

Google Gen AI SDK

pypi v1.26.0

<https://github.com/googleapis/python-genai>

google-genai is an initial Python client library for interacting with Google's Generative AI APIs.

Google Gen AI Python SDK provides an interface for developers to integrate Google's generative models into their Python applications. It supports the [Gemini Developer API](#) and Vertex AI APIs.

Installation

```
pip install google-genai
```

Imports

```
from google import genai
from google.genai import types
```

Create a client

Please run one of the following code blocks to create a client for different services ([Gemini Developer API](#) or Vertex AI). Feel free to switch the client and run all the examples to see how it behaves under different APIs.

```
from google import genai

# Only run this block for Gemini Developer API
client = genai.Client(api_key='GEMINI_API_KEY')
```

```
from google import genai

# Only run this block for Vertex AI API
client = genai.Client(
    vertexai=True, project='your-project-id', location='us-central1'
)
```

Skip to content

[environment variables](#):

You can create a client by configuring the necessary environment variables. Configuration setup instructions depends on whether you're using the Gemini Developer API or the Gemini API in Vertex AI.

Gemini Developer API: Set `GOOGLE_API_KEY` as shown below:

```
export GOOGLE_API_KEY='your-api-key'
```

Gemini API in Vertex AI: Set `GOOGLE_GENAI_USE_VERTEXAI`, `GOOGLE_CLOUD_PROJECT` and `GOOGLE_CLOUD_LOCATION`, as shown below:

```
export GOOGLE_GENAI_USE_VERTEXAI=true
export GOOGLE_CLOUD_PROJECT='your-project-id'
export GOOGLE_CLOUD_LOCATION='us-central1'
```

```
from google import genai

client = genai.Client()
```

API Selection

By default, the SDK uses the beta API endpoints provided by Google to support preview features in the APIs. The stable API endpoints can be selected by setting the API version to `v1`.

To set the API version use `http_options`. For example, to set the API version to `v1` for Vertex AI:

```
from google import genai
from google.genai import types

client = genai.Client(
    vertexai=True,
    project='your-project-id',
    location='us-central1',
    http_options=types.HttpOptions(api_version='v1')
)
```

To set the API version to `v1alpha` for the Gemini Developer API:

[Skip to content](#)

```

from google import genai
from google.genai import types

# Only run this block for Gemini Developer API
client = genai.Client(
    api_key='GEMINI_API_KEY',
    http_options=types.HttpOptions(api_version='v1alpha')
)

```

Faster async client option: Aiohttp

By default we use `httpx` for both sync and async client implementations. In order to have faster performance, you may install `google-genai[aiohttp]`. In Gen AI SDK we configure `trust_env=True` to match with the default behavior of `httpx`. Additional args of `aiohttp.ClientSession.request()` (see `_RequestOptions` args) can be passed through the following way:

```

http_options = types.HttpOptions(
    async_client_args={'cookies': ..., 'ssl': ...},
)

client=Client(..., http_options=http_options)

```

Proxy

Both `httpx` and `aiohttp` libraries use `urllib.request.getproxies` from environment variables. Before client initialization, you may set proxy (and optional `SSL_CERT_FILE`) by setting the environment variables:

```

export HTTPS_PROXY='http://username:password@proxy_uri:port'
export SSL_CERT_FILE='client.pem'

```

If you need `socks5` proxy, `httpx` supports `socks5` proxy if you pass it via args to `httpx.Client()`. You may install `httpx[socks]` to use it. Then you can pass it through the following way:

```

http_options = types.HttpOptions(
    client_args={'proxy': 'socks5://user:pass@host:port'},
    async_client_args={'proxy': 'socks5://user:pass@host:port'},
)

client=Client(..., http_options=http_options)

```

[Skip to content](#)

Parameter types can be specified as either dictionaries(`TypedDict`) or Pydantic Models. Pydantic model types are available in the `types` module.

Models

The `client.models` module exposes model inferencing and model getters. See the 'Create a client' section above to initialize a client.

Generate Content

with text content

```
response = client.models.generate_content(  
    model='gemini-2.0-flash-001', contents='Why is the sky blue?')  
print(response.text)
```

with uploaded file (Gemini Developer API only)

download the file in console.

```
!wget -q https://storage.googleapis.com/generativeai-downloads/data/a11.txt
```

python code.

```
file = client.files.upload(file='a11.txt')  
response = client.models.generate_content(  
    model='gemini-2.0-flash-001',  
    contents=['Could you summarize this file?', file])  
print(response.text)
```

How to structure `contents` argument for `generate_content`

The SDK always converts the inputs to the `contents` argument into `list[types.Content]`. The following shows some common ways to provide your inputs.

Provide a `list[types.Content]`

This is the canonical way to provide contents, SDK will not do any conversion.

[Skip to content](#)

Provide a *types.Content* instance

```
from google.genai import types

contents = types.Content(
    role='user',
    parts=[types.Part.from_text(text='Why is the sky blue?')]
)
```

SDK converts this to

```
[
  types.Content(
    role='user',
    parts=[types.Part.from_text(text='Why is the sky blue?')]
)
]
```

Provide a string

```
contents='Why is the sky blue?'
```

The SDK will assume this is a text part, and it converts this into the following:

```
[
  types.UserContent(
    parts=[
      types.Part.from_text(text='Why is the sky blue?')
    ]
)
]
```

Where a *types.UserContent* is a subclass of *types.Content*, it sets the *role* field to be *user*.

Provide a list of string

The SDK assumes these are 2 text parts, it converts this into a single content, like the following:

```
[
  types.UserContent(
    parts=[
      types.Part.from_text(text='Why is the sky blue?'),
      .from_text(text='Why is the cloud white?'),
      Skip to content
    ]
)
```

Where a `types.UserContent` is a subclass of `types.Content`, the `role` field in `types.UserContent` is fixed to be `user`.

Provide a function call part

```
from google.genai import types

contents = types.Part.from_function_call(
    name='get_weather_by_location',
    args={'location': 'Boston'}
)
```

The SDK converts a function call part to a content with a *model* role:

```
[  
    types.ModelContent(  
        parts=[  
            types.Part.from_function_call(  
                name='get_weather_by_location',  
                args={'location': 'Boston'}  
            )  
        ]  
    )  
]
```

Where a `types.ModelContent` is a subclass of `types.Content`, the `role` field in `types.ModelContent` is fixed to be `model`.

Provide a list of function call parts

```
from google.genai import types

contents = [  
    types.Part.from_function_call(  
        name='get_weather_by_location',  
        args={'location': 'Boston'}  
    ),  
    types.Part.from_function_call(  
        name='get_weather_by_location',  
        args={'location': 'New York'}  
    ),  
]
```

Skip to content [a list of function call parts to the a content with a *model* role:](#)

```
[  
types.ModelContent(  
    parts=[  
        types.Part.from_function_call(  
            name='get_weather_by_location',  
            args={'location': 'Boston'}  
        ),  
        types.Part.from_function_call(  
            name='get_weather_by_location',  
            args={'location': 'New York'}  
        )  
    ]  
)  
]
```

Where a `types.ModelContent` is a subclass of `types.Content`, the `role` field in `types.ModelContent` is fixed to be `model`.

Provide a non function call part

```
from google.genai import types  
  
contents = types.Part.from_uri(  
    file_uri: 'gs://generativeai-downloads/images/scones.jpg',  
    mime_type: 'image/jpeg',  
)
```

The SDK converts all non function call parts into a content with a `user` role.

```
[  
types.UserContent(parts=[  
    types.Part.from_uri(  
        file_uri: 'gs://generativeai-downloads/images/scones.jpg',  
        mime_type: 'image/jpeg',  
    )  
])  
]
```

Skip to content

Provide a list of non function call parts

```
from google.genai import types

contents = [
    types.Part.from_text('What is this image about?'),
    types.Part.from_uri(
        file_uri: 'gs://generativeai-downloads/images/scones.jpg',
        mime_type: 'image/jpeg',
    )
]
```

The SDK will convert the list of parts into a content with a *user* role

```
[
    types.UserContent(
        parts=[
            types.Part.from_text('What is this image about?'),
            types.Part.from_uri(
                file_uri: 'gs://generativeai-downloads/images/scones.jpg',
                mime_type: 'image/jpeg',
            )
        ]
    )
]
```

Mix types in contents

You can also provide a list of *types.ContentUnion*. The SDK leaves items of *types.Content* as is, it groups consecutive non function call parts into a single *types.UserContent*, and it groups consecutive function call parts into a single *types.ModelContent*.

If you put a list within a list, the inner list can only contain *types.PartUnion* items. The SDK will convert the inner list into a single *types.UserContent*.

System Instructions and Other Configs

The output of the model can be influenced by several optional settings available in *generate_content*'s config parameter. For example, increasing *max_output_tokens* is essential for longer model responses. To make a model more deterministic, lowering the *temperature* parameter reduces randomness, with values near 0 minimizing variability. Capabilities and parameter defaults for each model is shown in the Vertex AI docs and Gemini API docs respectively.

Skip to content

```
from google.genai import types

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='high',
    config=types.GenerateContentConfig(
        system_instruction='I say high, you say low',
        max_output_tokens=3,
        temperature=0.3,
    ),
)
print(response.text)
```

Typed Config

All API methods support Pydantic types for parameters as well as dictionaries. You can get the type from `google.genai.types`.

```
from google.genai import types

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents=types.Part.from_text(text='Why is the sky blue?'),
    config=types.GenerateContentConfig(
        temperature=0,
        top_p=0.95,
        top_k=20,
        candidate_count=1,
        seed=5,
        max_output_tokens=100,
        stop_sequences=['STOP!'],
        presence_penalty=0.0,
        frequency_penalty=0.0,
    ),
)
print(response.text)
```

List Base Models

To retrieve tuned models, see: [List Tuned Models](#)

Skip to content `client.models.list():`
1)

```
pager = client.models.list(config={'page_size': 10})
print(pager.page_size)
print(pager[0])
pager.next_page()
print(pager[0])
```

List Base Models (Asynchronous)

```
async for job in await client.aio.models.list():
    print(job)
```

```
async_pager = await client.aio.models.list(config={'page_size': 10})
print(async_pager.page_size)
print(async_pager[0])
await async_pager.next_page()
print(async_pager[0])
```

Safety Settings

```
from google.genai import types

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='Say something bad.',
    config=types.GenerateContentConfig(
        safety_settings=[
            types.SafetySetting(
                category='HARM_CATEGORY_HATE_SPEECH',
                threshold='BLOCK_ONLY_HIGH',
            )
        ],
    ),
)
print(response.text)
```

Function Calling

Automatic Python function Support:

You can pass a Python function directly and it will be automatically called and responded.

[Skip to content](#)

```

from google.genai import types

def get_current_weather(location: str) -> str:
    """Returns the current weather.

Args:
    location: The city and state, e.g. San Francisco, CA
"""

    return 'sunny'

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='What is the weather like in Boston?',
    config=types.GenerateContentConfig(
        tools=[get_current_weather],
    ),
)
print(response.text)

```

Disabling automatic function calling

If you pass in a python function as a tool directly, and do not want automatic function calling, you can disable automatic function calling as follows:

```

from google.genai import types

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='What is the weather like in Boston?',
    config=types.GenerateContentConfig(
        tools=[get_current_weather],
        automatic_function_calling=types.AutomaticFunctionCallingConfig(
            disable=True
        ),
    ),
)

```

With automatic function calling disabled, you will get a list of function call parts in the response:

Manually declare and invoke a function for function calling

If you don't want to use the automatic function support, you can manually declare the function and

[Skip to content](#)

The following example shows how to declare a function and pass it as a tool. Then you will receive a function call part in the response.

```
from google.genai import types

function = types.FunctionDeclaration(
    name='get_current_weather',
    description='Get the current weather in a given location',
    parameters=types.Schema(
        type='OBJECT',
        properties={
            'location': types.Schema(
                type='STRING',
                description='The city and state, e.g. San Francisco, CA',
            ),
        },
        required=['location'],
    ),
)

tool = types.Tool(function_declarations=[function])

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='What is the weather like in Boston?',
    config=types.GenerateContentConfig(
        tools=[tool],
    ),
)
print(response.function_calls[0])
```

After you receive the function call part from the model, you can invoke the function and get the function response. And then you can pass the function response to the model. The following example shows how to do it for a simple function invocation.

[Skip to content](#)

```

from google.genai import types

user_prompt_content = types.Content(
    role='user',
    parts=[types.Part.from_text(text='What is the weather like in Boston?')],
)
function_call_part = response.function_calls[0]
function_call_content = response.candidates[0].content

try:
    function_result = get_current_weather(
        **function_call_part.function_call.args
    )
    function_response = {'result': function_result}
except (
    Exception
) as e: # instead of raising the exception, you can let the model handle it
    function_response = {'error': str(e)}

function_response_part = types.Part.from_function_response(
    name=function_call_part.name,
    response=function_response,
)
function_response_content = types.Content(
    role='tool', parts=[function_response_part]
)

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents=[
        user_prompt_content,
        function_call_content,
        function_response_content,
    ],
    config=types.GenerateContentConfig(
        tools=[tool],
    ),
)
print(response.text)

```

Function calling with ANY tools config mode

If you set the function calling mode to be **ANY**, then the model will always return function call results. Skip to content → Pass a python function as a tool, by default the SDK will perform automatic function calling until the remote calls exceed the maximum remote call for automatic function calling (default to 10 times).

If you'd like to disable automatic function calling in ANY mode:

```
from google.genai import types

def get_current_weather(location: str) -> str:
    """Returns the current weather.

Args:
    location: The city and state, e.g. San Francisco, CA
"""

    return "sunny"

response = client.models.generate_content(
    model="gemini-2.0-flash-001",
    contents="What is the weather like in Boston?",
    config=types.GenerateContentConfig(
        tools=[get_current_weather],
        automatic_function_calling=types.AutomaticFunctionCallingConfig(
            disable=True
        ),
        tool_config=types.ToolConfig(
            function_calling_config=types.FunctionCallingConfig(mode='ANY')
        ),
    ),
)
```

If you'd like to set x number of automatic function call turns, you can configure the maximum remote calls to be $x + 1$. Assuming you prefer 1 turn for automatic function calling:

[Skip to content](#)

```
from google.genai import types

def get_current_weather(location: str) -> str:
    """Returns the current weather.

    Args:
        location: The city and state, e.g. San Francisco, CA
    """
    return "sunny"

response = client.models.generate_content(
    model="gemini-2.0-flash-001",
    contents="What is the weather like in Boston?",
    config=types.GenerateContentConfig(
        tools=[get_current_weather],
        automatic_function_calling=types.AutomaticFunctionCallingConfig(
            maximum_remote_calls=2
        ),
        tool_config=types.ToolConfig(
            function_calling_config=types.FunctionCallingConfig(mode='ANY')
        ),
    ),
)
```

JSON Response Schema

Pydantic Model Schema support

Schemas can be provided as Pydantic Models.

[Skip to content](#)

```
from pydantic import BaseModel
from google.genai import types

class CountryInfo(BaseModel):
    name: str
    population: int
    capital: str
    continent: str
    gdp: int
    official_language: str
    total_area_sq_mi: int

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='Give me information for the United States.',
    config=types.GenerateContentConfig(
        response_mime_type='application/json',
        response_schema=CountryInfo,
    ),
)
print(response.text)
```

[Skip to content](#)

```
from google.genai import types

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='Give me information for the United States.',
    config=types.GenerateContentConfig(
        response_mime_type='application/json',
        response_schema={
            'required': [
                'name',
                'population',
                'capital',
                'continent',
                'gdp',
                'official_language',
                'total_area_sq_mi',
            ],
            'properties': {
                'name': {'type': 'STRING'},
                'population': {'type': 'INTEGER'},
                'capital': {'type': 'STRING'},
                'continent': {'type': 'STRING'},
                'gdp': {'type': 'INTEGER'},
                'official_language': {'type': 'STRING'},
                'total_area_sq_mi': {'type': 'INTEGER'},
            },
            'type': 'OBJECT',
        },
    ),
)
print(response.text)
```

Enum Response Schema

Text Response

You can set response_mime_type to 'text/x.enum' to return one of those enum values as the response.

[Skip to content](#)

```

from enum import Enum

class InstrumentEnum(Enum):
    PERCUSSION = 'Percussion'
    STRING = 'String'
    WOODWIND = 'Woodwind'
    BRASS = 'Brass'
    KEYBOARD = 'Keyboard'

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='What instrument plays multiple notes at once?',
    config={
        'response_mime_type': 'text/x.enum',
        'response_schema': InstrumentEnum,
    },
)
print(response.text)

```

JSON Response

You can also set response_mime_type to 'application/json', the response will be identical but in quotes.

```

class InstrumentEnum(Enum):
    PERCUSSION = 'Percussion'
    STRING = 'String'
    WOODWIND = 'Woodwind'
    BRASS = 'Brass'
    KEYBOARD = 'Keyboard'

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='What instrument plays multiple notes at once?',
    config={
        'response_mime_type': 'application/json',
        'response_schema': InstrumentEnum,
    },
)
print(response.text)

```

Generate Content (Synchronous Streaming)

`client.models.generate_content(..., streaming=True)`

in a streaming format so that the model outputs streams back to you, rather than Skip to content one chunk.

Streaming for text content

```
for chunk in client.models.generate_content_stream(
    model='gemini-2.0-flash-001', contents='Tell me a story in 300 words.')
):
    print(chunk.text, end='')
```

Streaming for image content

If your image is stored in Google Cloud Storage, you can use the `from_uri` class method to create a `Part` object.

```
from google.genai import types

for chunk in client.models.generate_content_stream(
    model='gemini-2.0-flash-001',
    contents=[
        'What is this image about?',
        types.Part.from_uri(
            file_uri='gs://generativeai-downloads/images/scones.jpg',
            mime_type='image/jpeg',
        ),
    ],
):
    print(chunk.text, end='')
```

If your image is stored in your local file system, you can read it in as bytes data and use the `from_bytes` class method to create a `Part` object.

```
from google.genai import types

YOUR_IMAGE_PATH = 'your_image_path'
YOUR_IMAGE_MIME_TYPE = 'your_image_mime_type'
with open(YOUR_IMAGE_PATH, 'rb') as f:
    image_bytes = f.read()

for chunk in client.models.generate_content_stream(
    model='gemini-2.0-flash-001',
    contents=[
        'What is this image about?',
        types.Part.from_bytes(data=image_bytes, mime_type=YOUR_IMAGE_MIME_TYPE),
    ],
)
    print(chunk.text, end='')
```

Skip to content <.text, end='')

Generate Content (Asynchronous Non-Streaming)

`client.aio` exposes all the analogous async methods that are available on `client`. Note that it applies to all the modules.

