

PanguLM Service

API Reference

Issue	01
Date	2025-08-19



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

1 Before You Start.....**1**

1.1 Overview..... 1

1.2 API Calling.....2

1.3 Request URI..... 2

1.4 Concepts..... 5

2 Calling REST APIs.....**6**

2.1 Making an API Request..... 6

2.2 Authentication.....8

2.3 Response..... 12

3 API.....**14**

3.1 Model Inference APIs..... 14

3.1.1 Third-Party Models..... 14

3.1.1.1 DeepSeek..... 14

3.2 Data Engineering APIs.....31

3.2.1 Querying Data Lineages.....31

3.2.2 Permanently Deleting a Dataset..... 33

3.3 Agent APIs..... 37

3.3.1 Calling an Application..... 37

3.3.2 Calling a Workflow..... 43

3.4 Token Calculator..... 51

4 Appendix.....**54**

4.1 Status Codes..... 54

4.2 Error Codes.....58

4.3 Obtaining the Project ID.....63

4.4 Obtaining the Model Deployment ID..... 65

Issue 01 (2025-08-19)

Copyright © Huawei Cloud Computing Technologies Co., Ltd.

ii

1 Before You Start

1.1 Overview

ModelArts Studio supports the management and deployment of Pangu models and third-party models. Models deployed on ModelArts Studio can be accessed through inference APIs.

Table 1-1 APIs

Category	Model	API	Function
Model Inference APIs	Third-party models	DeepSeek	DeepSeek API is an API service based on the DeepSeek model. It supports text-based interaction in multiple scenarios and can quickly generate high-quality dialogs, copywriting passages, and stories. It is suitable for scenarios such as text summarization, intelligent Q&A, and content creation.
Data engineering APIs	-	Querying Data Lineages	Allows you to import raw datasets from OBS. You can use the OBS path to query all raw datasets created based on the path and the subsequent lineage information.
	-	Permanently Deleting a Dataset	For data uploaded from OBS, you need to delete the associated raw data in OBS when deleting the datasets. Instead of retaining raw data in OBS for a long time, customers require that the raw data be archived in their big data center.
Agent APIs	-	Calling an Application	Allows you to call the API of a created application and input a question to obtain the application execution result.

Category	Model	API	Function
	-	Calling a Workflow	Allows you to call the API of a created workflow and input a question to obtain the workflow execution result.
Token calculator	-	Token Calculator	To help you manage tokens more effectively and optimize token usage, the platform provides a token calculator. The token calculator evaluates the number of tokens in the specified text before model inference, estimate the cost, and optimize the data preprocessing policy.

 **NOTE**

It is recommended that you enable the security guardrail function during service deployment to ensure content security.

1.2 API Calling

PanguLM provides a broad range of Representational State Transfer (REST) APIs that you can call through HTTPS. For details about API calling, see [Calling REST APIs](#).

Before calling an API, ensure that your network can communicate with the Internet.

1.3 Request URI

A request URI of a service is the endpoint for calling an API. The endpoint is used to communicate and interact with the API.

To obtain the URI, perform the following steps:

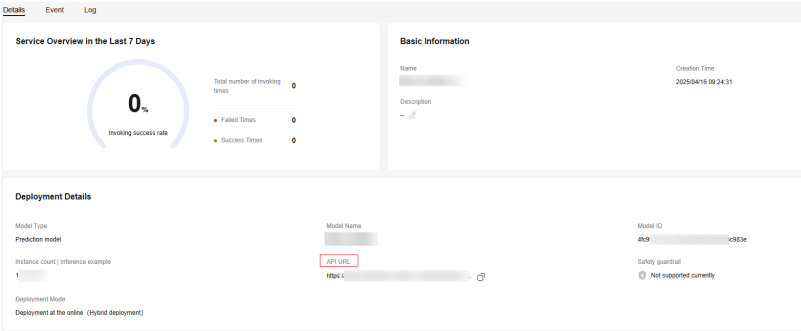
Step 1 Log in to ModelArts Studio.

