

```
19         "type": "output_text",
20         "text": "Silent circuits hum, \nThoughts emerge in data streams– \nDigital
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,
42 "truncation": "disabled",
43 "usage": {
44     "input_tokens": 32,
45     "input_tokens_details": {
46         "cached_tokens": 0
47     },
48     "output_tokens": 18,
49     "output_tokens_details": {
50         "reasoning_tokens": 0
51     },
52     "total_tokens": 50
53 },
54 "user": null,
55 "metadata": {}
56 }
```

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.

include array Optional

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

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.

Example request

curl ↕



```
1 curl https://api.openai.com/v1/responses/resp_abc123/input_items \  
2   -H "Content-Type: application/json" \  
3   -H "Authorization: Bearer $OPENAI_API_KEY"
```

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 bedtime story about a unicorn."
12       }
13     ]
14   },
15 ],
16 "first_id": "msg_abc123",
17 "last_id": "msg_abc123",
18 "has_more": false
19 }
```

Get input token counts

POST https://api.openai.com/v1/responses/input_tokens

Get input token counts

Request body

conversation string or object Optional Defaults to null

The conversation that this response belongs to. Items from this conversation are prepended to `input_items` for this response request. Input items and output items from this response are automatically added to this conversation after this response completes.

> Show possible types

input string or array Optional

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

> Show possible types

instructions string Optional

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

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` Optional

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

previous_response_id `string` Optional

The unique ID of the previous response to the model. Use this to create multi-turn conversations. Learn more about [conversation state](#). Cannot be used in conjunction with `conversation`.

reasoning `object` Optional

gpt-5 and o-series models only

Configuration options for [reasoning models](#).

> Show properties

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](#)

> Show properties

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.

> Show possible types

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.

> Show possible types

truncation `string` Optional

The truncation strategy to use for the model response. - `auto`: If the input to this Response exceeds the model's context window size, the model will truncate the response to fit the context window by dropping items from the beginning of the conversation. - `disabled` (default): If the input size will exceed the context window size for a model, the request will fail with a 400 error.

Returns

The input token counts.

```
1 {  
2   object: "response.input_tokens"  
3   input_tokens: 123  
4 }
```



Example request

curl ↕



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

Response



```
1 {  
2   "object": "response.input_tokens",  
3   "input_tokens": 11  
4 }
```

The response object

background boolean

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

conversation object

The conversation that this response belongs to. Input items and output items from this response are automatically added to this conversation.

> Show properties

created_at number

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

error object

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

> Show properties

id string

Unique identifier for this Response.

incomplete_details object

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

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

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

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` `SDK Only`

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`

The unique ID of the previous response to the model. Use this to create multi-turn conversations. Learn more about [conversation state](#). Cannot be used in conjunction with `conversation` .

prompt `object`

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](#).

prompt_cache_retention `string`

The retention policy for the prompt cache. Set to `24h` to enable extended prompt caching, which keeps cached prefixes active for longer, up to a maximum of 24 hours. [Learn more](#).

reasoning `object`

gpt-5 and 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

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.

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

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.

We support the following categories of tools:

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](#).

MCP Tools: Integrations with third-party systems via custom MCP servers or predefined connectors such as Google Drive and SharePoint. Learn more about [MCP 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.

> Show possible types

top_logprobs integer

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

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

The truncation strategy to use for the model response.

`auto` : If the input to this Response exceeds the model's context window size, the model will truncate the response to fit the context window by dropping items from the beginning of the conversation.

`disabled` (default): If the input size 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](#).



OBJECT The response object

```
1  {
2    "id": "resp_67ccd3a9da748190baa7f1570fe91ac604becb25c45c1d41",
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_67ccd3acc8d48190a77525dc6de64b4104becb25c45c1d41",
15       "status": "completed",
16       "role": "assistant",
17       "content": [
18         {
19           "type": "output_text",
20           "text": "The image depicts a scenic landscape with a wooden boardwalk or path",
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_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": {}
56 }
```

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 bedtime story about a unicorn."  
12        }  
13    ]  
14 }  
15 ],  
16 "first_id": "msg_abc123",  
17 "last_id": "msg_abc123",  
18 "has_more": false  
19 }
```

[< PREVIOUS](#)
Introduction

[NEXT >](#)
Conversations