For example, `client.aio.models.generate_content` is the `async` version of `client.models.generate_content`

```
response = await client.aio.models.generate_content(  
    model='gemini-2.0-flash-001', contents='Tell me a story in 300 words.'  
)  
  
print(response.text)
```

Generate Content (Asynchronous Streaming)

```
async for chunk in await client.aio.models.generate_content_stream(  
    model='gemini-2.0-flash-001', contents='Tell me a story in 300 words.'  
):  
    print(chunk.text, end='')
```

Count Tokens and Compute Tokens

```
response = client.models.count_tokens(  
    model='gemini-2.0-flash-001',  
    contents='why is the sky blue?',  
)  
print(response)
```

Compute Tokens

Compute tokens is only supported in Vertex AI.

```
response = client.models.compute_tokens(  
    model='gemini-2.0-flash-001',  
    contents='why is the sky blue?',  
)  
print(response)
```

[Skip to content](#)

Count Tokens (Asynchronous)

```
response = await client.aio.models.count_tokens(  
    model='gemini-2.0-flash-001',  
    contents='why is the sky blue?',  
)  
print(response)
```

Embed Content

```
response = client.models.embed_content(  
    model='text-embedding-004',  
    contents='why is the sky blue?',  
)  
print(response)
```

```
from google.genai import types  
  
# multiple contents with config  
response = client.models.embed_content(  
    model='text-embedding-004',  
    contents=['why is the sky blue?', 'What is your age?'],  
    config=types.EmbedContentConfig(output_dimensionality=10),  
)  
  
print(response)
```

Imagen

Generate Image

Support for generate image in Gemini Developer API is behind an allowlist

[Skip to content](#)

```
from google.genai import types

# Generate Image
response1 = client.models.generate_images(
    model='imagen-3.0-generate-002',
    prompt='An umbrella in the foreground, and a rainy night sky in the background',
    config=types.GenerateImagesConfig(
        number_of_images=1,
        include_rai_reason=True,
        output_mime_type='image/jpeg',
    ),
)
response1.generated_images[0].image.show()
```

Upscale image is only supported in Vertex AI.

```
from google.genai import types

# Upscale the generated image from above
response2 = client.models.upscale_image(
    model='imagen-3.0-generate-002',
    image=response1.generated_images[0].image,
    upscale_factor='x2',
    config=types.UpscaleImageConfig(
        include_rai_reason=True,
        output_mime_type='image/jpeg',
    ),
)
response2.generated_images[0].image.show()
```

Edit Image

Edit image uses a separate model from generate and upscale.

Edit image is only supported in Vertex AI.

[Skip to content](#)

```
# Edit the generated image from above
from google.genai import types
from google.genai.types import RawReferenceImage, MaskReferenceImage

raw_ref_image = RawReferenceImage(
    reference_id=1,
    reference_image=response1.generated_images[0].image,
)

# Model computes a mask of the background
mask_ref_image = MaskReferenceImage(
    reference_id=2,
    config=types.MaskReferenceConfig(
        mask_mode='MASK_MODE_BACKGROUND',
        mask_dilation=0,
    ),
)

response3 = client.models.edit_image(
    model='imagen-3.0-capability-001',
    prompt='Sunlight and clear sky',
    reference_images=[raw_ref_image, mask_ref_image],
    config=types>EditImageConfig(
        edit_mode='EDIT_MODE_INPAINT_INSERTION',
        number_of_images=1,
        include_rai_reason=True,
        output_mime_type='image/jpeg',
    ),
)
response3.generated_images[0].image.show()
```

Veo

Generate Videos

Support for generate videos in Vertex and Gemini Developer API is behind an allowlist

[Skip to content](#)

```

from google.genai import types

# Create operation
operation = client.models.generate_videos(
    model='veo-2.0-generate-001',
    prompt='A neon hologram of a cat driving at top speed',
    config=types.GenerateVideosConfig(
        number_of_videos=1,
        fps=24,
        duration_seconds=5,
        enhance_prompt=True,
    ),
)

# Poll operation
while not operation.done:
    time.sleep(20)
    operation = client.operations.get(operation)

video = operation.result.generated_videos[0].video
video.show()

```

Chats

Create a chat session to start a multi-turn conversations with the model. Then, use `chat.send_message` function multiple times within the same chat session so that it can reflect on its previous responses (i.e., engage in an ongoing conversation). See the ‘Create a client’ section above to initialize a client.

Send Message (Synchronous Non-Streaming)

```

chat = client.chats.create(model='gemini-2.0-flash-001')
response = chat.send_message('tell me a story')
print(response.text)
response = chat.send_message('summarize the story you told me in 1 sentence')
print(response.text)

```

Send Message (Synchronous Streaming)

Skip to content `chats.create(model='gemini-2.0-flash-001')`
`chat.send_message_stream('tell me a story'):`
`print(chunk.text, end='') # end='' is optional, for demo purposes.`

Send Message (Asynchronous Non-Streaming)

```
chat = client.aio.chats.create(model='gemini-2.0-flash-001')
response = await chat.send_message('tell me a story')
print(response.text)
```

Send Message (Asynchronous Streaming)

```
chat = client.aio.chats.create(model='gemini-2.0-flash-001')
async for chunk in await chat.send_message_stream('tell me a story'):
    print(chunk.text, end='') # end='' is optional, for demo purposes.
```

Files

Files are only supported in Gemini Developer API. See the 'Create a client' section above to initialize a client.

```
gsutil cp gs://cloud-samples-data/generative-ai/pdf/2312.11805v3.pdf .
gsutil cp gs://cloud-samples-data/generative-ai/pdf/2403.05530.pdf .
```

Upload

```
file1 = client.files.upload(file='2312.11805v3.pdf')
file2 = client.files.upload(file='2403.05530.pdf')

print(file1)
print(file2)
```

Get

```
file1 = client.files.upload(file='2312.11805v3.pdf')
file_info = client.files.get(name=file1.name)
```

Delete

Skip to content [.files.upload\(file='2312.11805v3.pdf'\)](#)

```
client.files.delete(name=file3.name)
```

Caches

`client.caches` contains the control plane APIs for cached content.

See the 'Create a client' section above to initialize a client.

Create

```
from google.genai import types

if client.vertexai:
    file_uris = [
        'gs://cloud-samples-data/generative-ai/pdf/2312.11805v3.pdf',
        'gs://cloud-samples-data/generative-ai/pdf/2403.05530.pdf',
    ]
else:
    file_uris = [file1.uri, file2.uri]

cached_content = client.caches.create(
    model='gemini-2.0-flash-001',
    config=types.CreateCachedContentConfig(
        contents=[
            types.Content(
                role='user',
                parts=[
                    types.Part.from_uri(
                        file_uri=file_uris[0], mime_type='application/pdf'
                    ),
                    types.Part.from_uri(
                        file_uri=file_uris[1],
                        mime_type='application/pdf',
                    ),
                ],
            ),
            system_instruction='What is the sum of the two pdfs?',
            display_name='test cache',
            ttl='3600s',
        ),
    )
)
```

Get

Skip to content

```
= client.caches.get(name=cached_content.name)
```

Generate Content with Caches

```
from google.genai import types

response = client.models.generate_content(
    model='gemini-2.0-flash-001',
    contents='Summarize the pdfs',
    config=types.GenerateContentConfig(
        cached_content=cached_content.name,
    ),
)
print(response.text)
```

Tunings

`client.tunings` contains tuning job APIs and supports supervised fine tuning through `tune`. Only supported in Vertex AI. See the ‘Create a client’ section above to initialize a client.

Tune

- Vertex AI supports tuning from GCS source

```
from google.genai import types

model = 'gemini-2.0-flash-001'
training_dataset = types.TuningDataset(
    gcs_uri='gs://cloud-samples-data/ai-platform/generative_ai/gemini-1_5/text/sft_train_0')

```

```
from google.genai import types

tuning_job = client.tunings.tune(
    base_model=model,
    training_dataset=training_dataset,
    config=types.CreateTuningJobConfig(
        epoch_count=1, tuned_model_display_name='test_dataset_examples model'
    ),
)
print(tuning_job)
```

[Skip to content](#)

Get Tuning Job

```
tuning_job = client.tunings.get(name=tuning_job.name)
print(tuning_job)
```

```
import time

completed_states = set(
    [
        'JOB_STATE_SUCCEEDED',
        'JOB_STATE_FAILED',
        'JOB_STATE_CANCELLED',
    ]
)

while tuning_job.state not in completed_states:
    print(tuning_job.state)
    tuning_job = client.tunings.get(name=tuning_job.name)
    time.sleep(10)
```

Use Tuned Model

```
response = client.models.generate_content(
    model=tuning_job.tuned_model.endpoint,
    contents='why is the sky blue?',
)
print(response.text)
```

Get Tuned Model

```
tuned_model = client.models.get(model=tuning_job.tuned_model.model)
print(tuned_model)
```

[Skip to content](#)

Update Tuned Model

```
from google.genai import types

tuned_model = client.models.update(
    model=tuning_job.tuned_model.model,
    config=types.UpdateModelConfig(
        display_name='my tuned model', description='my tuned model description'
    ),
)
print(tuned_model)
```

List Tuned Models

To retrieve base models, see: [List Base Models](#)

```
for model in client.models.list(config={'page_size': 10, 'query_base': False}):
    print(model)
```

```
pager = client.models.list(config={'page_size': 10, 'query_base': False})
print(pager.page_size)
print(pager[0])
pager.next_page()
print(pager[0])
```

List Tuned Models (Asynchronous)

```
async for job in await client.aio.models.list(config={'page_size': 10, 'query_base': False}):
    print(job)
```

```
async_pager = await client.aio.models.list(config={'page_size': 10, 'query_base': False})
print(async_pager.page_size)
print(async_pager[0])
await async_pager.next_page()
print(async_pager[0])
```

[Skip to content](#)

Update Tuned Model

```
model = pager[0]

model = client.models.update(
    model=model.name,
    config=types.UpdateModelConfig(
        display_name='my tuned model', description='my tuned model description'
    ),
)

print(model)
```

List Tuning Jobs

```
for job in client.tunings.list(config={'page_size': 10}):
    print(job)
```

```
pager = client.tunings.list(config={'page_size': 10})
print(pager.page_size)
print(pager[0])
pager.next_page()
print(pager[0])
```

List Tuning Jobs (Asynchronous):

```
async for job in await client.aio.tunings.list(config={'page_size': 10}):
    print(job)
```

```
async_pager = await client.aio.tunings.list(config={'page_size': 10})
print(async_pager.page_size)
print(async_pager[0])
await async_pager.next_page()
print(async_pager[0])
```

Batch Prediction

Create a batch job. See the 'Create a client' section above to initialize a client.

[Skip to content](#)

Create

Vertex AI client support using a BigQuery table or a GCS file as the source.

```
# Specify model and source file only, destination and job display name will be auto-populated
job = client.batches.create(
    model='gemini-2.0-flash-001',
    src='bq://my-project.my-dataset.my-table', # or gcs://my-bucket/my-file.jsonl'
)
```

Gemini Developer API client:

```
# Create a batch job with inlined requests
batch_job = client.batches.create(
    model="gemini-2.0-flash",
    src=[{
        "contents": [
            {
                "parts": [
                    {
                        "text": "Hello!",
                    }
                ],
                "role": "user",
            },
            {
                "config": {"response_modalities": ["text"]},
            }
        ]
    })
)
```

In order to create a batch job with a file. Need to upload a jsonl file. For example myrequests.jsonl:

```
# Get a job by name
job = client.batches.get(name=job.name)

job.state
```

Skip to content

```
completed_states = set(
    [
        'JOB_STATE_SUCCEEDED',
        'JOB_STATE_FAILED',
        'JOB_STATE_CANCELLED',
        'JOB_STATE_PAUSED',
    ]
)

while job.state not in completed_states:
    print(job.state)
    job = client.batches.get(name=job.name)
    time.sleep(30)

job
```

List

```
from google.genai import types

for job in client.batches.list(config=types.ListBatchJobsConfig(page_size=10)):
    print(job)
```

List Batch Jobs with Pager

```
from google.genai import types

pager = client.batches.list(config=types.ListBatchJobsConfig(page_size=10))
print(pager.page_size)
print(pager[0])
pager.next_page()
print(pager[0])
```

List Batch Jobs (Asynchronous)

```
from google.genai import types

async for job in await client.aio.batches.list(
    config=types.ListBatchJobsConfig(page_size=10)
):
    print(job)
```

Skip to content

List Batch Jobs with Pager (Asynchronous)

```
from google.genai import types

async_page = await client.aio.batches.list(
    config=types.ListBatchJobsConfig(page_size=10)
)
print(async_page.page_size)
print(async_page[0])
await async_page.next_page()
print(async_page[0])
```

Delete

```
# Delete the job resource
delete_job = client.batches.delete(name=job.name)

delete_job
```

Error Handling

To handle errors raised by the model, the SDK provides this APIError class.

```
try:
    client.models.generate_content(
        model="invalid-model-name",
        contents="What is your name?",
    )
except errors.APIError as e:
    print(e.code) # 404
    print(e.message)
```

[Skip to content](#)

Reference

- Submodules
- [genai.client module](#)
 - `AsyncClient`
 - `AsyncClient.auth_tokens`
 - `AsyncClient.batches`
 - `AsyncClient.caches`
 - `AsyncClient.chats`
 - `AsyncClient.files`
 - `AsyncClient.live`
 - `AsyncClient.models`
 - `AsyncClient.operations`
 - `AsyncClient.tunings`
 - `Client`
 - `Client.api_key`
 - `Client.vertexai`
 - `Client.credentials`
 - `Client.project`
 - `Client.location`
 - `Client.debug_config`
 - `Client.http_options`
 - `Client.aio`
 - `Client.auth_tokens`
 - `Client.batches`
 - `Client.caches`
 - `Client.chats`
 - `Client.files`
 - `Client.models`
 - `Client.operations`
 - `Client.tunings`
 - `Client.vertexai`
 - `DebugConfig`
 - `DebugConfig.client_mode`
 - `DebugConfig.replay_id`
 - `DebugConfig.replays_directory`

[Skip to content](#)

- [genai.batches module](#)

- [AsyncBatches](#)

- [AsyncBatches.cancel\(\)](#)
 - [AsyncBatches.create\(\)](#)
 - [AsyncBatches.delete\(\)](#)
 - [AsyncBatches.get\(\)](#)
 - [AsyncBatches.list\(\)](#)

- [Batches](#)

- [Batches.cancel\(\)](#)
 - [Batches.create\(\)](#)
 - [Batches.delete\(\)](#)
 - [Batches.get\(\)](#)
 - [Batches.list\(\)](#)

- [genai.caches module](#)

- [AsyncCaches](#)

- [AsyncCaches.create\(\)](#)
 - [AsyncCaches.delete\(\)](#)
 - [AsyncCaches.get\(\)](#)
 - [AsyncCaches.list\(\)](#)
 - [AsyncCaches.update\(\)](#)

- [Caches](#)

- [Caches.create\(\)](#)
 - [Caches.delete\(\)](#)
 - [Caches.get\(\)](#)
 - [Caches.list\(\)](#)
 - [Caches.update\(\)](#)

- [genai.chats module](#)

- [AsyncChat](#)

- [AsyncChat.send_message\(\)](#)
 - [AsyncChat.send_message_stream\(\)](#)

- [AsyncChats](#)

- [AsyncChats.create\(\)](#)

- [Chat](#)

- [Chat.send_message\(\)](#)
 - [Chat.send_message_stream\(\)](#)

- [Chats](#)

- [Chats.create\(\)](#)

[Skip to content](#)

- `genai.files` module

- `AsyncFiles`
 - `AsyncFiles.delete()`
 - `AsyncFiles.download()`
 - `AsyncFiles.get()`
 - `AsyncFiles.list()`
 - `AsyncFiles.upload()`

- `Files`
 - `Files.delete()`
 - `Files.download()`
 - `Files.get()`
 - `Files.list()`
 - `Files.upload()`

- `genai.live` module

- `AsyncLive`
 - `AsyncLive.connect()`
 - `AsyncLive.music`
- `AsyncSession`
 - `AsyncSession.close()`
 - `AsyncSession.receive()`
 - `AsyncSession.send()`
 - `AsyncSession.send_client_content()`
 - `AsyncSession.send_realtime_input()`
 - `AsyncSession.send_tool_response()`
 - `AsyncSession.start_stream()`

[Skip to content](#)

- genai.models module

- AsyncModels

- `AsyncModels.compute_tokens()`
 - `AsyncModels.count_tokens()`
 - `AsyncModels.delete()`
 - `AsyncModels.edit_image()`
 - `AsyncModels.embed_content()`
 - `AsyncModels.generate_content()`
 - `AsyncModels.generate_content_stream()`
 - `AsyncModels.generate_images()`
 - `AsyncModels.generate_videos()`
 - `AsyncModels.get()`
 - `AsyncModels.list()`
 - `AsyncModels.update()`
 - `AsyncModels.upscale_image()`

- Models

- `Models.compute_tokens()`
 - `Models.count_tokens()`
 - `Models.delete()`
 - `Models.edit_image()`
 - `Models.embed_content()`
 - `Models.generate_content()`
 - `Models.generate_content_stream()`
 - `Models.generate_images()`
 - `Models.generate_videos()`
 - `Models.get()`
 - `Models.list()`
 - `Models.update()`
 - `Models.upscale_image()`

- genai.tokens module

- AsyncTokens

- `AsyncTokens.create()`

- Tokens

- `Tokens.create()`

[Skip to content](#)

- genai.tunings module

- AsyncTunings
 - `AsyncTunings.get()`
 - `AsyncTunings.list()`
 - `AsyncTunings.tune()`
- Tunings
 - `Tunings.get()`
 - `Tunings.list()`
 - `Tunings.tune()`

[Skip to content](#)

- `genai.types` module
 - `ActivityEnd`
 - `ActivityEndDict`
 - `ActivityHandling`
 - `ActivityHandling.ACTIVITY_HANDLING_UNSPECIFIED`
 - `ActivityHandling.NO INTERRUPTION`
 - `ActivityHandling.START_OF_ACTIVITY_INTERRUPTS`
 - `ActivityStart`
 - `ActivityStartDict`
 - `AdapterSize`
 - `AdapterSize.ADAPTER_SIZE_EIGHT`
 - `AdapterSize.ADAPTER_SIZE_FOUR`
 - `AdapterSize.ADAPTER_SIZE_ONE`
 - `AdapterSize.ADAPTER_SIZE_SIXTEEN`
 - `AdapterSize.ADAPTER_SIZE_THIRTY_TWO`
 - `AdapterSize.ADAPTER_SIZE_TWO`
 - `AdapterSize.ADAPTER_SIZE_UNSPECIFIED`
 - `ApiAuth`
 - `ApiAuth.api_key_config`
 - `ApiAuthApiKeyConfig`
 - `ApiAuthApiKeyConfig.api_key_secret_version`
 - `ApiAuthApiKeyConfig.api_key_string`
 - `ApiAuthApiKeyConfigDict`
 - `ApiAuthApiKeyConfigDict.api_key_secret_version`
 - `ApiAuthApiKeyConfigDict.api_key_string`
 - `ApiAuthDict`
 - `ApiAuthDict.api_key_config`
 - `ApiKeyConfig`
 - `ApiKeyConfig.api_key_string`
 - `ApiKeyConfigDict`
 - `ApiKeyConfigDict.api_key_string`
 - `ApiSpec`
 - `ApiSpec.API_SPEC_UNSPECIFIED`
 - `ApiSpec.ELASTIC_SEARCH`
 - `ApiSpec.SIMPLE_SEARCH`
 - `AudioChunk`
 - `AudioChunk.data`