Step 2 Go to the target workspace.

Step 3 Obtain the request URI.

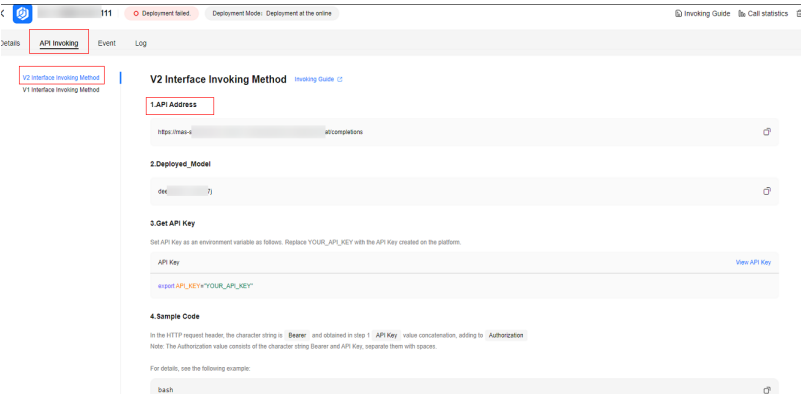
- Obtain the request URI of the model.
 - To call a deployed model, choose **Model Development > Model Deployment** in the navigation pane on the left. On the **My Service** tab page, click the model name in the model deployment list. On the **Details** tab page, obtain the request URI of the model.

Figure 1-1 Calling a deployed model



To call the inference service you deployed, obtain the URL of the V1 API or the URI of the V2 API on the **API Invoking** page.

