The Interactions API (/gemini-api/docs/interactions-overview) is now generally available. We recommend using this API for access to all the latest features and models.

Models

The models endpoint provides a way for you to programmatically list the available models, and retrieve extended metadata such as supported functionality and context window sizing. Read more in the Models guide (https://ai.google.dev/gemini-api/docs/models/gemini).

Method: models.get

Gets information about a specific `Model` such as its version number, token limits, parameters (https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters) and other metadata. Refer to the Gemini models guide (https://ai.google.dev/gemini-api/docs/models/gemini) for detailed model information.

Endpoint

```
GET https://generativelanguage.googleapis.com/v1beta/{name=models/*}
```

Path parameters

`name` `string`

Required. The resource name of the model.
This name should match a model name returned by the `models.list` method.
Format: `models/{model}` It takes the form `models/{model}`.

Request body

The request body must be empty.

Example request

PythonGo (#go)Shell (#shell)
(#python)

```
from google import genai

client = genai.Client()
model_info = client.models.get(model="gemini-3.5-flash")
print(model_info)
```

es/blob/42d4b937cd119db8543b9dc7d5cf22b783306d29/python/models.py#L41-L45)

Response body

If successful, the response body contains an instance of `Model` (/api/models#Model).

Method: models.list

Lists the `Models` (https://ai.google.dev/gemini-api/docs/models/gemini) available through the Gemini API.

Endpoint

```
GET https://generativelanguage.googleapis.com/v1beta/models
```

Query parameters

`pageSize` `integer`

The maximum number of `Models` to return (per page).
If unspecified, 50 models will be returned per page. This method returns at most 1000 models per page, even if you pass a larger pageSize.

`pageToken` `string`

A page token, received from a previous `models.list` call.
Provide the `pageToken` returned by one request as an argument to the next request to retrieve the next page.
When paginating, all other parameters provided to `models.list` must match the call that provided the page token.

Request body

The request body must be empty.

Example request

PythonGo (#go)Shell (#shell)
(#python)

```
from google import genai

client = genai.Client()

print("List of models that support generateContent:\n")
for m in client.models.list():
    for action in m.supported_actions:
        if action == "generateContent":
            print(m.name)

print("List of models that support embedContent:\n")
for m in client.models.list():
    for action in m.supported_actions:
        if action == "embedContent":
            print(m.name)
```

es/blob/42d4b937cd119db8543b9dc7d5cf22b783306d29/python/models.py#L22-L36)

Response body

Response from `ListModel` containing a paginated list of Models.

If successful, the response body contains data with the following structure:

Fields

`models[]` `object (Model (/api/models#Model))`

The returned Models.

`nextPageToken` `string`

A token, which can be sent as `pageToken` to retrieve the next page.
If this field is omitted, there are no more pages.

JSON representation

JSON representation

```
{
  "models": [
    {
      object (Model (/api/models#Model))
    }
  ],
  "nextPageToken": string
}
```

REST Resource: models

Resource: Model

Information about a Generative Language Model.

Fields

`name` `string`

Required. The resource name of the `Model`. Refer to Model variants (https://ai.google.dev/gemini-api/docs/models/gemini#model-variations) for all allowed values.
Format: `models/{model}` with a `{model}` naming convention of:

- "{baseModelId}-{version}"

Examples:

- `models/gemini-1.5-flash-001`

`baseModelId` `string`

Required. The name of the base model, pass this to the generation request.
Examples:

- `gemini-1.5-flash`

`version` `string`

Required. The version number of the model.
This represents the major version (`1.0` or `1.5`)

`displayName` `string`

The human-readable name of the model. E.g. "Gemini 1.5 Flash".
The name can be up to 128 characters long and can consist of any UTF-8 characters.

`description` `string`

A short description of the model.

`inputTokenLimit` `integer`

Maximum number of input tokens allowed for this model.

`outputTokenLimit` `integer`

Maximum number of output tokens available for this model.

`supportedGenerationMethods[]` `string`

The model's supported generation methods.
The corresponding API method names are defined as Pascal case strings, such as `generateMessage` and `generateContent`.

`thinking` `boolean`

Whether the model supports thinking.

`temperature` `number`

Controls the randomness of the output.
Values can range over `[0.0,maxTemperature]`, inclusive. A higher value will produce responses that are more varied, while a value closer to `0.0` will typically result in less surprising responses from the model. This value specifies default to be used by the backend while making the call to the model.

`maxTemperature` `number`

The maximum temperature this model can use.

`topP` `number`

For Nucleus sampling (https://ai.google.dev/gemini-api/docs/prompting-strategies#top-p).
Nucleus sampling considers the smallest set of tokens whose probability sum is at least `topP`. This value specifies default to be used by the backend while making the call to the model.

`topK` `integer`

For Top-k sampling.
Top-k sampling considers the set of `topK` most probable tokens. This value specifies default to be used by the backend while making the call to the model. If empty, indicates the model doesn't use top-k sampling, and `topK` isn't allowed as a generation parameter.

JSON representation

```
{
  "name": string,
  "baseModelId": string,
  "version": string,
  "displayName": string,
  "description": string,
  "inputTokenLimit": integer,
  "outputTokenLimit": integer,
  "supportedGenerationMethods": [
    string
  ],
  "thinking": boolean,
  "temperature": number,
  "maxTemperature": number,
  "topP": number,
  "topK": integer
}
```

Method: models.predict

Performs a prediction request.

Endpoint

`POST https://generativelanguage.googleapis.com/v1beta/{model=models/*}:predict`

Path parameters

`model` `string`

Required. The name of the model for prediction. Format: `name=models/{model}`. It takes the form `models/{model}`.

Request body

The request body contains data with the following structure:

Fields

instances[] `value` (Value (https://protobuf.dev/reference/protobuf/google.protobuf/#value) `format`)

Required. The instances that are the input to the prediction call.

parameters `value` (Value (https://protobuf.dev/reference/protobuf/google.protobuf/#value) `format`)

Optional. The parameters that govern the prediction call.

Response body

Response message for [PredictionService.Predict].

If successful, the response body contains data with the following structure:

Fields

predictions[] `value` (Value (https://protobuf.dev/reference/protobuf/google.protobuf/#value) `format`)

The outputs of the prediction call.

JSON representation

```
{
  "predictions": [
    value
  ]
}
```

Method: models.predictLongRunning

Same as models.predict but returns an LRO.

Endpoint

```
POST https://generativelanguage.googleapis.com/v1beta/{model=models/*}:predictLong
     Running
```

Path parameters

`model` `string`

Required. The name of the model for prediction. Format: `name=models/{model}`.

Request body

The request body contains data with the following structure:

Fields

`instances[]` `value` (`Value` (https://protobuf.dev/reference/protobuf/google.protobuf/#value) `format`)

Required. The instances that are the input to the prediction call.

`parameters` `value` (`Value` (https://protobuf.dev/reference/protobuf/google.protobuf/#value) `format`)

Optional. The parameters that govern the prediction call.

Response body

If successful, the response body contains an instance of `Operation` (/api/batch-api#Operation).