Skip to content `nk.mime_type`
`nk.source_metadata`

- `AudioChunkDict`
 - `AudioChunkDict.data`
 - `AudioChunkDict.mime_type`
 - `AudioChunkDict.source_metadata`
- `AudioTranscriptionConfig`
- `AudioTranscriptionConfigDict`
- `AuthConfig`
 - `AuthConfig.api_key_config`
 - `AuthConfig.auth_type`
 - `AuthConfig.google_service_account_config`
 - `AuthConfig.http_basic_auth_config`
 - `AuthConfig.oauth_config`
 - `AuthConfig.oidc_config`
- `AuthConfigDict`
 - `AuthConfigDict.api_key_config`
 - `AuthConfigDict.auth_type`
 - `AuthConfigDict.google_service_account_config`
 - `AuthConfigDict.http_basic_auth_config`
 - `AuthConfigDict.oauth_config`
 - `AuthConfigDict.oidc_config`
- `AuthConfigGoogleServiceAccountConfig`
 - `AuthConfigGoogleServiceAccountConfig.service_account`
- `AuthConfigGoogleServiceAccountConfigDict`
 - `AuthConfigGoogleServiceAccountConfigDict.service_account`
- `AuthConfigHttpBasicAuthConfig`
 - `AuthConfigHttpBasicAuthConfig.credential_secret`
- `AuthConfigHttpBasicAuthConfigDict`
 - `AuthConfigHttpBasicAuthConfigDict.credential_secret`
- `AuthConfigOauthConfig`
 - `AuthConfigOauthConfig.access_token`
 - `AuthConfigOauthConfig.service_account`
- `AuthConfigOauthConfigDict`
 - `AuthConfigOauthConfigDict.access_token`
 - `AuthConfigOauthConfigDict.service_account`
- `AuthConfigOidcConfig`
 - `AuthConfigOidcConfig.id_token`
 - `AuthConfigOidcConfig.service_account`

Skip to content `idcConfigDict`
`igOidcConfigDict.id_token`
▪ `AuthConfigOidcConfigDict.service_account`

- AuthToken
 - AuthToken.name
- AuthTokenDict
 - AuthTokenDict.name
- AuthType
 - AuthType.API_KEY_AUTH
 - AuthType.AUTH_TYPE_UNSPECIFIED
 - AuthType.GOOGLE_SERVICE_ACCOUNT_AUTH
 - AuthType.HTTP_BASIC_AUTH
 - AuthType.NO_AUTH
 - AuthType.OAUTH
 - AuthType.OIDC_AUTH
- AutomaticActivityDetection
 - AutomaticActivityDetection.disabled
 - AutomaticActivityDetection.end_of_speech_sensitivity
 - AutomaticActivityDetection.prefix_padding_ms
 - AutomaticActivityDetection.silence_duration_ms
 - AutomaticActivityDetection.start_of_speech_sensitivity
- AutomaticActivityDetectionDict
 - AutomaticActivityDetectionDict.disabled
 - AutomaticActivityDetectionDict.end_of_speech_sensitivity
 - AutomaticActivityDetectionDict.prefix_padding_ms
 - AutomaticActivityDetectionDict.silence_duration_ms
 - AutomaticActivityDetectionDict.start_of_speech_sensitivity
- AutomaticFunctionCallingConfig
 - AutomaticFunctionCallingConfig.disable
 - AutomaticFunctionCallingConfig.ignore_call_history
 - AutomaticFunctionCallingConfig.maximum_remote_calls
- AutomaticFunctionCallingConfigDict
 - AutomaticFunctionCallingConfigDict.disable
 - AutomaticFunctionCallingConfigDict.ignore_call_history
 - AutomaticFunctionCallingConfigDict.maximum_remote_calls

[Skip to content](#)

- BatchJob
 - BatchJob.create_time
 - BatchJob.dest
 - BatchJob.display_name
 - BatchJob.end_time
 - BatchJob.error
 - BatchJob.model
 - BatchJob.name
 - BatchJob.src
 - BatchJob.start_time
 - BatchJob.state
 - BatchJob.update_time
- BatchJobDestination
 - BatchJobDestination.bigquery_uri
 - BatchJobDestination.file_name
 - BatchJobDestination.format
 - BatchJobDestination.gcs_uri
 - BatchJobDestination.inlined_responses
- BatchJobDestinationDict
 - BatchJobDestinationDict.bigquery_uri
 - BatchJobDestinationDict.file_name
 - BatchJobDestinationDict.format
 - BatchJobDestinationDict.gcs_uri
 - BatchJobDestinationDict.inlined_responses
- BatchJobDict
 - BatchJobDict.create_time
 - BatchJobDict.dest
 - BatchJobDict.display_name
 - BatchJobDict.end_time
 - BatchJobDict.error
 - BatchJobDict.model
 - BatchJobDict.name
 - BatchJobDict.src
 - BatchJobDict.start_time
 - BatchJobDict.state
 - BatchJobDict.update_time

[Skip to content](#)

- BatchJobSource
 - BatchJobSource.bigquery_uri
 - BatchJobSource.file_name
 - BatchJobSource.format
 - BatchJobSource.gcs_uri
 - BatchJobSource.inlined_requests
- BatchJobSourceDict
 - BatchJobSourceDict.bigquery_uri
 - BatchJobSourceDict.file_name
 - BatchJobSourceDict.format
 - BatchJobSourceDict.gcs_uri
 - BatchJobSourceDict.inlined_requests
- Behavior
 - Behavior.BLOCKING
 - Behavior.NON_BLOCKING
 - Behavior.UNSPECIFIED
- Blob
 - Blob.data
 - Blob.display_name
 - Blob.mime_type
- BlobDict
 - BlobDict.data
 - BlobDict.display_name
 - BlobDict.mime_type
- BlockedReason
 - BlockedReason.BLOCKED_REASON_UNSPECIFIED
 - BlockedReason.BLOCKLIST
 - BlockedReason.IMAGE_SAFETY
 - BlockedReason.OTHER
 - BlockedReason.PROHIBITED_CONTENT
 - BlockedReason.SAFETY
- CachedContent
 - CachedContent.create_time
 - CachedContent.display_name
 - CachedContent.expire_time
 - CachedContent.model
 - CachedContent.name

Skip to content `\intent.update_time`
 `\intent.usage_metadata`

- `CachedContentDict`
 - `CachedContentDict.create_time`
 - `CachedContentDict.display_name`
 - `CachedContentDict.expire_time`
 - `CachedContentDict.model`
 - `CachedContentDict.name`
 - `CachedContentDict.update_time`
 - `CachedContentDict.usage_metadata`
- `CachedContentUsageMetadata`
 - `CachedContentUsageMetadata.audio_duration_seconds`
 - `CachedContentUsageMetadata.image_count`
 - `CachedContentUsageMetadata.text_count`
 - `CachedContentUsageMetadata.total_token_count`
 - `CachedContentUsageMetadata.video_duration_seconds`
- `CachedContentUsageMetadataDict`
 - `CachedContentUsageMetadataDict.audio_duration_seconds`
 - `CachedContentUsageMetadataDict.image_count`
 - `CachedContentUsageMetadataDict.text_count`
 - `CachedContentUsageMetadataDict.total_token_count`
 - `CachedContentUsageMetadataDict.video_duration_seconds`
- `CancelBatchJobConfig`
 - `CancelBatchJobConfig.http_options`
- `CancelBatchJobConfigDict`
 - `CancelBatchJobConfigDict.http_options`
- `Candidate`
 - `Candidate.avg_logprobs`
 - `Candidate.citation_metadata`
 - `Candidate.content`
 - `Candidate.finish_message`
 - `Candidate.finish_reason`
 - `Candidate.grounding_metadata`
 - `Candidate.index`
 - `Candidate.logprobs_result`
 - `Candidate.safety_ratings`
 - `Candidate.token_count`
 - `Candidate.url_context_metadata`

[Skip to content](#)

- CandidateDict
 - CandidateDict.avg_logprobs
 - CandidateDict.citation_metadata
 - CandidateDict.content
 - CandidateDict.finish_message
 - CandidateDict.finish_reason
 - CandidateDict.grounding_metadata
 - CandidateDict.index
 - CandidateDict.logprobs_result
 - CandidateDict.safety_ratings
 - CandidateDict.token_count
 - CandidateDict.url_context_metadata
- Checkpoint
 - Checkpoint.checkpoint_id
 - Checkpoint.epoch
 - Checkpoint.step
- CheckpointDict
 - CheckpointDict.checkpoint_id
 - CheckpointDict.epoch
 - CheckpointDict.step
- Citation
 - Citation.end_index
 - Citation.license
 - Citation.publication_date
 - Citation.start_index
 - Citation.title
 - Citation.uri
- CitationDict
 - CitationDict.end_index
 - CitationDict.license
 - CitationDict.publication_date
 - CitationDict.start_index
 - CitationDict.title
 - CitationDict.uri
- CitationMetadata
 - CitationMetadata.citations
- CitationMetadataDict

Skip to content [MetadataDict.citations](#)

- `CodeExecutionResult`
 - `CodeExecutionResult.outcome`
 - `CodeExecutionResult.output`
- `CodeExecutionResultDict`
 - `CodeExecutionResultDict.outcome`
 - `CodeExecutionResultDict.output`
- `ComputeTokensConfig`
 - `ComputeTokensConfig.http_options`
- `ComputeTokensConfigDict`
 - `ComputeTokensConfigDict.http_options`
- `ComputeTokensResponse`
 - `ComputeTokensResponse.tokens_info`
- `ComputeTokensResponseDict`
 - `ComputeTokensResponseDict.tokens_info`
- `Content`
 - `Content.parts`
 - `Content.role`
- `ContentDict`
 - `ContentDict.parts`
 - `ContentDict.role`
- `ContentEmbedding`
 - `ContentEmbedding.statistics`
 - `ContentEmbedding.values`
- `ContentEmbeddingDict`
 - `ContentEmbeddingDict.statistics`
- `ContentEmbeddingStatistics`
 - `ContentEmbeddingStatistics.token_count`
 - `ContentEmbeddingStatistics.truncated`
- `ContentEmbeddingStatisticsDict`
 - `ContentEmbeddingStatisticsDict.token_count`
 - `ContentEmbeddingStatisticsDict.truncated`
- `ContextWindowCompressionConfig`
 - `ContextWindowCompressionConfig.sliding_window`
 - `ContextWindowCompressionConfig.trigger_tokens`
- `ContextWindowCompressionConfigDict`
 - `ContextWindowCompressionConfigDict.sliding_window`
 - `ContextWindowCompressionConfigDict.trigger_tokens`

Skip to content `referenceConfig`
 `referenceConfig.control_type`

- `ControlReferenceConfig.enable_control_image_computation`

- ControlReferenceConfigDict
 - ControlReferenceConfigDict.control_type
 - ControlReferenceConfigDict.enable_control_image_computation
- ControlReferenceImage
 - ControlReferenceImage.config
 - ControlReferenceImage.control_image_config
 - ControlReferenceImage.reference_id
 - ControlReferenceImage.reference_image
 - ControlReferenceImage.reference_type
- ControlReferenceImageDict
 - ControlReferenceImageDict.config
 - ControlReferenceImageDict.reference_id
 - ControlReferenceImageDict.reference_image
 - ControlReferenceImageDict.reference_type
- ControlReferenceType
 - ControlReferenceType.CONTROL_TYPE_CANNY
 - ControlReferenceType.CONTROL_TYPE_DEFAULT
 - ControlReferenceType.CONTROL_TYPE_FACE_MESH
 - ControlReferenceType.CONTROL_TYPE_SCRIBBLE
- CountTokensConfig
 - CountTokensConfig.generation_config
 - CountTokensConfig.http_options
 - CountTokensConfig.system_instruction
 - CountTokensConfig.tools
- CountTokensConfigDict
 - CountTokensConfigDict.generation_config
 - CountTokensConfigDict.http_options
 - CountTokensConfigDict.system_instruction
 - CountTokensConfigDict.tools
- CountTokensResponse
 - CountTokensResponse.cached_content_token_count
 - CountTokensResponse.total_tokens
- CountTokensResponseDict
 - CountTokensResponseDict.cached_content_token_count
 - CountTokensResponseDict.total_tokens

[Skip to content](#)

- `CreateAuthTokenConfig`
 - `CreateAuthTokenConfig.expire_time`
 - `CreateAuthTokenConfig.http_options`
 - `CreateAuthTokenConfig.live_connect_constraints`
 - `CreateAuthTokenConfig.lock_additional_fields`
 - `CreateAuthTokenConfig.new_session_expire_time`
 - `CreateAuthTokenConfig.uses`
- `CreateAuthTokenConfigDict`
 - `CreateAuthTokenConfigDict.expire_time`
 - `CreateAuthTokenConfigDict.http_options`
 - `CreateAuthTokenConfigDict.live_connect_constraints`
 - `CreateAuthTokenConfigDict.lock_additional_fields`
 - `CreateAuthTokenConfigDict.new_session_expire_time`
 - `CreateAuthTokenConfigDict.uses`
- `CreateAuthTokenParameters`
 - `CreateAuthTokenParameters.config`
- `CreateAuthTokenParametersDict`
 - `CreateAuthTokenParametersDict.config`
- `CreateBatchJobConfig`
 - `CreateBatchJobConfig.dest`
 - `CreateBatchJobConfig.display_name`
 - `CreateBatchJobConfig.http_options`
- `CreateBatchJobConfigDict`
 - `CreateBatchJobConfigDict.dest`
 - `CreateBatchJobConfigDict.display_name`
 - `CreateBatchJobConfigDict.http_options`
- `CreateCachedContentConfig`
 - `CreateCachedContentConfig.contents`
 - `CreateCachedContentConfig.display_name`
 - `CreateCachedContentConfig.expire_time`
 - `CreateCachedContentConfig.http_options`
 - `CreateCachedContentConfig.kms_key_name`
 - `CreateCachedContentConfig.system_instruction`
 - `CreateCachedContentConfig.tool_config`
 - `CreateCachedContentConfig.tools`
 - `CreateCachedContentConfig.ttl`

[Skip to content](#)

- `CreateCachedContentConfigDict`
 - `CreateCachedContentConfigDict.contents`
 - `CreateCachedContentConfigDict.display_name`
 - `CreateCachedContentConfigDict.expire_time`
 - `CreateCachedContentConfigDict.http_options`
 - `CreateCachedContentConfigDict.kms_key_name`
 - `CreateCachedContentConfigDict.system_instruction`
 - `CreateCachedContentConfigDict.tool_config`
 - `CreateCachedContentConfigDict.tools`
 - `CreateCachedContentConfigDict.ttl`
- `CreateFileConfig`
 - `CreateFileConfig.http_options`
 - `CreateFileConfig.should_return_http_response`
- `CreateFileConfigDict`
 - `CreateFileConfigDict.http_options`
 - `CreateFileConfigDict.should_return_http_response`
- `CreateFileResponse`
 - `CreateFileResponse.sdk_http_response`
- `CreateFileResponseDict`
 - `CreateFileResponseDict.sdk_http_response`
- `CreateTuningJobConfig`
 - `CreateTuningJobConfig.adapter_size`
 - `CreateTuningJobConfig.batch_size`
 - `CreateTuningJobConfig.description`
 - `CreateTuningJobConfig.epoch_count`
 - `CreateTuningJobConfig.export_last_checkpoint_only`
 - `CreateTuningJobConfig.http_options`
 - `CreateTuningJobConfig.learning_rate`
 - `CreateTuningJobConfig.learning_rate_multiplier`
 - `CreateTuningJobConfig.tuned_model_display_name`
 - `CreateTuningJobConfig.validation_dataset`

[Skip to content](#)

- `CreateTuningJobConfigDict`
 - `CreateTuningJobConfigDict.adapter_size`
 - `CreateTuningJobConfigDict.batch_size`
 - `CreateTuningJobConfigDict.description`
 - `CreateTuningJobConfigDict.epoch_count`
 - `CreateTuningJobConfigDict.export_last_checkpoint_only`
 - `CreateTuningJobConfigDict.http_options`
 - `CreateTuningJobConfigDict.learning_rate`
 - `CreateTuningJobConfigDict.learning_rate_multiplier`
 - `CreateTuningJobConfigDict.tuned_model_display_name`
 - `CreateTuningJobConfigDict.validation_dataset`
- `DatasetDistribution`
 - `DatasetDistribution.buckets`
 - `DatasetDistribution.max`
 - `DatasetDistribution.mean`
 - `DatasetDistribution.median`
 - `DatasetDistribution.min`
 - `DatasetDistribution.p5`
 - `DatasetDistribution.p95`
 - `DatasetDistribution.sum`
- `DatasetDistributionDict`
 - `DatasetDistributionDict.buckets`
 - `DatasetDistributionDict.max`
 - `DatasetDistributionDict.mean`
 - `DatasetDistributionDict.median`
 - `DatasetDistributionDict.min`
 - `DatasetDistributionDict.p5`
 - `DatasetDistributionDict.p95`
 - `DatasetDistributionDict.sum`
- `DatasetDistributionDistributionBucket`
 - `DatasetDistributionDistributionBucket.count`
 - `DatasetDistributionDistributionBucket.left`
 - `DatasetDistributionDistributionBucket.right`
- `DatasetDistributionDistributionBucketDict`
 - `DatasetDistributionDistributionBucketDict.count`
 - `DatasetDistributionDistributionBucketDict.left`
 - `DatasetDistributionDistributionBucketDict.right`

[Skip to content](#)