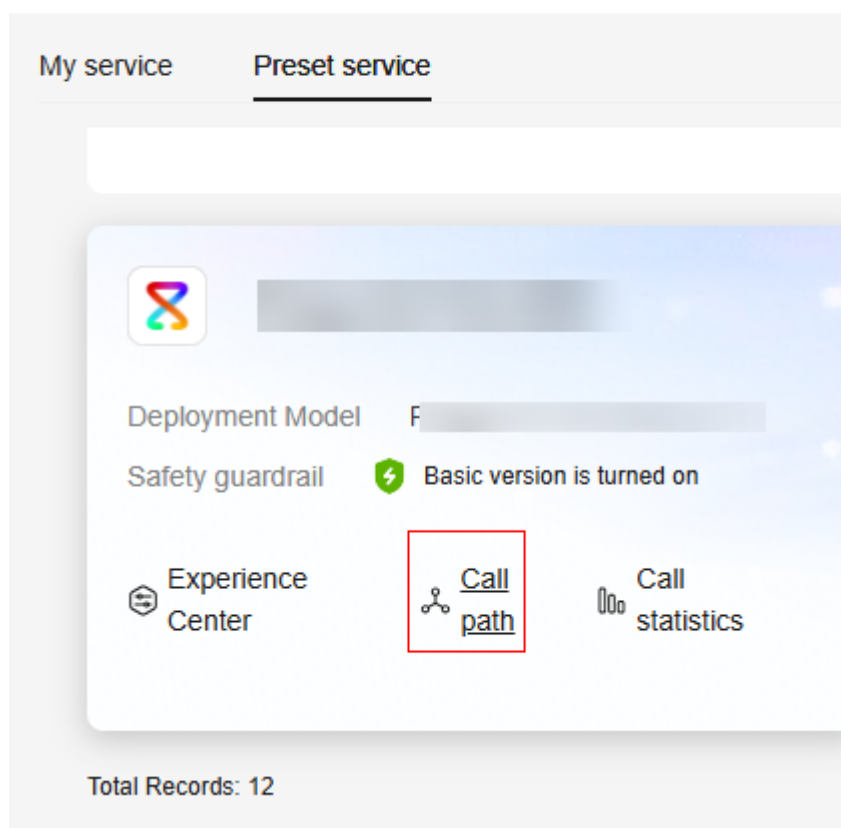
Figure 1-2 Calling an service



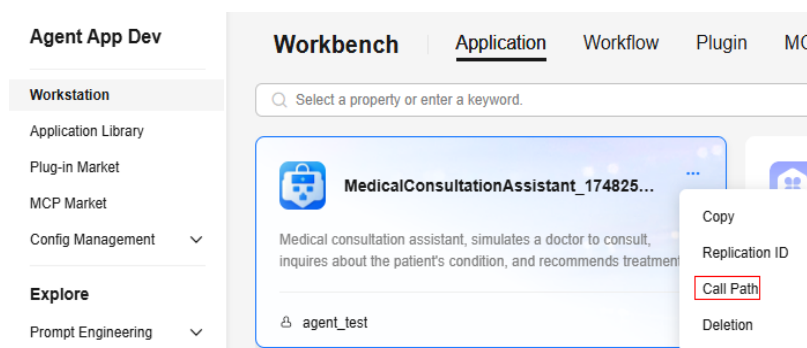
- To call a preset model, choose **Model Development > Model Deployment** in the navigation pane on the left, click **Call Path** in the model list on the **Preset service** page, and obtain the request URI of the model.

Figure 1-3 Calling a preset model

Model Deployment



- Obtain the request URI of the Agent application.
 - In the navigation pane on the left, choose **Agent Dev**. On the displayed page, choose **Workstation**. In the right pane, click the **Application** tab, select the application to be deployed, and choose **...** > **Call Path**.
 - On the **Call Path** page, you can obtain the request URI of the Agent application.

Figure 1-4 Call path

----End

1.4 Concepts

- A has full access permissions for all the resources and cloud services. It can be used to reset user passwords and grant users permissions. The should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant user permissions for routine management.
- IAM user

A user is created to use cloud services. Each user has its own identity credentials (password and access keys).

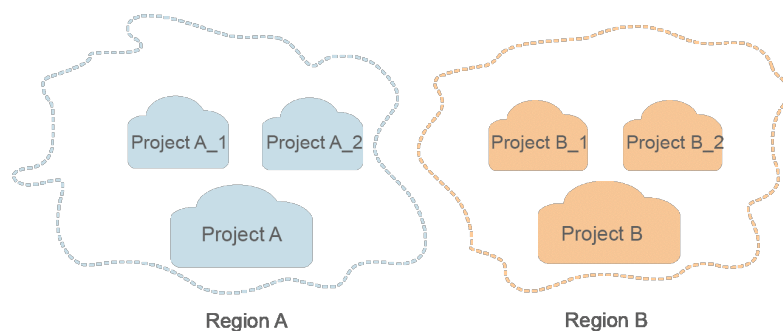
You can view the ID and user ID on the [My Credentials](#) page of the console. The name, username, and password will be required for API authentication.
- Region

Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified as universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides services of the same type or only provides services for specific tenants.
- AZ

AZs are physically isolated locations in a region, but are interconnected through an internal network for enhanced application availability.
- Project

A project corresponds to a region. Default projects are defined to group and physically isolate resources (including compute, storage, and network resources) between different regions. Users can be granted permissions in a default project to access all resources under their in the region associated with the project. For more refined access control, create subprojects under a project and resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.

Figure 1-5 Project isolating model



2 Calling REST APIs

2.1 Making an API Request

This section describes the structure of a REST API, and uses the API for **obtaining a user token** as an example to demonstrate how to call an API.

A request consists of the **request URI**, **request method**, **request header**, and **request body**.

Request URI

A request URI is in the following format:

{URI-scheme} :// {Endpoint} / {resource-path} ? {query-string}

Table 2-1 Request URI

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service.
resource-path	Access path of an API for performing a specified operation. You can obtain the path from the URI of a specific API.
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of "Parameter name=Parameter value".