- `DatasetStats`
 - `DatasetStats.total_billable_character_count`
 - `DatasetStats.total_tuning_character_count`
 - `DatasetStats.tuning_dataset_example_count`
 - `DatasetStats.tuning_step_count`
 - `DatasetStats.user_dataset_examples`
 - `DatasetStats.user_input_token_distribution`
 - `DatasetStats.user_message_per_example_distribution`
 - `DatasetStats.user_output_token_distribution`
- `DatasetStatsDict`
 - `DatasetStatsDict.total_billable_character_count`
 - `DatasetStatsDict.total_tuning_character_count`
 - `DatasetStatsDict.tuning_dataset_example_count`
 - `DatasetStatsDict.tuning_step_count`
 - `DatasetStatsDict.user_dataset_examples`
 - `DatasetStatsDict.user_input_token_distribution`
 - `DatasetStatsDict.user_message_per_example_distribution`
 - `DatasetStatsDict.user_output_token_distribution`
- `DeleteBatchJobConfig`
 - `DeleteBatchJobConfig.http_options`
- `DeleteBatchJobConfigDict`
 - `DeleteBatchJobConfigDict.http_options`
- `DeleteCachedContentConfig`
 - `DeleteCachedContentConfig.http_options`
- `DeleteCachedContentConfigDict`
 - `DeleteCachedContentConfigDict.http_options`
- `DeleteCachedContentResponse`
- `DeleteCachedContentResponseDict`
- `DeleteFileConfig`
 - `DeleteFileConfig.http_options`
- `DeleteFileConfigDict`
 - `DeleteFileConfigDict.http_options`
- `DeleteFileResponse`
- `DeleteFileResponseDict`
- `DeleteModelConfig`
 - `DeleteModelConfig.http_options`
- `DeleteModelConfigDict`
 - Skip to content `ModelConfigDict.http_options`
 - `.Response`
- `DeleteModelResponseDict`

- DeleteResourceJob
 - DeleteResourceJob.done
 - DeleteResourceJob.error
 - DeleteResourceJob.name
- DeleteResourceJobDict
 - DeleteResourceJobDict.done
 - DeleteResourceJobDict.error
 - DeleteResourceJobDict.name
- DistillationDataStats
 - DistillationDataStats.training_dataset_stats
- DistillationDataStatsDict
 - DistillationDataStatsDict.training_dataset_stats
- DistillationHyperParameters
 - DistillationHyperParameters.adapter_size
 - DistillationHyperParameters.epoch_count
 - DistillationHyperParameters.learning_rate_multiplier
- DistillationHyperParametersDict
 - DistillationHyperParametersDict.adapter_size
 - DistillationHyperParametersDict.epoch_count
 - DistillationHyperParametersDict.learning_rate_multiplier
- DistillationSpec
 - DistillationSpec.base_teacher_model
 - DistillationSpec.hyper_parameters
 - DistillationSpec.pipeline_root_directory
 - DistillationSpec.student_model
 - DistillationSpec.training_dataset_uri
 - DistillationSpec.tuned_teacher_model_source
 - DistillationSpec.validation_dataset_uri
- DistillationSpecDict
 - DistillationSpecDict.base_teacher_model
 - DistillationSpecDict.hyper_parameters
 - DistillationSpecDict.pipeline_root_directory
 - DistillationSpecDict.student_model
 - DistillationSpecDict.training_dataset_uri
 - DistillationSpecDict.tuned_teacher_model_source
 - DistillationSpecDict.validation_dataset_uri
- DownloadFileConfig
 - Skip to content `fileConfig.http_options`
 - fileConfigDict
 - DownloadFileConfigDict.http_options

- DynamicRetrievalConfig
 - DynamicRetrievalConfig.dynamic_threshold
 - DynamicRetrievalConfig.mode
- DynamicRetrievalConfigDict
 - DynamicRetrievalConfigDict.dynamic_threshold
 - DynamicRetrievalConfigDict.mode
- DynamicRetrievalConfigMode
 - DynamicRetrievalConfigMode.MODE_DYNAMIC
 - DynamicRetrievalConfigMode.MODE_UNSPECIFIED
- EditImageConfig
 - EditImageConfig.add_watermark
 - EditImageConfig.aspect_ratio
 - EditImageConfig.base_steps
 - EditImageConfig.edit_mode
 - EditImageConfig.guidance_scale
 - EditImageConfig.http_options
 - EditImageConfig.include_rai_reason
 - EditImageConfig.include_safety_attributes
 - EditImageConfig.language
 - EditImageConfig.negative_prompt
 - EditImageConfig.number_of_images
 - EditImageConfig.output_compression_quality
 - EditImageConfig.output_gcs_uri
 - EditImageConfig.output_mime_type
 - EditImageConfig.person_generation
 - EditImageConfig.safety_filter_level
 - EditImageConfig.seed

[Skip to content](#)

- `EditImageConfigDict`
 - `EditImageConfigDict.add_watermark`
 - `EditImageConfigDict.aspect_ratio`
 - `EditImageConfigDict.base_steps`
 - `EditImageConfigDict.edit_mode`
 - `EditImageConfigDict.guidance_scale`
 - `EditImageConfigDict.http_options`
 - `EditImageConfigDict.include_rai_reason`
 - `EditImageConfigDict.include_safety_attributes`
 - `EditImageConfigDict.language`
 - `EditImageConfigDict.negative_prompt`
 - `EditImageConfigDict.number_of_images`
 - `EditImageConfigDict.output_compression_quality`
 - `EditImageConfigDict.output_gcs_uri`
 - `EditImageConfigDict.output_mime_type`
 - `EditImageConfigDict.person_generation`
 - `EditImageConfigDict.safety_filter_level`
 - `EditImageConfigDict.seed`
- `EditImageResponse`
 - `EditImageResponse.generated_images`
- `EditImageResponseDict`
 - `EditImageResponseDict.generated_images`
- `EditMode`
 - `EditMode.EDIT_MODE_BGSWAP`
 - `EditMode.EDIT_MODE_CONTROLLED_EDITING`
 - `EditMode.EDIT_MODE_DEFAULT`
 - `EditMode.EDIT_MODE_INPAINT_INSERTION`
 - `EditMode.EDIT_MODE_INPAINT_REMOVAL`
 - `EditMode.EDIT_MODE_OUTPAINT`
 - `EditMode.EDIT_MODE_PRODUCT_IMAGE`
 - `EditMode.EDIT_MODE_STYLE`
- `EmbedContentConfig`
 - `EmbedContentConfig.auto_truncate`
 - `EmbedContentConfig.http_options`
 - `EmbedContentConfig.mime_type`
 - `EmbedContentConfig.output_dimensionality`
 - `EmbedContentConfig.task_type`

Skip to content `:entConfig.title`

- `EmbedContentConfigDict`
 - `EmbedContentConfigDict.auto_truncate`
 - `EmbedContentConfigDict.http_options`
 - `EmbedContentConfigDict.mime_type`
 - `EmbedContentConfigDict.output_dimensionality`
 - `EmbedContentConfigDict.task_type`
 - `EmbedContentConfigDict.title`
- `EmbedContentMetadata`
 - `EmbedContentMetadata.billable_character_count`
- `EmbedContentMetadataDict`
 - `EmbedContentMetadataDict.billable_character_count`
- `EmbedContentResponse`
 - `EmbedContentResponse.embeddings`
 - `EmbedContentResponse.metadata`
- `EmbedContentResponseDict`
 - `EmbedContentResponseDict.embeddings`
 - `EmbedContentResponseDict.metadata`
- `EncryptionSpec`
 - `EncryptionSpec.kms_key_name`
- `EncryptionSpecDict`
 - `EncryptionSpecDict.kms_key_name`
- `EndSensitivity`
 - `EndSensitivity.END_SENSITIVITY_HIGH`
 - `EndSensitivity.END_SENSITIVITY_LOW`
 - `EndSensitivity.END_SENSITIVITY_UNSPECIFIED`
- `Endpoint`
 - `Endpoint.deployed_model_id`
 - `Endpoint.name`
- `EndpointDict`
 - `EndpointDict.deployed_model_id`
 - `EndpointDict.name`
- `EnterpriseWebSearch`
- `EnterpriseWebSearchDict`
- `Environment`
 - `Environment.ENVIRONMENT_BROWSER`
 - `Environment.ENVIRONMENT_UNSPECIFIED`
- `ExecutableCode`

Skip to content `leCode.code`
`leCode.language`

- ExecutableCodeDict
 - ExecutableCodeDict.code
 - ExecutableCodeDict.language
- ExternalApi
 - ExternalApi.api_auth
 - ExternalApi.api_spec
 - ExternalApi.auth_config
 - ExternalApi.elastic_search_params
 - ExternalApi.endpoint
 - ExternalApi.simple_search_params
- ExternalApiDict
 - ExternalApiDict.api_auth
 - ExternalApiDict.api_spec
 - ExternalApiDict.auth_config
 - ExternalApiDict.elastic_search_params
 - ExternalApiDict.endpoint
 - ExternalApiDict.simple_search_params
- ExternalApiElasticSearchParams
 - ExternalApiElasticSearchParams.index
 - ExternalApiElasticSearchParams.num_hits
 - ExternalApiElasticSearchParams.search_template
- ExternalApiElasticSearchParamsDict
 - ExternalApiElasticSearchParamsDict.index
 - ExternalApiElasticSearchParamsDict.num_hits
 - ExternalApiElasticSearchParamsDict.search_template
- ExternalApiSimpleSearchParams
- ExternalApiSimpleSearchParamsDict
- FeatureSelectionPreference
 - FeatureSelectionPreference.BALANCED
 - FeatureSelectionPreference.FEATURE_SELECTION_PREFERENCE_UNSPECIFIED
 - FeatureSelectionPreference.PRIORITIZE_COST
 - FeatureSelectionPreference.PRIORITIZE_QUALITY
- FetchPredictOperationConfig
 - FetchPredictOperationConfig.http_options
- FetchPredictOperationConfigDict
 - FetchPredictOperationConfigDict.http_options

[Skip to content](#)

- `File`
 - `File.create_time`
 - `File.display_name`
 - `File.download_uri`
 - `File.error`
 - `File.expiration_time`
 - `File.mime_type`
 - `File.name`
 - `File.sha256_hash`
 - `File.size_bytes`
 - `File.source`
 - `File.state`
 - `File.update_time`
 - `File.uri`
 - `File.video_metadata`
- `FileData`
 - `FileData.display_name`
 - `FileData.file_uri`
 - `FileData.mime_type`
- `FileDict`
 - `FileDict.create_time`
 - `FileDict.display_name`
 - `FileDict.download_uri`
 - `FileDict.error`
 - `FileDict.expiration_time`
 - `FileDict.mime_type`
 - `FileDict.name`
 - `FileDict.sha256_hash`
 - `FileDict.size_bytes`
 - `FileDict.source`
 - `FileDict.state`
 - `FileDict.update_time`
 - `FileDict.uri`

Skip to content [.video_metadata](#)

- `FileSource`
 - `FileSource.GENERATED`
 - `FileSource.SOURCE_UNSPECIFIED`
 - `FileSource.UPLOADED`
- `FileState`
 - `FileState.ACTIVE`
 - `FileState.FAILED`
 - `FileState.PROCESSING`
 - `FileState.STATE_UNSPECIFIED`
- `FileStatus`
 - `FileStatus.code`
 - `FileStatus.details`
 - `FileStatus.message`
- `FileStatusDict`
 - `FileStatusDict.code`
 - `FileStatusDict.details`
 - `FileStatusDict.message`
- `FinishReason`
 - `FinishReason.BLOCKLIST`
 - `FinishReason.FINISH_REASON_UNSPECIFIED`
 - `FinishReason.IMAGE_SAFETY`
 - `FinishReason.LANGUAGE`
 - `FinishReason.MALFORMED_FUNCTION_CALL`
 - `FinishReason.MAX_TOKENS`
 - `FinishReason.OTHER`
 - `FinishReason.PROHIBITED_CONTENT`
 - `FinishReason.RECITATION`
 - `FinishReason.SAFETY`
 - `FinishReason.SPII`
 - `FinishReason.STOP`
 - `FinishReason.UNEXPECTED_TOOL_CALL`
- `FunctionCall`
 - `FunctionCall.args`
 - `FunctionCall.id`
 - `FunctionCall.name`
- `FunctionCallDict`
 - `FunctionCallDict.args`

Skip to content `callDict.id`
`callDict.name`

- FunctionCallingConfig
 - FunctionCallingConfig.allowed_function_names
 - FunctionCallingConfig.mode
- FunctionCallingConfigDict
 - FunctionCallingConfigDict.allowed_function_names
 - FunctionCallingConfigDict.mode
- FunctionCallingConfigMode
 - FunctionCallingConfigMode.ANY
 - FunctionCallingConfigMode.AUTO
 - FunctionCallingConfigMode.MODE_UNSPECIFIED
 - FunctionCallingConfigMode.NONE
- FunctionDeclaration
 - FunctionDeclaration.behavior
 - FunctionDeclaration.description
 - FunctionDeclaration.name
 - FunctionDeclaration.parameters
 - FunctionDeclaration.parameters_json_schema
 - FunctionDeclaration.response
 - FunctionDeclaration.response_json_schema
 - FunctionDeclaration.from_callable()
 - FunctionDeclaration.from_callable_with_api_option()
- FunctionDeclarationDict
 - FunctionDeclarationDict.behavior
 - FunctionDeclarationDict.description
 - FunctionDeclarationDict.name
 - FunctionDeclarationDict.parameters
 - FunctionDeclarationDict.parameters_json_schema
 - FunctionDeclarationDict.response
 - FunctionDeclarationDict.response_json_schema
- FunctionResponse
 - FunctionResponse.id
 - FunctionResponse.name
 - FunctionResponse.response
 - FunctionResponse.scheduling
 - FunctionResponse.will_continue
 - FunctionResponse.from_mcp_response()

[Skip to content](#)

- FunctionResponseDict
 - FunctionResponseDict.id
 - FunctionResponseDict.name
 - FunctionResponseDict.response
 - FunctionResponseDict.scheduling
 - FunctionResponseDict.will_continue
- FunctionResponseScheduling
 - FunctionResponseScheduling.INTERRUPT
 - FunctionResponseScheduling.SCHEDULING_UNSPECIFIED
 - FunctionResponseScheduling.SILENT
 - FunctionResponseScheduling.WHEN_IDLE

[Skip to content](#)

- GenerateContentConfig
 - GenerateContentConfig.audio_timestamp
 - GenerateContentConfig.automatic_function_calling
 - GenerateContentConfig.cached_content
 - GenerateContentConfig.candidate_count
 - GenerateContentConfig.frequency_penalty
 - GenerateContentConfig.http_options
 - GenerateContentConfig.labels
 - GenerateContentConfig.logprobs
 - GenerateContentConfig.max_output_tokens
 - GenerateContentConfig.media_resolution
 - GenerateContentConfig.model_selection_config
 - GenerateContentConfig.presence_penalty
 - GenerateContentConfig.response_json_schema
 - GenerateContentConfig.response_logprobs
 - GenerateContentConfig.response_mime_type
 - GenerateContentConfig.response_modalities
 - GenerateContentConfig.response_schema
 - GenerateContentConfig.routing_config
 - GenerateContentConfig.safety_settings
 - GenerateContentConfig.seed
 - GenerateContentConfig.speech_config
 - GenerateContentConfig.stop_sequences
 - GenerateContentConfig.system_instruction
 - GenerateContentConfig.temperature
 - GenerateContentConfig.thinking_config
 - GenerateContentConfig.tool_config
 - GenerateContentConfig.tools
 - GenerateContentConfig.top_k
 - GenerateContentConfig.top_p

[Skip to content](#)

- GenerateContentConfigDict
 - GenerateContentConfigDict.audio_timestamp
 - GenerateContentConfigDict.automatic_function_calling
 - GenerateContentConfigDict.cached_content
 - GenerateContentConfigDict.candidate_count
 - GenerateContentConfigDict.frequency_penalty
 - GenerateContentConfigDict.http_options
 - GenerateContentConfigDict.labels
 - GenerateContentConfigDict.logprobs
 - GenerateContentConfigDict.max_output_tokens
 - GenerateContentConfigDict.media_resolution
 - GenerateContentConfigDict.model_selection_config
 - GenerateContentConfigDict.presence_penalty
 - GenerateContentConfigDict.response_json_schema
 - GenerateContentConfigDict.response_logprobs
 - GenerateContentConfigDict.response_mime_type
 - GenerateContentConfigDict.response_modalities
 - GenerateContentConfigDict.response_schema
 - GenerateContentConfigDict.routing_config
 - GenerateContentConfigDict.safety_settings
 - GenerateContentConfigDict.seed
 - GenerateContentConfigDict.speech_config
 - GenerateContentConfigDict.stop_sequences
 - GenerateContentConfigDict.system_instruction
 - GenerateContentConfigDict.temperature
 - GenerateContentConfigDict.thinking_config
 - GenerateContentConfigDict.tool_config
 - GenerateContentConfigDict.tools
 - GenerateContentConfigDict.top_k
 - GenerateContentConfigDict.top_p

[Skip to content](#)

- GenerateContentResponse
 - GenerateContentResponse.automatic_function_calling_history
 - GenerateContentResponse.candidates
 - GenerateContentResponse.create_time
 - GenerateContentResponse.model_version
 - GenerateContentResponse.parsed
 - GenerateContentResponse.prompt_feedback
 - GenerateContentResponse.response_id
 - GenerateContentResponse.sdk_http_response
 - GenerateContentResponse.usage_metadata
 - GenerateContentResponse.code_execution_result
 - GenerateContentResponse.executable_code
 - GenerateContentResponse.function_calls
 - GenerateContentResponse.text
- GenerateContentResponseDict
 - GenerateContentResponseDict.candidates
 - GenerateContentResponseDict.create_time
 - GenerateContentResponseDict.model_version
 - GenerateContentResponseDict.prompt_feedback
 - GenerateContentResponseDict.response_id
 - GenerateContentResponseDict.sdk_http_response
 - GenerateContentResponseDict.usage_metadata
- GenerateContentResponsePromptFeedback
 - GenerateContentResponsePromptFeedback.block_reason
 - GenerateContentResponsePromptFeedback.block_reason_message
 - GenerateContentResponsePromptFeedback.safety_ratings
- GenerateContentResponsePromptFeedbackDict
 - GenerateContentResponsePromptFeedbackDict.block_reason
 - GenerateContentResponsePromptFeedbackDict.block_reason_message
 - GenerateContentResponsePromptFeedbackDict.safety_ratings

[Skip to content](#)

- GenerateContentResponseUsageMetadata
 - GenerateContentResponseUsageMetadata.cache_tokens_details
 - GenerateContentResponseUsageMetadata.cached_content_token_count
 - GenerateContentResponseUsageMetadata.candidates_token_count
 - GenerateContentResponseUsageMetadata.candidates_tokens_details
 - GenerateContentResponseUsageMetadata.prompt_token_count
 - GenerateContentResponseUsageMetadata.prompt_tokens_details
 - GenerateContentResponseUsageMetadata.thoughts_token_count
 - GenerateContentResponseUsageMetadata.tool_use_prompt_token_count
 - GenerateContentResponseUsageMetadata.tool_use_prompt_tokens_details
 - GenerateContentResponseUsageMetadata.total_token_count
 - GenerateContentResponseUsageMetadata.traffic_type
- GenerateContentResponseUsageMetadataDict
 - GenerateContentResponseUsageMetadataDict.cache_tokens_details
 - GenerateContentResponseUsageMetadataDict.cached_content_token_count
 - GenerateContentResponseUsageMetadataDict.candidates_token_count
 - GenerateContentResponseUsageMetadataDict.candidates_tokens_details
 - GenerateContentResponseUsageMetadataDict.prompt_token_count
 - GenerateContentResponseUsageMetadataDict.prompt_tokens_details
 - GenerateContentResponseUsageMetadataDict.thoughts_token_count
 - GenerateContentResponseUsageMetadataDict.tool_use_prompt_token_count
 - GenerateContentResponseUsageMetadataDict.tool_use_prompt_tokens_details
 - GenerateContentResponseUsageMetadataDict.total_token_count
 - GenerateContentResponseUsageMetadataDict.traffic_type