For details about how to obtain the request URI, see **Request URI**. The following is an example:

https://{endpoint}/v1/{project_id}/deployments/{deployment_id}/chat/completions

Request Methods

HTTP request method, which indicates the type of operation that the service is requesting. The options are as follows:

- **GET**: requests the server to return specified resources.
- **PUT**: requests the server to update specified resources.
- **POST**: requests the server to add resources or perform special operations.
- **DELETE**: requests the server to delete specified resources, for example, an object.
- **HEAD**: same as GET except that the server must return only the response header.
- **PATCH**: requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

The request method in the URI of the API is **POST**, for example:

```
POST https://{endpoint}/v1/{project_id}/deployments/{deployment_id}/chat/completions
```

Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Common request headers are as follows:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json**.
- **X-Auth-Token**: specifies a user token only for token-based API authentication. For details about user tokens, see **Token-based Authentication** in [Authentication](#).

NOTE

In addition to supporting token-based authentication, APIs also support authentication using access key ID/secret access key (AK/SK). During AK/SK-based authentication, an SDK is used to sign the request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request. For more details, see [Authentication](#).

The following provides an example request with a header included.

```
POST https://{endpoint}/v1/{project_id}/deployments/{deployment_id}/chat/completions
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZlhvcNAQcCoIINOD...
```

Request Body

The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header. If the request body contains Chinese characters, these characters must be coded in UTF-8.

The request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

The following provides an example request with a body included. For details about the parameters, see the specific API.

```
POST https://{endpoint}/v1/{project_id}/deployments/{deployment_id}/chat/completions
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZlhvcNAQcCoIINOD...
{
  "messages": [
    {
      "content": "Introduce the Yangtze River and its typical fish species."
    }
  ],
  "temperature": 0.9,
  "max_tokens": 600
}
```

You can send the request to call an API through [curl](#), [Postman](#), or coding. For an API, you can obtain the request parameters and parameter descriptions in the response.

2.2 Authentication

You can choose any of the following authentication modes:

- Token-based authentication: Requests are authenticated using a token.
- API key authentication: If the model service deployed by a user needs to be called by other users, the original token-based authentication requires dynamic authentication and credential management, which is complex. In this case, API key authentication can be used. This method is more convenient than token-based authentication and complies with mainstream model calling specifications in the industry.

Token-based Authentication

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API.

NOTE

- A token is valid for 24 hours. When you need to use a token for authentication, you can cache it to prevent frequent calls.

Method for obtaining a token:

You can obtain a token by calling the related API. The following is an example of an API call:

- Pseudocode
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens //Obtain the token of the CN-Hong Kong region.
Content-Type: application/json
{
 "auth": {
 "identity": {
 "methods": [
 "password"
],
 "password": {
 "user": {
 "name": "*username*", // IAM username
 }
 }
 }
 }
}

```
        "password": "*****", //IAM user password
        "domain": {
            "name": "domainname" //Account name
        }
    },
    "scope": {
        "project": {
            "name": "ap-southeast-1" //PanguLM is currently deployed in the CN-Hong Kong region,
            and the value is ap-southeast-1.
        }
    }
}
```

- **Python**

```
import requests
import json

url = "https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens"
payload = json.dumps({
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "*****",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "projectname"
            }
        }
    }
})
headers = {
    'Content-Type': 'application/json'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.headers["X-Subject-Token"])
```

Procedure for obtaining the token:

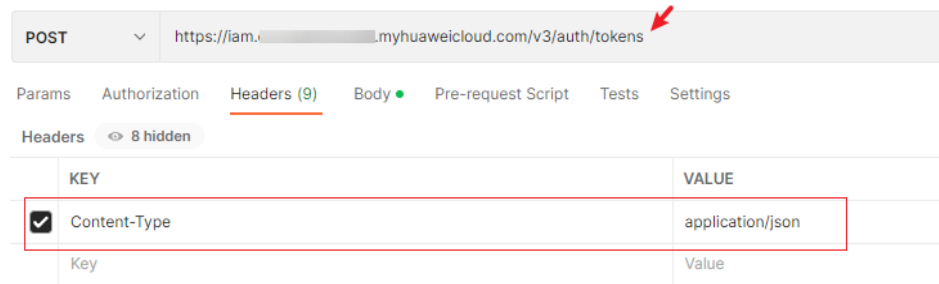
In this example, Postman is used to obtain the token.

1. Log in to the system and access the **My Credentials > API Credentials** page to obtain the username, domain name, and project ID.

The PanguLM is deployed in the CN-Hong Kong region. You need to obtain the project ID that belongs to the CN-Hong Kong region.

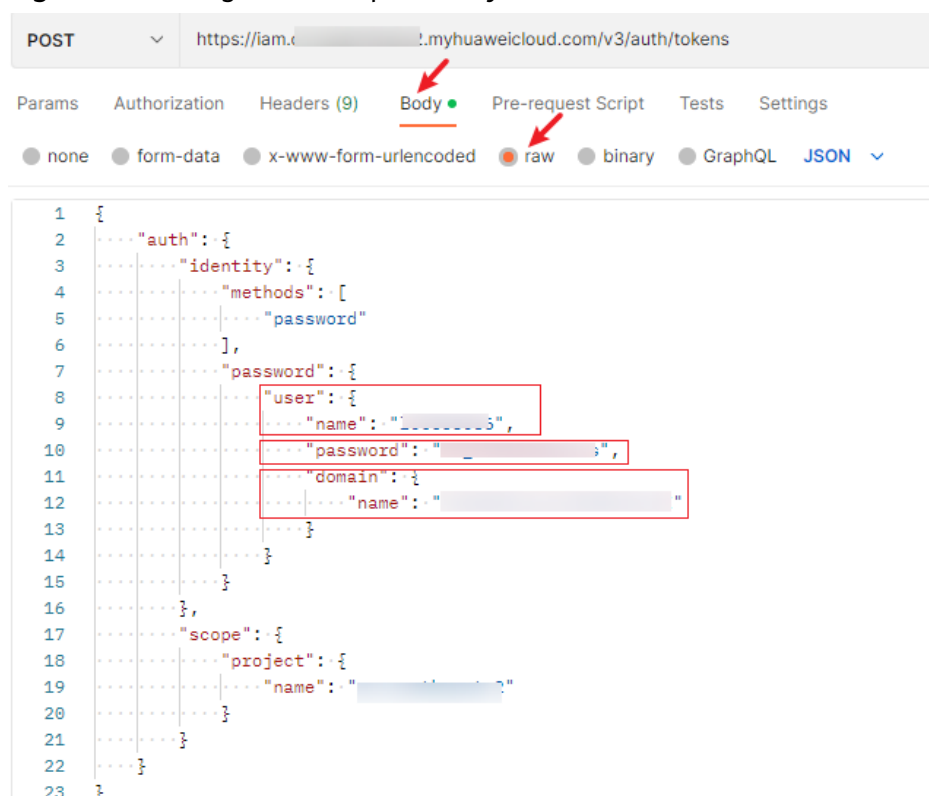
Figure 2-1 Obtaining the username, domain name, and project ID

2. Start Postman and create a POST request. Enter the URL of the token obtaining API in the CN-Hong Kong region. Set the request header parameters.
 - API address: **https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens**
 - Request header parameter: **Content-Type**; parameter value: **application/json**