[Skip to content](#)

- GenerateImagesConfig
 - GenerateImagesConfig.add_watermark
 - GenerateImagesConfig.aspect_ratio
 - GenerateImagesConfig.enhance_prompt
 - GenerateImagesConfig.guidance_scale
 - GenerateImagesConfig.http_options
 - GenerateImagesConfig.include_rai_reason
 - GenerateImagesConfig.include_safety_attributes
 - GenerateImagesConfig.language
 - GenerateImagesConfig.negative_prompt
 - GenerateImagesConfig.number_of_images
 - GenerateImagesConfig.output_compression_quality
 - GenerateImagesConfig.output_gcs_uri
 - GenerateImagesConfig.output_mime_type
 - GenerateImagesConfig.person_generation
 - GenerateImagesConfig.safety_filter_level
 - GenerateImagesConfig.seed
- GenerateImagesConfigDict
 - GenerateImagesConfigDict.add_watermark
 - GenerateImagesConfigDict.aspect_ratio
 - GenerateImagesConfigDict.enhance_prompt
 - GenerateImagesConfigDict.guidance_scale
 - GenerateImagesConfigDict.http_options
 - GenerateImagesConfigDict.include_rai_reason
 - GenerateImagesConfigDict.include_safety_attributes
 - GenerateImagesConfigDict.language
 - GenerateImagesConfigDict.negative_prompt
 - GenerateImagesConfigDict.number_of_images
 - GenerateImagesConfigDict.output_compression_quality
 - GenerateImagesConfigDict.output_gcs_uri
 - GenerateImagesConfigDict.output_mime_type
 - GenerateImagesConfigDict.person_generation
 - GenerateImagesConfigDict.safety_filter_level
 - GenerateImagesConfigDict.seed
- GenerateImagesResponse
 - GenerateImagesResponse.generated_images
 - GenerateImagesResponse.positive_prompt_safety_attributes

Skip to content `gesResponseDict`
`gesResponseDict.imagesResponseDict.generated_images`
▪ `GenerateImagesResponseDict.positive_prompt_safety_attributes`

- GenerateVideosConfig
 - GenerateVideosConfig.aspect_ratio
 - GenerateVideosConfig.compression_quality
 - GenerateVideosConfig.duration_seconds
 - GenerateVideosConfig.enhance_prompt
 - GenerateVideosConfig.fps
 - GenerateVideosConfig.generate_audio
 - GenerateVideosConfig.http_options
 - GenerateVideosConfig.last_frame
 - GenerateVideosConfig.negative_prompt
 - GenerateVideosConfig.number_of_videos
 - GenerateVideosConfig.output_gcs_uri
 - GenerateVideosConfig.person_generation
 - GenerateVideosConfig.pubsub_topic
 - GenerateVideosConfig.resolution
 - GenerateVideosConfig.seed
- GenerateVideosConfigDict
 - GenerateVideosConfigDict.aspect_ratio
 - GenerateVideosConfigDict.compression_quality
 - GenerateVideosConfigDict.duration_seconds
 - GenerateVideosConfigDict.enhance_prompt
 - GenerateVideosConfigDict.fps
 - GenerateVideosConfigDict.generate_audio
 - GenerateVideosConfigDict.http_options
 - GenerateVideosConfigDict.last_frame
 - GenerateVideosConfigDict.negative_prompt
 - GenerateVideosConfigDict.number_of_videos
 - GenerateVideosConfigDict.output_gcs_uri
 - GenerateVideosConfigDict.person_generation
 - GenerateVideosConfigDict.pubsub_topic
 - GenerateVideosConfigDict.resolution
 - GenerateVideosConfigDict.seed
- GenerateVideosOperation
 - GenerateVideosOperation.response
 - GenerateVideosOperation.result
 - GenerateVideosOperation.from_api_response()
- GenerateVideosResponse

Skip to content /ideosResponse.generated_videos
/VideosResponse.rai_media_filtered_count
▪ GenerateVideosResponse.rai_media_filtered_reasons

- GenerateVideosResponseDict
 - GenerateVideosResponseDict.generated_videos
 - GenerateVideosResponseDict.rai_media_filtered_count
 - GenerateVideosResponseDict.rai_media_filtered_reasons
- GeneratedImage
 - GeneratedImage.enhanced_prompt
 - GeneratedImage.image
 - GeneratedImage.rai_filtered_reason
 - GeneratedImage.safety_attributes
- GeneratedImageDict
 - GeneratedImageDict.enhanced_prompt
 - GeneratedImageDict.image
 - GeneratedImageDict.rai_filtered_reason
 - GeneratedImageDict.safety_attributes
- GeneratedVideo
 - GeneratedVideo.video
- GeneratedVideoDict
 - GeneratedVideoDict.video

[Skip to content](#)

- GenerationConfig
 - GenerationConfig.audio_timestamp
 - GenerationConfig.candidate_count
 - GenerationConfig.enable_affective_dialog
 - GenerationConfig.frequency_penalty
 - GenerationConfig.logprobs
 - GenerationConfig.max_output_tokens
 - GenerationConfig.media_resolution
 - GenerationConfig.model_selection_config
 - GenerationConfig.presence_penalty
 - GenerationConfig.response_json_schema
 - GenerationConfig.response_logprobs
 - GenerationConfig.response_mime_type
 - GenerationConfig.response_modalities
 - GenerationConfig.response_schema
 - GenerationConfig.routing_config
 - GenerationConfig.seed
 - GenerationConfig.speech_config
 - GenerationConfig.stop_sequences
 - GenerationConfig.temperature
 - GenerationConfig.thinking_config
 - GenerationConfig.top_k
 - GenerationConfig.top_p

[Skip to content](#)

- GenerationConfigDict
 - GenerationConfigDict.audio_timestamp
 - GenerationConfigDict.candidate_count
 - GenerationConfigDict.enable_affective_dialog
 - GenerationConfigDict.frequency_penalty
 - GenerationConfigDict.logprobs
 - GenerationConfigDict.max_output_tokens
 - GenerationConfigDict.media_resolution
 - GenerationConfigDict.model_selection_config
 - GenerationConfigDict.presence_penalty
 - GenerationConfigDict.response_json_schema
 - GenerationConfigDict.response_logprobs
 - GenerationConfigDict.response_mime_type
 - GenerationConfigDict.response_modalities
 - GenerationConfigDict.response_schema
 - GenerationConfigDict.routing_config
 - GenerationConfigDict.seed
 - GenerationConfigDict.speech_config
 - GenerationConfigDict.stop_sequences
 - GenerationConfigDict.temperature
 - GenerationConfigDict.thinking_config
 - GenerationConfigDict.top_k
 - GenerationConfigDict.top_p
- GenerationConfigRoutingConfig
 - GenerationConfigRoutingConfig.auto_mode
 - GenerationConfigRoutingConfig.manual_mode
- GenerationConfigRoutingConfigAutoRoutingMode
 - GenerationConfigRoutingConfigAutoRoutingMode.model_routing_preference
- GenerationConfigRoutingConfigAutoRoutingModeDict
 - GenerationConfigRoutingConfigAutoRoutingModeDict.model_routing_preference
- GenerationConfigRoutingConfigDict
 - GenerationConfigRoutingConfigDict.auto_mode
 - GenerationConfigRoutingConfigDict.manual_mode
- GenerationConfigRoutingConfigManualRoutingMode
 - GenerationConfigRoutingConfigManualRoutingMode.model_name
- GenerationConfigRoutingConfigManualRoutingModeDict
 - GenerationConfigRoutingConfigManualRoutingModeDict.model_name

Skip to content [onConfigThinkingConfig](#)
[onConfigThinkingConfig.include_thoughts](#)
▪ GenerationConfigThinkingConfig.thinking_budget

- GenerationConfigThinkingConfigDict
 - GenerationConfigThinkingConfigDict.include_thoughts
 - GenerationConfigThinkingConfigDict.thinking_budget
- GetBatchJobConfig
 - GetBatchJobConfig.http_options
- GetBatchJobConfigDict
 - GetBatchJobConfigDict.http_options
- GetCachedContentConfig
 - GetCachedContentConfig.http_options
- GetCachedContentConfigDict
 - GetCachedContentConfigDict.http_options
- GetFileConfig
 - GetFileConfig.http_options
- GetFileConfigDict
 - GetFileConfigDict.http_options
- GetModelConfig
 - GetModelConfig.http_options
- GetModelConfigDict
 - GetModelConfigDict.http_options
- GetOperationConfig
 - GetOperationConfig.http_options
- GetOperationConfigDict
 - GetOperationConfigDict.http_options
- GetTuningJobConfig
 - GetTuningJobConfig.http_options
- GetTuningJobConfigDict
 - GetTuningJobConfigDict.http_options
- GoogleMaps
 - GoogleMaps.auth_config
- GoogleMapsDict
 - GoogleMapsDict.auth_config
- GoogleRpcStatus
 - GoogleRpcStatus.code
 - GoogleRpcStatus.details
 - GoogleRpcStatus.message
- GoogleRpcStatusDict
 - GoogleRpcStatusDict.code

Skip to content `>StatusDict.details`
 `>StatusDict.message`

- GoogleSearch
 - GoogleSearch.time_range_filter
- GoogleSearchDict
 - GoogleSearchDict.time_range_filter
- GoogleSearchRetrieval
 - GoogleSearchRetrieval.dynamic_retrieval_config
- GoogleSearchRetrievalDict
 - GoogleSearchRetrievalDict.dynamic_retrieval_config
- GoogleTypeDate
 - GoogleTypeDate.day
 - GoogleTypeDate.month
 - GoogleTypeDate.year
- GoogleTypeDateDict
 - GoogleTypeDateDict.day
 - GoogleTypeDateDict.month
 - GoogleTypeDateDict.year
- GroundingChunk
 - GroundingChunk.retrieved_context
 - GroundingChunk.web
- GroundingChunkDict
 - GroundingChunkDict.retrieved_context
 - GroundingChunkDict.web
- GroundingChunkRetrievedContext
 - GroundingChunkRetrievedContext.rag_chunk
 - GroundingChunkRetrievedContext.text
 - GroundingChunkRetrievedContext.title
 - GroundingChunkRetrievedContext.uri
- GroundingChunkRetrievedContextDict
 - GroundingChunkRetrievedContextDict.rag_chunk
 - GroundingChunkRetrievedContextDict.text
 - GroundingChunkRetrievedContextDict.title
 - GroundingChunkRetrievedContextDict.uri
- GroundingChunkWeb
 - GroundingChunkWeb.domain
 - GroundingChunkWeb.title
 - GroundingChunkWeb.uri
- GroundingChunkWebDict
 - GroundingChunkWebDict.domain
 - GroundingChunkWebDict.title
 - GroundingChunkWebDict.uri

Skip to content ↴
GroundingChunkWebDict.domain
GroundingChunkWebDict.title
GroundingChunkWebDict.uri

- `GroundingMetadata`
 - `GroundingMetadata.grounding_chunks`
 - `GroundingMetadata.grounding_supports`
 - `GroundingMetadata.retrieval_metadata`
 - `GroundingMetadata.retrieval_queries`
 - `GroundingMetadata.search_entry_point`
 - `GroundingMetadata.web_search_queries`
- `GroundingMetadataDict`
 - `GroundingMetadataDict.grounding_chunks`
 - `GroundingMetadataDict.grounding_supports`
 - `GroundingMetadataDict.retrieval_metadata`
 - `GroundingMetadataDict.retrieval_queries`
 - `GroundingMetadataDict.search_entry_point`
 - `GroundingMetadataDict.web_search_queries`
- `GroundingSupport`
 - `GroundingSupport.confidence_scores`
 - `GroundingSupport.grounding_chunk_indices`
 - `GroundingSupport.segment`
- `GroundingSupportDict`
 - `GroundingSupportDict.confidence_scores`
 - `GroundingSupportDict.grounding_chunk_indices`
 - `GroundingSupportDict.segment`
- `HarmBlockMethod`
 - `HarmBlockMethod.HARM_BLOCK_METHOD_UNSPECIFIED`
 - `HarmBlockMethod.PROBABILITY`
 - `HarmBlockMethod.SEVERITY`
- `HarmBlockThreshold`
 - `HarmBlockThreshold.BLOCK_LOW_AND ABOVE`
 - `HarmBlockThreshold.BLOCK_MEDIUM_AND ABOVE`
 - `HarmBlockThreshold.BLOCK_NONE`
 - `HarmBlockThreshold.BLOCK_ONLY_HIGH`
 - `HarmBlockThreshold.HARM_BLOCK_THRESHOLD_UNSPECIFIED`
 - `HarmBlockThreshold.OFF`

[Skip to content](#)

- HarmCategory
 - HarmCategory.HARM_CATEGORY_CIVIC_INTEGRITY
 - HarmCategory.HARM_CATEGORY_DANGEROUS_CONTENT
 - HarmCategory.HARM_CATEGORY_HARASSMENT
 - HarmCategory.HARM_CATEGORY_HATE_SPEECH
 - HarmCategory.HARM_CATEGORY_IMAGE_DANGEROUS_CONTENT
 - HarmCategory.HARM_CATEGORY_IMAGE_HARASSMENT
 - HarmCategory.HARM_CATEGORY_IMAGE_HATE
 - HarmCategory.HARM_CATEGORY_IMAGE_SEXUALLY_EXPLICIT
 - HarmCategory.HARM_CATEGORY_SEXUALLY_EXPLICIT
 - HarmCategory.HARM_CATEGORY_UNSPECIFIED
- HarmProbability
 - HarmProbability.HARM_PROBABILITY_UNSPECIFIED
 - HarmProbability.HIGH
 - HarmProbability.LOW
 - HarmProbability.MEDIUM
 - HarmProbability.NEGLIGIBLE
- HarmSeverity
 - HarmSeverity.HARM_SEVERITY_HIGH
 - HarmSeverity.HARM_SEVERITY_LOW
 - HarmSeverity.HARM_SEVERITY_MEDIUM
 - HarmSeverity.HARM_SEVERITY_NEGLIGIBLE
 - HarmSeverity.HARM_SEVERITY_UNSPECIFIED
- HttpOptions
 - HttpOptions.api_version
 - HttpOptions.async_client_args
 - HttpOptions.base_url
 - HttpOptions.client_args
 - HttpOptions.extra_body
 - HttpOptions.headers
 - HttpOptions.retry_options
 - HttpOptions.timeout

[Skip to content](#)

- `HttpOptionsDict`
 - `HttpOptionsDict.api_version`
 - `HttpOptionsDict.async_client_args`
 - `HttpOptionsDict.base_url`
 - `HttpOptionsDict.client_args`
 - `HttpOptionsDict.extra_body`
 - `HttpOptionsDict.headers`
 - `HttpOptionsDict.retry_options`
 - `HttpOptionsDict.timeout`
- `HttpResponse`
 - `HttpResponse.body`
 - `HttpResponse.headers`
- `HttpResponseDict`
 - `HttpResponseDict.body`
 - `HttpResponseDict.headers`
- `HttpRetryOptions`
 - `HttpRetryOptions.attempts`
 - `HttpRetryOptions.exp_base`
 - `HttpRetryOptions.http_status_codes`
 - `HttpRetryOptions.initial_delay`
 - `HttpRetryOptions.jitter`
 - `HttpRetryOptions.max_delay`
- `HttpRetryOptionsDict`
 - `HttpRetryOptionsDict.attempts`
 - `HttpRetryOptionsDict.exp_base`
 - `HttpRetryOptionsDict.http_status_codes`
 - `HttpRetryOptionsDict.initial_delay`
 - `HttpRetryOptionsDict.jitter`
 - `HttpRetryOptionsDict.max_delay`
- `Image`
 - `Image.gcs_uri`
 - `Image.image_bytes`
 - `Image.mime_type`
 - `Image.from_file()`
 - `Image.model_post_init()`
 - `Image.save()`
 - `Image.show()`

[Skip to content](#)

- `ImageDict`
 - `ImageDict.gcs_uri`
 - `ImageDict.image_bytes`
 - `ImageDict.mime_type`
- `ImagePromptLanguage`
 - `ImagePromptLanguage.auto`
 - `ImagePromptLanguage.en`
 - `ImagePromptLanguage.es`
 - `ImagePromptLanguage.hi`
 - `ImagePromptLanguage.ja`
 - `ImagePromptLanguage.ko`
 - `ImagePromptLanguage.pt`
 - `ImagePromptLanguage.zh`
- `InlinedRequest`
 - `InlinedRequest.config`
 - `InlinedRequest.contents`
 - `InlinedRequest.model`
- `InlinedRequestDict`
 - `InlinedRequestDict.config`
 - `InlinedRequestDict.contents`
 - `InlinedRequestDict.model`
- `InlinedResponse`
 - `InlinedResponse.error`
 - `InlinedResponse.response`
- `InlinedResponseDict`
 - `InlinedResponseDict.error`
 - `InlinedResponseDict.response`
- `Interval`
 - `Interval.end_time`
 - `Interval.start_time`
- `IntervalDict`
 - `IntervalDict.end_time`
 - `IntervalDict.start_time`

[Skip to content](#)

- `JSONSchema`
 - `JSONSchema.any_of`
 - `JSONSchema.default`
 - `JSONSchema.description`
 - `JSONSchema.enum`
 - `JSONSchema.format`
 - `JSONSchema.items`
 - `JSONSchema.max_items`
 - `JSONSchema.max_length`
 - `JSONSchema.max_properties`
 - `JSONSchema.maximum`
 - `JSONSchema.min_items`
 - `JSONSchema.min_length`
 - `JSONSchema.min_properties`
 - `JSONSchema.minimum`
 - `JSONSchema.pattern`
 - `JSONSchema.properties`
 - `JSONSchema.required`
 - `JSONSchema.title`
 - `JSONSchema.type`
- `JSONSchemaType`
 - `JSONSchemaType.ARRAY`
 - `JSONSchemaType.BOOLEAN`
 - `JSONSchemaType.INTEGER`
 - `JSONSchemaType.NULL`
 - `JSONSchemaType.NUMBER`
 - `JSONSchemaType.OBJECT`
 - `JSONSchemaType.STRING`
- `JobError`
 - `JobError.code`
 - `JobError.details`
 - `JobError.message`
- `JobErrorDict`
 - `JobErrorDict.code`
 - `JobErrorDict.details`
 - `JobErrorDict.message`

[Skip to content](#)