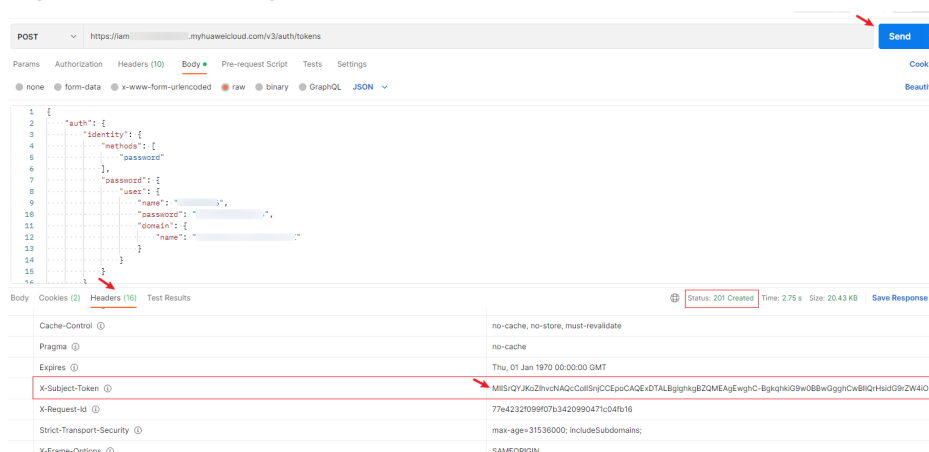
Figure 2-2 Entering the URL and request header parameter

3. Enter the request body of the API for obtaining a token. Click **Body**, select **raw**, copy and enter the following code by referring to [Figure 2-3](#), and enter the username, domain name, and password.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username", //IAM username
          "password": "*****", //Huawei Cloud account password
          "domain": {
            "name": "domainname" //Account name
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "ap-southeast-1" //PanguLM is currently deployed in the CN-Hong Kong region,
        and the value is ap-southeast-1.
      }
    }
  }
}
```

Figure 2-3 Configure the request body.

4. Click **Send** on Postman to send the request. If the status code **201** is returned, the API is successfully called. In this case, click **Headers**, find and copy the **X-Subject-Token** value, which is the token.

Figure 2-4 Obtaining a token

API Key Authentication

If the API service deployed by a user needs to be opened to other users, the original **token-based authentication** is not supported. In this case, API key authentication can be used to call APIs.

API key authentication is a method where the **X-Apig-AppCode** parameter (the parameter value is the API key value) is added to the HTTP request header during API calling. You do not need to sign the request content.

Before using this authentication mode, ensure that a large model has been deployed.

To obtain an API key, perform the following steps:

1. Log in to ModelArts Studio and access the desired workspace.
2. In the navigation pane on the left, choose **Model Development > Application Access**. Click **Create application access** in the upper right corner.
3. Select **All services** for **Associated Services**, or select **specifying service** and then select a deployed large model. Then, click **sure**.
4. Obtain the API key in the **API Key** column on the application access page.

2.3 Response

Status Codes

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see [Status Codes](#).

If status code **200** is returned for the calling of Pangu APIs, the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**.

Response Body

The body of a response is often returned in structured format as specified in the **Content-Type** header field. The response body transfers content except the response header.

The following is a response body of a successful API call:

```
{
  "id": "180f5745-4ee4-42a9-9869-23f829654bb7",
  "created": 1724915285,
  "choices": [
    {
      "index": 0,
      "text": "Once upon a time, in a land far, far away, there was a kingdom called Eldoria. Eldoria was a kingdom of great beauty, with lush green forests, sparkling rivers, and majestic mountains. The kingdom was ruled by a wise and just king, King Alaric, and his queen, Queen Isolde. They had a son, Prince Aiden, who was brave and kind, and a daughter, Princess Elara, who was beautiful and intelligent. Prince Aiden and Princess Elara were very close. They grew up together, playing in the gardens, exploring the forests, and learning about their kingdom. As they grew older, their bond deepened, and they realized that they had fallen in love with each other. However, their love was not without challenges. King Alaric and Queen Isolde had other plans for their children. They wanted Prince Aiden to marry a princess from a neighboring kingdom to form an alliance, and they wanted Princess Elara to marry a prince from a distant land to strengthen their kingdom's power. Prince Aiden and Princess Elara were heartbroken. They knew that they had to do something to be together. One night, they decided to run away from the palace and elope. They packed their bags and left the palace under the cover of darkness. They traveled through the forests and mountains, facing many dangers along the way. But their love for each other gave them the strength to overcome every obstacle. They finally reached a small village on the
```

outskirts of Eldoria, where they decided to settle down. They lived a simple life in the village, working hard and loving each other deeply. They were happy, but they knew that they could not stay hidden forever. One day, King Alaric and Queen Isolde found out about their escape and came to the village to bring them back. Prince Aiden and Princess Elara pleaded with their parents to let them stay together. They told them about their love and how they could not imagine living without each other. King Alaric and Queen Isolde were moved by their love and decided to let them stay together. They returned to the palace and announced their decision to the kingdom. The people of Eldoria were overjoyed and celebrated the union of Prince Aiden and Princess Elara. They were married in a grand ceremony, and their love story became a legend in Eldoria. Prince Aiden and Princess Elara ruled Eldoria together, bringing peace and prosperity to the kingdom. Their love story inspired many, and their kingdom flourished under their rule. They lived happily ever after, proving that true love can overcome any obstacle.