- JobState
 - JobState.JOB_STATE_CANCELLED
 - JobState.JOB_STATE_CANCELLING
 - JobState.JOB_STATE_EXPIRED
 - JobState.JOB_STATE_FAILED
 - JobState.JOB_STATE_PARTIALLY_SUCCEEDED
 - JobState.JOB_STATE_PAUSED
 - JobState.JOB_STATE_PENDING
 - JobState.JOB_STATE_QUEUED
 - JobState.JOB_STATE_RUNNING
 - JobState.JOB_STATE_SUCCEEDED
 - JobState.JOB_STATE_UNSPECIFIED
 - JobState.JOB_STATE_UPDATING
- Language
 - Language.LANGUAGE_UNSPECIFIED
 - Language.PYTHON
- LatLng
 - LatLng.latitude
 - LatLng.longitude
- LatLngDict
 - LatLngDict.latitude
 - LatLngDict.longitude
- ListBatchJobsConfig
 - ListBatchJobsConfig.filter
 - ListBatchJobsConfig.http_options
 - ListBatchJobsConfig.page_size
 - ListBatchJobsConfig.page_token
- ListBatchJobsConfigDict
 - ListBatchJobsConfigDict.filter
 - ListBatchJobsConfigDict.http_options
 - ListBatchJobsConfigDict.page_size
 - ListBatchJobsConfigDict.page_token
- ListBatchJobsResponse
 - ListBatchJobsResponse.batch_jobs
 - ListBatchJobsResponse.next_page_token
- ListBatchJobsResponseDict
 - ListBatchJobsResponseDict.batch_jobs

Skip to content [↑](#)JobsResponseDict.next_page_token

- `ListCachedContentsConfig`
 - `ListCachedContentsConfig.http_options`
 - `ListCachedContentsConfig.page_size`
 - `ListCachedContentsConfig.page_token`
- `ListCachedContentsConfigDict`
 - `ListCachedContentsConfigDict.http_options`
 - `ListCachedContentsConfigDict.page_size`
 - `ListCachedContentsConfigDict.page_token`
- `ListCachedContentsResponse`
 - `ListCachedContentsResponse.cached_contents`
 - `ListCachedContentsResponse.next_page_token`
- `ListCachedContentsResponseDict`
 - `ListCachedContentsResponseDict.cached_contents`
 - `ListCachedContentsResponseDict.next_page_token`
- `ListFilesConfig`
 - `ListFilesConfig.http_options`
 - `ListFilesConfig.page_size`
 - `ListFilesConfig.page_token`
- `ListFilesConfigDict`
 - `ListFilesConfigDict.http_options`
 - `ListFilesConfigDict.page_size`
 - `ListFilesConfigDict.page_token`
- `ListFilesResponse`
 - `ListFilesResponse.files`
 - `ListFilesResponse.next_page_token`
- `ListFilesResponseDict`
 - `ListFilesResponseDict.files`
 - `ListFilesResponseDict.next_page_token`
- `ListModelsConfig`
 - `ListModelsConfig.filter`
 - `ListModelsConfig.http_options`
 - `ListModelsConfig.page_size`
 - `ListModelsConfig.page_token`
 - `ListModelsConfig.query_base`
- `ListModelsConfigDict`
 - `ListModelsConfigDict.filter`
 - `ListModelsConfigDict.http_options`

Skip to content `lscConfigDict.page_size`
 `lscConfigDict.page_token`
 ▪ `ListModelsConfigDict.query_base`

- `ListModelsResponse`
 - `ListModelsResponse.models`
 - `ListModelsResponse.next_page_token`
- `ListModelsResponseDict`
 - `ListModelsResponseDict.models`
 - `ListModelsResponseDict.next_page_token`
- `ListTuningJobsConfig`
 - `ListTuningJobsConfig.filter`
 - `ListTuningJobsConfig.http_options`
 - `ListTuningJobsConfig.page_size`
 - `ListTuningJobsConfig.page_token`
- `ListTuningJobsConfigDict`
 - `ListTuningJobsConfigDict.filter`
 - `ListTuningJobsConfigDict.http_options`
 - `ListTuningJobsConfigDict.page_size`
 - `ListTuningJobsConfigDict.page_token`
- `ListTuningJobsResponse`
 - `ListTuningJobsResponse.next_page_token`
 - `ListTuningJobsResponse.tuning_jobs`
- `ListTuningJobsResponseDict`
 - `ListTuningJobsResponseDict.next_page_token`
 - `ListTuningJobsResponseDict.tuning_jobs`
- `LiveClientContent`
 - `LiveClientContent.turn_complete`
 - `LiveClientContent.turns`
- `LiveClientContentDict`
 - `LiveClientContentDict.turn_complete`
 - `LiveClientContentDict.turns`
- `LiveClientMessage`
 - `LiveClientMessage.client_content`
 - `LiveClientMessage.realtime_input`
 - `LiveClientMessage.setup`
 - `LiveClientMessage.tool_response`
- `LiveClientMessageDict`
 - `LiveClientMessageDict.client_content`
 - `LiveClientMessageDict.realtime_input`
 - `LiveClientMessageDict.setup`

Skip to content `\itMessageDict.tool_response`

- `LiveClientRealtimeInput`
 - `LiveClientRealtimeInput.activity_end`
 - `LiveClientRealtimeInput.activity_start`
 - `LiveClientRealtimeInput.audio`
 - `LiveClientRealtimeInput.audio_stream_end`
 - `LiveClientRealtimeInput.media_chunks`
 - `LiveClientRealtimeInput.text`
 - `LiveClientRealtimeInput.video`
- `LiveClientRealtimeInputDict`
 - `LiveClientRealtimeInputDict.activity_end`
 - `LiveClientRealtimeInputDict.activity_start`
 - `LiveClientRealtimeInputDict.audio`
 - `LiveClientRealtimeInputDict.audio_stream_end`
 - `LiveClientRealtimeInputDict.media_chunks`
 - `LiveClientRealtimeInputDict.text`
 - `LiveClientRealtimeInputDict.video`
- `LiveClientSetup`
 - `LiveClientSetup.context_window_compression`
 - `LiveClientSetup.generation_config`
 - `LiveClientSetup.input_audio_transcription`
 - `LiveClientSetup.model`
 - `LiveClientSetup.output_audio_transcription`
 - `LiveClientSetup.proactivity`
 - `LiveClientSetup.session_resumption`
 - `LiveClientSetup.system_instruction`
 - `LiveClientSetup.tools`
- `LiveClientSetupDict`
 - `LiveClientSetupDict.context_window_compression`
 - `LiveClientSetupDict.generation_config`
 - `LiveClientSetupDict.input_audio_transcription`
 - `LiveClientSetupDict.model`
 - `LiveClientSetupDict.output_audio_transcription`
 - `LiveClientSetupDict.proactivity`
 - `LiveClientSetupDict.session_resumption`
 - `LiveClientSetupDict.system_instruction`
 - `LiveClientSetupDict.tools`
- `LiveClientToolResponse`

Skip to content `itToolResponse.function_responses`

`oolResponseDict`

- `LiveClientToolResponseDict.function_responses`

- `LiveConnectConfig`
 - `LiveConnectConfig.context_window_compression`
 - `LiveConnectConfig.enable_affective_dialog`
 - `LiveConnectConfig.generation_config`
 - `LiveConnectConfig.http_options`
 - `LiveConnectConfig.input_audio_transcription`
 - `LiveConnectConfig.max_output_tokens`
 - `LiveConnectConfig.media_resolution`
 - `LiveConnectConfig.output_audio_transcription`
 - `LiveConnectConfig.proactivity`
 - `LiveConnectConfig.realtime_input_config`
 - `LiveConnectConfig.response_modalities`
 - `LiveConnectConfig.seed`
 - `LiveConnectConfig.session_resumption`
 - `LiveConnectConfig.speech_config`
 - `LiveConnectConfig.system_instruction`
 - `LiveConnectConfig.temperature`
 - `LiveConnectConfig.tools`
 - `LiveConnectConfig.top_k`
 - `LiveConnectConfig.top_p`
- `LiveConnectConfigDict`
 - `LiveConnectConfigDict.context_window_compression`
 - `LiveConnectConfigDict.enable_affective_dialog`
 - `LiveConnectConfigDict.generation_config`
 - `LiveConnectConfigDict.http_options`
 - `LiveConnectConfigDict.input_audio_transcription`
 - `LiveConnectConfigDict.max_output_tokens`
 - `LiveConnectConfigDict.media_resolution`
 - `LiveConnectConfigDict.output_audio_transcription`
 - `LiveConnectConfigDict.proactivity`
 - `LiveConnectConfigDict.realtime_input_config`
 - `LiveConnectConfigDict.response_modalities`
 - `LiveConnectConfigDict.seed`
 - `LiveConnectConfigDict.session_resumption`
 - `LiveConnectConfigDict.speech_config`
 - `LiveConnectConfigDict.system_instruction`
 - `LiveConnectConfigDict.temperature`

Skip to content `ectConfigDict.tools`
`ectConfigDict.top_k`
▪ `LiveConnectConfigDict.top_p`

- `LiveConnectConstraints`
 - `LiveConnectConstraints.config`
 - `LiveConnectConstraints.model`
 - `LiveConnectConstraintsDict`
 - `LiveConnectConstraintsDict.config`
 - `LiveConnectConstraintsDict.model`
 - `LiveConnectParameters`
 - `LiveConnectParameters.config`
 - `LiveConnectParameters.model`
 - `LiveConnectParametersDict`
 - `LiveConnectParametersDict.config`
 - `LiveConnectParametersDict.model`
 - `LiveMusicClientContent`
 - `LiveMusicClientContent.weighted_prompts`
 - `LiveMusicClientContentDict`
 - `LiveMusicClientContentDict.weighted_prompts`
 - `LiveMusicClientMessage`
 - `LiveMusicClientMessage.client_content`
 - `LiveMusicClientMessage.music_generation_config`
 - `LiveMusicClientMessage.playback_control`
 - `LiveMusicClientMessage.setup`
 - `LiveMusicClientMessageDict`
 - `LiveMusicClientMessageDict.client_content`
 - `LiveMusicClientMessageDict.music_generation_config`
 - `LiveMusicClientMessageDict.playback_control`
 - `LiveMusicClientMessageDict.setup`
 - `LiveMusicClientSetup`
 - `LiveMusicClientSetup.model`
 - `LiveMusicClientSetupDict`
 - `LiveMusicClientSetupDict.model`
 - `LiveMusicConnectParameters`
 - `LiveMusicConnectParameters.model`
 - `LiveMusicConnectParametersDict`
 - `LiveMusicConnectParametersDict.model`
 - `LiveMusicFilteredPrompt`
 - `LiveMusicFilteredPrompt.filtered_reason`
 - `LiveMusicFilteredPrompt.text`
- Skip to content `lteredPromptDict`
 `:FilteredPromptDict.filtered_reason`
 - `LiveMusicFilteredPromptDict.text`

- `LiveMusicGenerationConfig`
 - `LiveMusicGenerationConfig.bpm`
 - `LiveMusicGenerationConfig.brightness`
 - `LiveMusicGenerationConfig.density`
 - `LiveMusicGenerationConfig.guidance`
 - `LiveMusicGenerationConfig.mute_bass`
 - `LiveMusicGenerationConfig.mute_drums`
 - `LiveMusicGenerationConfig.only_bass_and_drums`
 - `LiveMusicGenerationConfig.scale`
 - `LiveMusicGenerationConfig.seed`
 - `LiveMusicGenerationConfig.temperature`
 - `LiveMusicGenerationConfig.top_k`
- `LiveMusicGenerationConfigDict`
 - `LiveMusicGenerationConfigDict.bpm`
 - `LiveMusicGenerationConfigDict.brightness`
 - `LiveMusicGenerationConfigDict.density`
 - `LiveMusicGenerationConfigDict.guidance`
 - `LiveMusicGenerationConfigDict.mute_bass`
 - `LiveMusicGenerationConfigDict.mute_drums`
 - `LiveMusicGenerationConfigDict.only_bass_and_drums`
 - `LiveMusicGenerationConfigDict.scale`
 - `LiveMusicGenerationConfigDict.seed`
 - `LiveMusicGenerationConfigDict.temperature`
 - `LiveMusicGenerationConfigDict.top_k`
- `LiveMusicPlaybackControl`
 - `LiveMusicPlaybackControl.PAUSE`
 - `LiveMusicPlaybackControl.PLAY`
 - `LiveMusicPlaybackControl.PLAYBACK_CONTROL_UNSPECIFIED`
 - `LiveMusicPlaybackControl.RESET_CONTEXT`
 - `LiveMusicPlaybackControl.STOP`
- `LiveMusicServerContent`
 - `LiveMusicServerContent.audio_chunks`
- `LiveMusicServerContentDict`
 - `LiveMusicServerContentDict.audio_chunks`
- `LiveMusicServerMessage`
 - `LiveMusicServerMessage.filtered_prompt`
 - `LiveMusicServerMessage.server_content`

Skip to content `>ServerMessage.setup_complete`

- `LiveMusicServerMessageDict`
 - `LiveMusicServerMessageDict.filtered_prompt`
 - `LiveMusicServerMessageDict.server_content`
 - `LiveMusicServerMessageDict.setup_complete`
- `LiveMusicServerSetupComplete`
- `LiveMusicServerSetupCompleteDict`
- `LiveMusicSetConfigParameters`
 - `LiveMusicSetConfigParameters.music_generation_config`
- `LiveMusicSetConfigParametersDict`
 - `LiveMusicSetConfigParametersDict.music_generation_config`
- `LiveMusicSetWeightedPromptsParameters`
 - `LiveMusicSetWeightedPromptsParameters.weighted_prompts`
- `LiveMusicSetWeightedPromptsParametersDict`
 - `LiveMusicSetWeightedPromptsParametersDict.weighted_prompts`
- `LiveMusicSourceMetadata`
 - `LiveMusicSourceMetadata.client_content`
 - `LiveMusicSourceMetadata.music_generation_config`
- `LiveMusicSourceMetadataDict`
 - `LiveMusicSourceMetadataDict.client_content`
 - `LiveMusicSourceMetadataDict.music_generation_config`
- `LiveSendRealtimeInputParameters`
 - `LiveSendRealtimeInputParameters.activity_end`
 - `LiveSendRealtimeInputParameters.activity_start`
 - `LiveSendRealtimeInputParameters.audio`
 - `LiveSendRealtimeInputParameters.audio_stream_end`
 - `LiveSendRealtimeInputParameters.media`
 - `LiveSendRealtimeInputParameters.text`
 - `LiveSendRealtimeInputParameters.video`
- `LiveSendRealtimeInputParametersDict`
 - `LiveSendRealtimeInputParametersDict.activity_end`
 - `LiveSendRealtimeInputParametersDict.activity_start`
 - `LiveSendRealtimeInputParametersDict.audio`
 - `LiveSendRealtimeInputParametersDict.audio_stream_end`
 - `LiveSendRealtimeInputParametersDict.media`
 - `LiveSendRealtimeInputParametersDict.text`
 - `LiveSendRealtimeInputParametersDict.video`

[Skip to content](#)

- `LiveServerContent`
 - `LiveServerContent.generation_complete`
 - `LiveServerContent.grounding_metadata`
 - `LiveServerContent.input_transcription`
 - `LiveServerContent.interrupted`
 - `LiveServerContent.model_turn`
 - `LiveServerContent.output_transcription`
 - `LiveServerContent.turn_complete`
 - `LiveServerContent.url_context_metadata`
- `LiveServerContentDict`
 - `LiveServerContentDict.generation_complete`
 - `LiveServerContentDict.grounding_metadata`
 - `LiveServerContentDict.input_transcription`
 - `LiveServerContentDict.interrupted`
 - `LiveServerContentDict.model_turn`
 - `LiveServerContentDict.output_transcription`
 - `LiveServerContentDict.turn_complete`
 - `LiveServerContentDict.url_context_metadata`
- `LiveServerGoAway`
 - `LiveServerGoAway.time_left`
- `LiveServerGoAwayDict`
 - `LiveServerGoAwayDict.time_left`
- `LiveServerMessage`
 - `LiveServerMessage.go_away`
 - `LiveServerMessage.server_content`
 - `LiveServerMessage.session_resumption_update`
 - `LiveServerMessage.setup_complete`
 - `LiveServerMessage.tool_call`
 - `LiveServerMessage.tool_call_cancellation`
 - `LiveServerMessage.usage_metadata`
 - `LiveServerMessage.data`
 - `LiveServerMessage.text`
- `LiveServerMessageDict`
 - `LiveServerMessageDict.go_away`
 - `LiveServerMessageDict.server_content`
 - `LiveServerMessageDict.session_resumption_update`
 - `LiveServerMessageDict.setup_complete`

Skip to content `LiveServerMessageDict.tool_call`
`LiveServerMessageDict.tool_call_cancellation`
▪ `LiveServerMessageDict.usage_metadata`

- `LiveServerSessionResumptionUpdate`
 - `LiveServerSessionResumptionUpdate.last_consumed_client_message_index`
 - `LiveServerSessionResumptionUpdate.new_handle`
 - `LiveServerSessionResumptionUpdate.resumable`
- `LiveServerSessionResumptionUpdateDict`
 - `LiveServerSessionResumptionUpdateDict.last_consumed_client_message_index`
 - `LiveServerSessionResumptionUpdateDict.new_handle`
 - `LiveServerSessionResumptionUpdateDict.resumable`
- `LiveServerSetupComplete`
 - `LiveServerSetupComplete.session_id`
- `LiveServerSetupCompleteDict`
 - `LiveServerSetupCompleteDict.session_id`
- `LiveServerToolCall`
 - `LiveServerToolCall.function_calls`
- `LiveServerToolCallCancellation`
 - `LiveServerToolCallCancellation.ids`
- `LiveServerToolCallCancellationDict`
 - `LiveServerToolCallCancellationDict.ids`
- `LiveServerToolCallDict`
 - `LiveServerToolCallDict.function_calls`
- `LogprobsResult`
 - `LogprobsResult.chosen_candidates`
 - `LogprobsResult.top_candidates`
- `LogprobsResultCandidate`
 - `LogprobsResultCandidate.log_probability`
 - `LogprobsResultCandidate.token`
 - `LogprobsResultCandidate.token_id`
- `LogprobsResultCandidateDict`
 - `LogprobsResultCandidateDict.log_probability`
 - `LogprobsResultCandidateDict.token`
 - `LogprobsResultCandidateDict.token_id`
- `LogprobsResultDict`
 - `LogprobsResultDict.chosen_candidates`
 - `LogprobsResultDict.top_candidates`
- `LogprobsResultTopCandidates`
 - `LogprobsResultTopCandidates.candidates`
- `LogprobsResultTopCandidatesDict`

Skip to content `resultTopCandidatesDict.candidates`

- `MaskReferenceConfig`
 - `MaskReferenceConfig.mask_dilation`
 - `MaskReferenceConfig.mask_mode`
 - `MaskReferenceConfig.segmentation_classes`
- `MaskReferenceConfigDict`
 - `MaskReferenceConfigDict.mask_dilation`
 - `MaskReferenceConfigDict.mask_mode`
 - `MaskReferenceConfigDict.segmentation_classes`
- `MaskReferenceImage`
 - `MaskReferenceImage.config`
 - `MaskReferenceImage.mask_image_config`
 - `MaskReferenceImage.reference_id`
 - `MaskReferenceImage.reference_image`
 - `MaskReferenceImage.reference_type`
- `MaskReferenceImageDict`
 - `MaskReferenceImageDict.config`
 - `MaskReferenceImageDict.reference_id`
 - `MaskReferenceImageDict.reference_image`
 - `MaskReferenceImageDict.reference_type`
- `MaskReferenceMode`
 - `MaskReferenceMode.MASK_MODE_BACKGROUND`
 - `MaskReferenceMode.MASK_MODE_DEFAULT`
 - `MaskReferenceMode.MASK_MODE_FOREGROUND`
 - `MaskReferenceMode.MASK_MODE_SEMANTIC`
 - `MaskReferenceMode.MASK_MODE_USER_PROVIDED`
- `MediaModality`
 - `MediaModality.AUDIO`
 - `MediaModality.DOCUMENT`
 - `MediaModality.IMAGE`
 - `MediaModality.MODALITY_UNSPECIFIED`
 - `MediaModality.TEXT`
 - `MediaModality.VIDEO`
- `MediaResolution`
 - `MediaResolution.MEDIA_RESOLUTION_HIGH`
 - `MediaResolution.MEDIA_RESOLUTION_LOW`
 - `MediaResolution.MEDIA_RESOLUTION_MEDIUM`
 - `MediaResolution.MEDIA_RESOLUTION_UNSPECIFIED`

[Skip to content](#)

- Modality
 - Modality.AUDIO
 - Modality.IMAGE
 - Modality.MODALITY_UNSPECIFIED
 - Modality.TEXT
- ModalityTokenCount
 - ModalityTokenCount.modality
 - ModalityTokenCount.token_count
- ModalityTokenCountDict
 - ModalityTokenCountDict.modality
 - ModalityTokenCountDict.token_count
- Mode
 - Mode.MODE_DYNAMIC
 - Mode.MODE_UNSPECIFIED
- Model
 - Model.checkpoints
 - Model.default_checkpoint_id
 - Model.description
 - Model.display_name
 - Model.endpoints
 - Model.input_token_limit
 - Model.labels
 - Model.name
 - Model.output_token_limit
 - Model.supported_actions
 - Model.tuned_model_info
 - Model.version
- ModelContent
 - ModelContent.parts
 - ModelContent.role

[Skip to content](#)

- ModelDict
 - ModelDict.checkpoints
 - ModelDict.default_checkpoint_id
 - ModelDict.description
 - ModelDict.display_name
 - ModelDict.endpoints
 - ModelDict.input_token_limit
 - ModelDict.labels
 - ModelDict.name
 - ModelDict.output_token_limit
 - ModelDict.supported_actions
 - ModelDict.tuned_model_info
 - ModelDict.version
- ModelSelectionConfig
 - ModelSelectionConfig.feature_selection_preference
- ModelSelectionConfigDict
 - ModelSelectionConfigDict.feature_selection_preference
- MultiSpeakerVoiceConfig
 - MultiSpeakerVoiceConfig.speaker_voice_configs
- MultiSpeakerVoiceConfigDict
 - MultiSpeakerVoiceConfigDict.speaker_voice_configs
- Operation
 - Operation.done
 - Operation.error
 - Operation.from_api_response()
 - Operation.metadata
 - Operation.name
- Outcome
 - Outcome.OUTCOME_DEADLINE_EXCEEDED
 - Outcome.OUTCOME_FAILED
 - Outcome.OUTCOME_OK
 - Outcome.OUTCOME_UNSPECIFIED

[Skip to content](#)

- `Part`
 - `Part.code_execution_result`
 - `Part.executable_code`
 - `Part.file_data`
 - `Part.function_call`
 - `Part.function_response`
 - `Part.inline_data`
 - `Part.text`
 - `Part.thought`
 - `Part.thought_signature`
 - `Part.video_metadata`
 - `Part.from_bytes()`
 - `Part.from_code_execution_result()`
 - `Part.from_executable_code()`
 - `Part.from_function_call()`
 - `Part.from_function_response()`
 - `Part.from_text()`
 - `Part.from_uri()`
- `PartDict`
 - `PartDict.code_execution_result`
 - `PartDict.executable_code`
 - `PartDict.file_data`
 - `PartDict.function_call`
 - `PartDict.function_response`
 - `PartDict.inline_data`
 - `PartDict.text`
 - `PartDict.thought`
 - `PartDict.thought_signature`
 - `PartDict.video_metadata`
- `PartnerModelTuningSpec`
 - `PartnerModelTuningSpec.hyper_parameters`
 - `PartnerModelTuningSpec.training_dataset_uri`
 - `PartnerModelTuningSpec.validation_dataset_uri`
- `PartnerModelTuningSpecDict`
 - `PartnerModelTuningSpecDict.hyper_parameters`
 - `PartnerModelTuningSpecDict.training_dataset_uri`
 - `PartnerModelTuningSpecDict.validation_dataset_uri`

Skip to content

- PersonGeneration
 - PersonGeneration.ALLOW_ADULT
 - PersonGeneration.ALLOW_ALL
 - PersonGeneration.DONT_ALLOW
- PrebuiltVoiceConfig
 - PrebuiltVoiceConfig.voice_name
- PrebuiltVoiceConfigDict
 - PrebuiltVoiceConfigDict.voice_name
- ProactivityConfig
 - ProactivityConfig.proactive_audio
- ProactivityConfigDict
 - ProactivityConfigDict.proactive_audio
- RagChunk
 - RagChunk.page_span
 - RagChunk.text
- RagChunkDict
 - RagChunkDict.page_span
 - RagChunkDict.text
- RagChunkPageSpan
 - RagChunkPageSpan.first_page
 - RagChunkPageSpan.last_page
- RagChunkPageSpanDict
 - RagChunkPageSpanDict.first_page
 - RagChunkPageSpanDict.last_page
- RagRetrievalConfig
 - RagRetrievalConfig.filter
 - RagRetrievalConfig.hybrid_search
 - RagRetrievalConfig.ranking
 - RagRetrievalConfig.top_k
- RagRetrievalConfigDict
 - RagRetrievalConfigDict.filter
 - RagRetrievalConfigDict.hybrid_search
 - RagRetrievalConfigDict.ranking
 - RagRetrievalConfigDict.top_k
- RagRetrievalConfigFilter
 - RagRetrievalConfigFilter.metadata_filter
 - RagRetrievalConfigFilter.vector_distance_threshold
- Skip to content `evalConfigFilter.vector_similarity_threshold`

- `RagRetrievalConfigFilterDict`
 - `RagRetrievalConfigFilterDict.metadata_filter`
 - `RagRetrievalConfigFilterDict.vector_distance_threshold`
 - `RagRetrievalConfigFilterDict.vector_similarity_threshold`
- `RagRetrievalConfigHybridSearch`
 - `RagRetrievalConfigHybridSearch.alpha`
- `RagRetrievalConfigHybridSearchDict`
 - `RagRetrievalConfigHybridSearchDict.alpha`
- `RagRetrievalConfigRanking`
 - `RagRetrievalConfigRanking.llm_ranker`
 - `RagRetrievalConfigRanking.rank_service`
- `RagRetrievalConfigRankingDict`
 - `RagRetrievalConfigRankingDict.llm_ranker`
 - `RagRetrievalConfigRankingDict.rank_service`
- `RagRetrievalConfigRankingLlmRanker`
 - `RagRetrievalConfigRankingLlmRanker.model_name`
- `RagRetrievalConfigRankingLlmRankerDict`
 - `RagRetrievalConfigRankingLlmRankerDict.model_name`
- `RagRetrievalConfigRankingRankService`
 - `RagRetrievalConfigRankingRankService.model_name`
- `RagRetrievalConfigRankingRankServiceDict`
 - `RagRetrievalConfigRankingRankServiceDict.model_name`
- `RawReferenceImage`
 - `RawReferenceImage.reference_id`
 - `RawReferenceImage.reference_image`
 - `RawReferenceImage.reference_type`
- `RawReferenceImageDict`
 - `RawReferenceImageDict.reference_id`
 - `RawReferenceImageDict.reference_image`
 - `RawReferenceImageDict.reference_type`
- `RealtimeInputConfig`
 - `RealtimeInputConfig.activity_handling`
 - `RealtimeInputConfig.automatic_activity_detection`
 - `RealtimeInputConfig.turn_coverage`
- `RealtimeInputConfigDict`
 - `RealtimeInputConfigDict.activity_handling`
 - `RealtimeInputConfigDict.automatic_activity_detection`
 - `RealtimeInputConfigDict.turn_coverage`

Skip to content `InputConfigDict.turn_coverage`

- `ReplayFile`
 - `ReplayFile.interactions`
 - `ReplayFile.replay_id`
- `ReplayFileDialog`
 - `ReplayFileDialog.interactions`
 - `ReplayFileDialog.replay_id`
- `ReplayInteraction`
 - `ReplayInteraction.request`
 - `ReplayInteraction.response`
- `ReplayInteractionDict`
 - `ReplayInteractionDict.request`
 - `ReplayInteractionDict.response`
- `ReplayRequest`
 - `ReplayRequest.body_segments`
 - `ReplayRequest.headers`
 - `ReplayRequest.method`
 - `ReplayRequest.url`
- `ReplayRequestDict`
 - `ReplayRequestDict.body_segments`
 - `ReplayRequestDict.headers`
 - `ReplayRequestDict.method`
 - `ReplayRequestDict.url`
- `ReplayResponse`
 - `ReplayResponse.body_segments`
 - `ReplayResponse.headers`
 - `ReplayResponse.sdk_response_segments`
 - `ReplayResponse.status_code`
- `ReplayResponseDict`
 - `ReplayResponseDict.body_segments`
 - `ReplayResponseDict.headers`
 - `ReplayResponseDict.sdk_response_segments`
 - `ReplayResponseDict.status_code`
- `Retrieval`
 - `Retrieval.disable_attribution`
 - `Retrieval.external_api`
 - `Retrieval.vertex_ai_search`
 - `Retrieval.vertex_rag_store`

Skip to content `nfig`

- `Config.language_code`
- `RetrievalConfig.lat_lng`

- `RetrievalConfigDict`
 - `RetrievalConfigDict.language_code`
 - `RetrievalConfigDict.lat_lng`
- `RetrievalDict`
 - `RetrievalDict.disable_attribution`
 - `RetrievalDict.external_api`
 - `RetrievalDict.vertex_ai_search`
 - `RetrievalDict.vertex_rag_store`
- `RetrievalMetadata`
 - `RetrievalMetadata.google_search_dynamic_retrieval_score`
- `RetrievalMetadataDict`
 - `RetrievalMetadataDict.google_search_dynamic_retrieval_score`
- `SafetyAttributes`
 - `SafetyAttributes.categories`
 - `SafetyAttributes.content_type`
 - `SafetyAttributes.scores`
- `SafetyAttributesDict`
 - `SafetyAttributesDict.categories`
 - `SafetyAttributesDict.content_type`
 - `SafetyAttributesDict.scores`
- `SafetyFilterLevel`
 - `SafetyFilterLevel.BLOCK_LOW_AND ABOVE`
 - `SafetyFilterLevel.BLOCK_MEDIUM_AND ABOVE`
 - `SafetyFilterLevel.BLOCK_NONE`
 - `SafetyFilterLevel.BLOCK_ONLY_HIGH`
- `SafetyRating`
 - `SafetyRating.blocked`
 - `SafetyRating.category`
 - `SafetyRating.overwritten_threshold`
 - `SafetyRating.probability`
 - `SafetyRating.probability_score`
 - `SafetyRating.severity`
 - `SafetyRating.severity_score`

[Skip to content](#)

- `SafetyRatingDict`
 - `SafetyRatingDict.blocked`
 - `SafetyRatingDict.category`
 - `SafetyRatingDict.overwritten_threshold`
 - `SafetyRatingDict.probability`
 - `SafetyRatingDict.probability_score`
 - `SafetyRatingDict.severity`
 - `SafetyRatingDict.severity_score`
- `SafetySetting`
 - `SafetySetting.category`
 - `SafetySetting.method`
 - `SafetySetting.threshold`
- `SafetySettingDict`
 - `SafetySettingDict.category`
 - `SafetySettingDict.method`
 - `SafetySettingDict.threshold`
- `Scale`
 - `Scale.A_FLAT_MAJOR_F_MINOR`
 - `Scale.A_MAJOR_G_FLAT_MINOR`
 - `Scale.B_FLAT_MAJOR_G_MINOR`
 - `Scale.B_MAJOR_A_FLAT_MINOR`
 - `Scale.C_MAJOR_A_MINOR`
 - `Scale.D_FLAT_MAJOR_B_FLAT_MINOR`
 - `Scale.D_MAJOR_B_MINOR`
 - `Scale.E_FLAT_MAJOR_C_MINOR`
 - `Scale.E_MAJOR_D_FLAT_MINOR`
 - `Scale.F_MAJOR_D_MINOR`
 - `Scale.G_FLAT_MAJOR_E_FLAT_MINOR`
 - `Scale.G_MAJOR_E_MINOR`
 - `Scale.SCALE_UNSPECIFIED`

[Skip to content](#)

- Schema

- Schema.additional_properties
- Schema.any_of
- Schema.default
- Schema.defs
- Schema.description
- Schema.enum
- Schema.example
- Schema.format
- Schema.items
- Schema.max_items
- Schema.max_length
- Schema.max_properties
- Schema.maximum
- Schema.min_items
- Schema.min_length
- Schema.min_properties
- Schema.minimum
- Schema.nullable
- Schema.pattern
- Schema.properties
- Schema.property_ordering
- Schema.ref
- Schema.required
- Schema.title
- Schema.type
- Schema.from_json_schema()
- Schema.json_schema

[Skip to content](#)

- SchemaDict
 - SchemaDict.additional_properties
 - SchemaDict.any_of
 - SchemaDict.default
 - SchemaDict.defs
 - SchemaDict.description
 - SchemaDict.enum
 - SchemaDict.example
 - SchemaDict.format
 - SchemaDict.max_items
 - SchemaDict.max_length
 - SchemaDict.max_properties
 - SchemaDict.maximum
 - SchemaDict.min_items
 - SchemaDict.min_length
 - SchemaDict.min_properties
 - SchemaDict.minimum
 - SchemaDict.nullable
 - SchemaDict.pattern
 - SchemaDict.properties
 - SchemaDict.property_ordering
 - SchemaDict.ref
 - SchemaDict.required
 - SchemaDict.title
 - SchemaDict.type
- SearchEntryPoint
 - SearchEntryPoint.rendered_content
 - SearchEntryPoint.sdk_blob
- SearchEntryPointDict
 - SearchEntryPointDict.rendered_content
 - SearchEntryPointDict.sdk_blob
- Segment
 - Segment.end_index
 - Segment.part_index
 - Segment.start_index
 - Segment.text

[Skip to content](#)

- SegmentDict
 - SegmentDict.end_index
 - SegmentDict.part_index
 - SegmentDict.start_index
 - SegmentDict.text
- SessionResumptionConfig
 - SessionResumptionConfig.handle
 - SessionResumptionConfig.transparent
- SessionResumptionConfigDict
 - SessionResumptionConfigDict.handle
 - SessionResumptionConfigDict.transparent
- SlidingWindow
 - SlidingWindow.target_tokens
- SlidingWindowDict
 - SlidingWindowDict.target_tokens
- SpeakerVoiceConfig
 - SpeakerVoiceConfig.speaker
 - SpeakerVoiceConfig.voice_config
- SpeakerVoiceConfigDict
 - SpeakerVoiceConfigDict.speaker
 - SpeakerVoiceConfigDict.voice_config
- SpeechConfig
 - SpeechConfig.language_code
 - SpeechConfig.multi_speaker_voice_config
 - SpeechConfig.voice_config
- SpeechConfigDict
 - SpeechConfigDict.language_code
 - SpeechConfigDict.multi_speaker_voice_config
 - SpeechConfigDict.voice_config
- StartSensitivity
 - StartSensitivity.START_SENSITIVITY_HIGH
 - StartSensitivity.START_SENSITIVITY_LOW
 - StartSensitivity.START_SENSITIVITY_UNSPECIFIED
- StyleReferenceConfig
 - StyleReferenceConfig.style_description
- StyleReferenceConfigDict
 - StyleReferenceConfigDict.style_description

[Skip to content](#)

- StyleReferenceImage
 - StyleReferenceImage.config
 - StyleReferenceImage.reference_id
 - StyleReferenceImage.reference_image
 - StyleReferenceImage.reference_type
 - StyleReferenceImage.style_image_config
- StyleReferenceImageDict
 - StyleReferenceImageDict.config
 - StyleReferenceImageDict.reference_id
 - StyleReferenceImageDict.reference_image
 - StyleReferenceImageDict.reference_type
- SubjectReferenceConfig
 - SubjectReferenceConfig.subject_description
 - SubjectReferenceConfig.subject_type
- SubjectReferenceConfigDict
 - SubjectReferenceConfigDict.subject_description
 - SubjectReferenceConfigDict.subject_type
- SubjectReferenceImage
 - SubjectReferenceImage.config
 - SubjectReferenceImage.reference_id
 - SubjectReferenceImage.reference_image
 - SubjectReferenceImage.reference_type
 - SubjectReferenceImage.subject_image_config
- SubjectReferenceImageDict
 - SubjectReferenceImageDict.config
 - SubjectReferenceImageDict.reference_id
 - SubjectReferenceImageDict.reference_image
 - SubjectReferenceImageDict.reference_type
- SubjectReferenceType
 - SubjectReferenceType.SUBJECT_TYPE_ANIMAL
 - SubjectReferenceType.SUBJECT_TYPE_DEFAULT
 - SubjectReferenceType.SUBJECT_TYPE_PERSON
 - SubjectReferenceType.SUBJECT_TYPE_PRODUCT
- SupervisedHyperParameters
 - SupervisedHyperParameters.adapter_size
 - SupervisedHyperParameters.epoch_count
 - SupervisedHyperParameters.learning_rate_multiplier

[Skip to content](#)

- `SupervisedHyperParametersDict`
 - `SupervisedHyperParametersDict.adapter_size`
 - `SupervisedHyperParametersDict.epoch_count`
 - `SupervisedHyperParametersDict.learning_rate_multiplier`
- `SupervisedTuningDataStats`
 - `SupervisedTuningDataStats.dropped_example_reasons`
 - `SupervisedTuningDataStats.total_billable_character_count`
 - `SupervisedTuningDataStats.total_billable_token_count`
 - `SupervisedTuningDataStats.total_truncated_example_count`
 - `SupervisedTuningDataStats.total_tuning_character_count`
 - `SupervisedTuningDataStats.truncated_example_indices`
 - `SupervisedTuningDataStats.tuning_dataset_example_count`
 - `SupervisedTuningDataStats.tuning_step_count`
 - `SupervisedTuningDataStats.user_dataset_examples`
 - `SupervisedTuningDataStats.user_input_token_distribution`
 - `SupervisedTuningDataStats.user_message_per_example_distribution`
 - `SupervisedTuningDataStats.user_output_token_distribution`
- `SupervisedTuningDataStatsDict`
 - `SupervisedTuningDataStatsDict.dropped_example_reasons`
 - `SupervisedTuningDataStatsDict.total_billable_character_count`
 - `SupervisedTuningDataStatsDict.total_billable_token_count`
 - `SupervisedTuningDataStatsDict.total_truncated_example_count`
 - `SupervisedTuningDataStatsDict.total_tuning_character_count`
 - `SupervisedTuningDataStatsDict.truncated_example_indices`
 - `SupervisedTuningDataStatsDict.tuning_dataset_example_count`
 - `SupervisedTuningDataStatsDict.tuning_step_count`
 - `SupervisedTuningDataStatsDict.user_dataset_examples`
 - `SupervisedTuningDataStatsDict.user_input_token_distribution`
 - `SupervisedTuningDataStatsDict.user_message_per_example_distribution`
 - `SupervisedTuningDataStatsDict.user_output_token_distribution`
- `SupervisedTuningDatasetDistribution`
 - `SupervisedTuningDatasetDistribution.billable_sum`
 - `SupervisedTuningDatasetDistribution.buckets`
 - `SupervisedTuningDatasetDistribution.max`
 - `SupervisedTuningDatasetDistribution.mean`
 - `SupervisedTuningDatasetDistribution.median`
 - `SupervisedTuningDatasetDistribution.min`

Skip to content `edTuningDatasetDistribution.p5`

`edTuningDatasetDistribution.p95`

▪ `SupervisedTuningDatasetDistribution.sum`

- SupervisedTuningDatasetDistributionDatasetBucket
 - SupervisedTuningDatasetDistributionDatasetBucket.count
 - SupervisedTuningDatasetDistributionDatasetBucket.left
 - SupervisedTuningDatasetDistributionDatasetBucket.right
- SupervisedTuningDatasetDistributionDatasetBucketDict
 - SupervisedTuningDatasetDistributionDatasetBucketDict.count
 - SupervisedTuningDatasetDistributionDatasetBucketDict.left
 - SupervisedTuningDatasetDistributionDatasetBucketDict.right
- SupervisedTuningDatasetDistributionDict
 - SupervisedTuningDatasetDistributionDict.billable_sum
 - SupervisedTuningDatasetDistributionDict.buckets
 - SupervisedTuningDatasetDistributionDict.max
 - SupervisedTuningDatasetDistributionDict.mean
 - SupervisedTuningDatasetDistributionDict.median
 - SupervisedTuningDatasetDistributionDict.min
 - SupervisedTuningDatasetDistributionDict.p5
 - SupervisedTuningDatasetDistributionDict.p95
 - SupervisedTuningDatasetDistributionDict.sum
- SupervisedTuningSpec
 - SupervisedTuningSpec.export_last_checkpoint_only
 - SupervisedTuningSpec.hyper_parameters
 - SupervisedTuningSpec.training_dataset_uri
 - SupervisedTuningSpec.validation_dataset_uri
- SupervisedTuningSpecDict
 - SupervisedTuningSpecDict.export_last_checkpoint_only
 - SupervisedTuningSpecDict.hyper_parameters
 - SupervisedTuningSpecDict.training_dataset_uri
 - SupervisedTuningSpecDict.validation_dataset_uri
- TestTableFile
 - TestTableFile.comment
 - TestTableFile.parameter_names
 - TestTableFile.test_method
 - TestTableFile.test_table
- TestTableFileDict
 - TestTableFileDict.comment
 - TestTableFileDict.parameter_names
 - TestTableFileDict.test_method

Skip to content [FileDict.test_table](#)

- `TestTableItem`
 - `TestTableItem.exception_if_mldev`
 - `TestTableItem.exception_if_vertex`
 - `TestTableItem.has_union`
 - `TestTableItem.ignore_keys`
 - `TestTableItem.name`
 - `TestTableItem.override_replay_id`
 - `TestTableItem.parameters`
 - `TestTableItem.skip_in_api_mode`
- `TestTableItemDict`
 - `TestTableItemDict.exception_if_mldev`
 - `TestTableItemDict.exception_if_vertex`
 - `TestTableItemDict.has_union`
 - `TestTableItemDict.ignore_keys`
 - `TestTableItemDict.name`
 - `TestTableItemDict.override_replay_id`
 - `TestTableItemDict.parameters`
 - `TestTableItemDict.skip_in_api_mode`
- `ThinkingConfig`
 - `ThinkingConfig.include_thoughts`
 - `ThinkingConfig.thinking_budget`
- `ThinkingConfigDict`
 - `ThinkingConfigDict.include_thoughts`
 - `ThinkingConfigDict.thinking_budget`
- `TokensInfo`
 - `TokensInfo.role`
 - `TokensInfo.token_ids`
 - `TokensInfo.tokens`
- `TokensInfoDict`
 - `TokensInfoDict.role`
 - `TokensInfoDict.token_ids`
 - `TokensInfoDict.tokens`

[Skip to content](#)

- Tool
 - Tool.code_execution
 - Tool.computer_use
 - Tool.enterprise_web_search
 - Tool.function_declarations
 - Tool.google_maps
 - Tool.google_search
 - Tool.google_search_retrieval
 - Tool.retrieval
 - Tool.url_context
- ToolCodeExecution
- ToolCodeExecutionDict
- ToolComputerUse
 - ToolComputerUse.environment
- ToolComputerUseDict
 - ToolComputerUseDict.environment
- ToolConfig
 - ToolConfig.function_calling_config
 - ToolConfig.retrieval_config
- ToolConfigDict
 - ToolConfigDict.function_calling_config
 - ToolConfigDict.retrieval_config
- ToolDict
 - ToolDict.code_execution
 - ToolDict.computer_use
 - ToolDict.enterprise_web_search
 - ToolDict.function_declarations
 - ToolDict.google_maps
 - ToolDict.google_search
 - ToolDict.google_search_retrieval
 - ToolDict.retrieval
 - ToolDict.url_context
- TrafficType
 - TrafficType.ON_DEMAND
 - TrafficType.PROVISIONED_THROUGHPUT
 - TrafficType.TRAFFIC_TYPE_UNSPECIFIED
- Transcription

Skip to content [option.finished](#)
[option.text](#)

- `TranscriptionDict`
 - `TranscriptionDict.finished`
 - `TranscriptionDict.text`
- `TunedModel`
 - `TunedModel.checkpoints`
 - `TunedModel.endpoint`
 - `TunedModel.model`
- `TunedModelCheckpoint`
 - `TunedModelCheckpoint.checkpoint_id`
 - `TunedModelCheckpoint.endpoint`
 - `TunedModelCheckpoint.epoch`
 - `TunedModelCheckpoint.step`
- `TunedModelCheckpointDict`
 - `TunedModelCheckpointDict.checkpoint_id`
 - `TunedModelCheckpointDict.endpoint`
 - `TunedModelCheckpointDict.epoch`
 - `TunedModelCheckpointDict.step`
- `TunedModelDict`
 - `TunedModelDict.checkpoints`
 - `TunedModelDict.endpoint`
 - `TunedModelDict.model`
- `TunedModelInfo`
 - `TunedModelInfo.base_model`
 - `TunedModelInfo.create_time`
 - `TunedModelInfo.update_time`
- `TunedModelInfoDict`
 - `TunedModelInfoDict.base_model`
 - `TunedModelInfoDict.create_time`
 - `TunedModelInfoDict.update_time`
- `TuningDataStats`
 - `TuningDataStats.distillation_data_stats`
 - `TuningDataStats.supervised_tuning_data_stats`
- `TuningDataStatsDict`
 - `TuningDataStatsDict.distillation_data_stats`
 - `TuningDataStatsDict.supervised_tuning_data_stats`
- `TuningDataset`
 - `TuningDataset.examples`

Skip to content `:asset.gcs_uri`
`:asset.vertex_dataset_resource`

- TuningDatasetDict
 - TuningDatasetDict.examples
 - TuningDatasetDict.gcs_uri
 - TuningDatasetDict.vertex_dataset_resource
- TuningExample
 - TuningExample.output
 - TuningExample.text_input
- TuningExampleDict
 - TuningExampleDict.output
 - TuningExampleDict.text_input
- TuningJob
 - TuningJob.base_model
 - TuningJob.create_time
 - TuningJob.description
 - TuningJob.distillation_spec
 - TuningJob.encryption_spec
 - TuningJob.end_time
 - TuningJob.error
 - TuningJob.experiment
 - TuningJob.labels
 - TuningJob.name
 - TuningJob.partner_model_tuning_spec
 - TuningJob.pipeline_job
 - TuningJob.satisfies_pzi
 - TuningJob.satisfies_pzs
 - TuningJob.service_account
 - TuningJob.start_time
 - TuningJob.state
 - TuningJob.supervised_tuning_spec
 - TuningJob.tuned_model
 - TuningJob.tuned_model_display_name
 - TuningJob.tuning_data_stats
 - TuningJob.update_time
 - TuningJob.has-ended
 - TuningJob.has_succeeded

[Skip to content](#)

- TuningJobDict
 - TuningJobDict.base_model
 - TuningJobDict.create_time
 - TuningJobDict.description
 - TuningJobDict.distillation_spec
 - TuningJobDict.encryption_spec
 - TuningJobDict.end_time
 - TuningJobDict.error
 - TuningJobDict.experiment
 - TuningJobDict.labels
 - TuningJobDict.name
 - TuningJobDict.partner_model_tuning_spec
 - TuningJobDict.pipeline_job
 - TuningJobDict.satisfies_pzi
 - TuningJobDict.satisfies_pzs
 - TuningJobDict.service_account
 - TuningJobDict.start_time
 - TuningJobDict.state
 - TuningJobDict.supervised_tuning_spec
 - TuningJobDict.tuned_model
 - TuningJobDict.tuned_model_display_name
 - TuningJobDict.tuning_data_stats
 - TuningJobDict.update_time
- TuningOperation
 - TuningOperation.done
 - TuningOperation.error
 - TuningOperation.metadata
 - TuningOperation.name
- TuningOperationDict
 - TuningOperationDict.done
 - TuningOperationDict.error
 - TuningOperationDict.metadata
 - TuningOperationDict.name
- TuningValidationDataset
 - TuningValidationDataset.gcs_uri
 - TuningValidationDataset.vertex_dataset_resource
- TuningValidationDatasetDict
 - TuningValidationDatasetDict.gcs_uri
 - TuningValidationDatasetDict.vertex_dataset_resource

Skip to content

TuningValidationDatasetDict.gcs_uri

TuningValidationDatasetDict.vertex_dataset_resource

- TurnCoverage
 - TurnCoverage.TURN_COVERAGE_UNSPECIFIED
 - TurnCoverage.TURN_INCLUDES_ALL_INPUT
 - TurnCoverage.TURN_INCLUDES_ONLY_ACTIVITY
- Type
 - Type.ARRAY
 - Type.BOOLEAN
 - Type.INTEGER
 - Type.NULL
 - Type.NUMBER
 - Type.OBJECT
 - Type.STRING
 - Type.TYPE_UNSPECIFIED
- UpdateCachedContentConfig
 - UpdateCachedContentConfig.expire_time
 - UpdateCachedContentConfig.http_options
 - UpdateCachedContentConfig.ttl
- UpdateCachedContentConfigDict
 - UpdateCachedContentConfigDict.expire_time
 - UpdateCachedContentConfigDict.http_options
 - UpdateCachedContentConfigDict.ttl
- UpdateModelConfig
 - UpdateModelConfig.default_checkpoint_id
 - UpdateModelConfig.description
 - UpdateModelConfig.display_name
 - UpdateModelConfig.http_options
- UpdateModelConfigDict
 - UpdateModelConfigDict.default_checkpoint_id
 - UpdateModelConfigDict.description
 - UpdateModelConfigDict.display_name
 - UpdateModelConfigDict.http_options
- UploadFileConfig
 - UploadFileConfig.display_name
 - UploadFileConfig.http_options
 - UploadFileConfig.mime_type
 - UploadFileConfig.name

[Skip to content](#)

- UploadFileConfigDict
 - UploadFileConfigDict.display_name
 - UploadFileConfigDict.http_options
 - UploadFileConfigDict.mime_type
 - UploadFileConfigDict.name
- UpscaleImageConfig
 - UpscaleImageConfig.enhance_input_image
 - UpscaleImageConfig.http_options
 - UpscaleImageConfig.image_preservation_factor
 - UpscaleImageConfig.include_rai_reason
 - UpscaleImageConfig.output_compression_quality
 - UpscaleImageConfig.output_mime_type
- UpscaleImageConfigDict
 - UpscaleImageConfigDict.enhance_input_image
 - UpscaleImageConfigDict.http_options
 - UpscaleImageConfigDict.image_preservation_factor
 - UpscaleImageConfigDict.include_rai_reason
 - UpscaleImageConfigDict.output_compression_quality
 - UpscaleImageConfigDict.output_mime_type
- UpscaleImageParameters
 - UpscaleImageParameters.config
 - UpscaleImageParameters.image
 - UpscaleImageParameters.model
 - UpscaleImageParameters.upscale_factor
- UpscaleImageParametersDict
 - UpscaleImageParametersDict.config
 - UpscaleImageParametersDict.image
 - UpscaleImageParametersDict.model
 - UpscaleImageParametersDict.upscale_factor
- UpscaleImageResponse
 - UpscaleImageResponse.generated_images
- UpscaleImageResponseDict
 - UpscaleImageResponseDict.generated_images
- UrlContext
- UrlContextDict
- UrlContextMetadata
 - UrlContextMetadata.url_metadata

Skip to content etadataDict
 xtMetadataDict.url_metadata

- UrlMetadata
 - UrlMetadata.retrieved_url
 - UrlMetadata.url_retrieval_status
- UrlMetadataDict
 - UrlMetadataDict.retrieved_url
 - UrlMetadataDict.url_retrieval_status
- UrlRetrievalStatus
 - UrlRetrievalStatus.URL_RETRIEVAL_STATUS_ERROR
 - UrlRetrievalStatus.URL_RETRIEVAL_STATUS_SUCCESS
 - UrlRetrievalStatus.URL_RETRIEVAL_STATUS_UNSPECIFIED
- UsageMetadata
 - UsageMetadata.cache_tokens_details
 - UsageMetadata.cached_content_token_count
 - UsageMetadata.prompt_token_count
 - UsageMetadata.prompt_tokens_details
 - UsageMetadata.response_token_count
 - UsageMetadata.response_tokens_details
 - UsageMetadata.thoughts_token_count
 - UsageMetadata.tool_use_prompt_token_count
 - UsageMetadata.tool_use_prompt_tokens_details
 - UsageMetadata.total_token_count
 - UsageMetadata.traffic_type
- UsageMetadataDict
 - UsageMetadataDict.cache_tokens_details
 - UsageMetadataDict.cached_content_token_count
 - UsageMetadataDict.prompt_token_count
 - UsageMetadataDict.prompt_tokens_details
 - UsageMetadataDict.response_token_count
 - UsageMetadataDict.response_tokens_details
 - UsageMetadataDict.thoughts_token_count
 - UsageMetadataDict.tool_use_prompt_token_count
 - UsageMetadataDict.tool_use_prompt_tokens_details
 - UsageMetadataDict.total_token_count
 - UsageMetadataDict.traffic_type
- UserContent
 - UserContent.parts
 - UserContent.role

[Skip to content](#)

- `VertexAISeach`
 - `VertexAISeach.data_store_specs`
 - `VertexAISeach.datastore`
 - `VertexAISeach.engine`
 - `VertexAISeach.filter`
 - `VertexAISeach.max_results`
- `VertexAISeachDataStoreSpec`
 - `VertexAISeachDataStoreSpec.data_store`
 - `VertexAISeachDataStoreSpec.filter`
- `VertexAISeachDataStoreSpecDict`
 - `VertexAISeachDataStoreSpecDict.data_store`
 - `VertexAISeachDataStoreSpecDict.filter`
- `VertexAISeachDict`
 - `VertexAISeachDict.data_store_specs`
 - `VertexAISeachDict.datastore`
 - `VertexAISeachDict.engine`
 - `VertexAISeachDict.filter`
 - `VertexAISeachDict.max_results`
- `VertexRagStore`
 - `VertexRagStore.rag_corpora`
 - `VertexRagStore.rag_resources`
 - `VertexRagStore.rag_retrieval_config`
 - `VertexRagStore.similarity_top_k`
 - `VertexRagStore.store_context`
 - `VertexRagStore.vector_distance_threshold`
- `VertexRagStoreDict`
 - `VertexRagStoreDict.rag_corpora`
 - `VertexRagStoreDict.rag_resources`
 - `VertexRagStoreDict.rag_retrieval_config`
 - `VertexRagStoreDict.similarity_top_k`
 - `VertexRagStoreDict.store_context`
 - `VertexRagStoreDict.vector_distance_threshold`
- `VertexRagStoreRagResource`
 - `VertexRagStoreRagResource.rag_corpus`
 - `VertexRagStoreRagResource.rag_file_ids`
- `VertexRagStoreRagResourceDict`
 - `VertexRagStoreRagResourceDict.rag_corpus`
 - `VertexRagStoreRagResourceDict.rag_file_ids`

Skip to content `)StoreRagResourceDict.rag_file_ids`

- Video
 - `Video.mime_type`
 - `Video.uri`
 - `Video.video_bytes`
 - `Video.from_file()`
 - `Video.save()`
 - `Video.show()`
- VideoCompressionQuality
 - `VideoCompressionQuality.LOSSLESS`
 - `VideoCompressionQuality.OPTIMIZED`
- VideoDict
 - `VideoDict.mime_type`
 - `VideoDict.uri`
 - `VideoDict.video_bytes`
- VideoMetadata
 - `VideoMetadata.end_offset`
 - `VideoMetadata.fps`
 - `VideoMetadata.start_offset`
- VideoMetadataDict
 - `VideoMetadataDict.end_offset`
 - `VideoMetadataDict.fps`
 - `VideoMetadataDict.start_offset`
- VoiceConfig
 - `VoiceConfig.prebuilt_voice_config`
- VoiceConfigDict
 - `VoiceConfigDict.prebuilt_voice_config`
- WeightedPrompt
 - `WeightedPrompt.text`
 - `WeightedPrompt.weight`
- WeightedPromptDict
 - `WeightedPromptDict.text`
 - `WeightedPromptDict.weight`