```
"ppl": 1.77809815678146e-36
  }
},
"usage": {
  "completion_tokens": 365,
  "prompt_tokens": 9,
  "total_tokens": 374
}
```

If an error occurs during API calling, an error code and error information will be returned.

The token is valid for 24 hours. The following error information indicates that the token has expired.

```
{
  "error_msg": "Incorrect IAM authentication information: token expires,
expires_at:2023-06-29T02:16:41.581000Z",
  "error_code": "APIG.0301",
  "request_id": "469967f55e6b225xxx"
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

3 API

3.1 Model Inference APIs

3.1.1 Third-Party Models

3.1.1.1 DeepSeek

Function

DeepSeek API is an API service based on the DeepSeek model. It supports text-based interaction in multiple scenarios and can quickly generate high-quality dialogs, copywriting passages, and stories. It is suitable for scenarios such as text summarization, intelligent Q&A, and content creation.

URI

The NLP inference service can be invoked using Pangu inference APIs (V1 inference APIs) or OpenAI APIs (V2 inference APIs).

The authentication modes of V1 and V2 APIs are different, and the request body and response body of V1 and V2 APIs are slightly different.

Table 3-1 NLP inference APIs

API Type	API URI
V1 inference API	POST /v1/{project_id}/deployments/{deployment_id}/chat/completions
V2 inference API	POST /api/v2/chat/completions

Table 3-2 Path parameters of the V1 inference API

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain a project ID, see Obtaining the Project ID .
deployment_id	Yes	String	Model deployment ID. For details about how to obtain the deployment ID, see Obtaining the Model Deployment ID .

Request Parameters

The authentication modes of the V1 and V2 inference APIs are different, and the request and response parameters are also different. The details are as follows:

Header parameters

1. The V1 inference API supports both token-based authentication and API key authentication. The request header parameters for the two authentication modes are as follows:
 - [Table 3-3](#) lists the request header parameters for [Token-based Authentication](#).

Table 3-3 Request header parameters (token-based authentication)

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. Used to obtain the permission required to call APIs. The token is the value of X-Subject-Token in the response header in Authentication .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .

- For details about the request header parameters in [API Key Authentication](#) mode, see [Table 3-4](#).

Table 3-4 Request header parameters (API key authentication)

Parameter	Mandatory	Type	Description
X-Apig-AppCode	Yes	String	API key. Used to obtain the permission required to call APIs. The API key is the value of X-Apig-AppCode in the response header in API Key Authentication .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .

2. The V2 inference API supports only API key authentication. For details about the request header parameters, see [Table 3-5](#).

Table 3-5 Request header parameters of V2 inference API (OpenAI-compatible API key authentication)

Parameter	Mandatory	Type	Description
Authorization	Yes	String	A character string consisting of Bearer and the API key obtained from created application access. A space is required between Bearer and the API key. For example, Bearer d59*****9C3 .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .

Request body parameters

The request body parameters of the V1 and V2 inference APIs are the same, as described in [Table 3-6](#).

Table 3-6 Request body parameters

Parameter	Mandatory	Type	Description
messages	Yes	Array of ChatCompletionMessageParam objects	<p>Multi-turn dialogue question-answer pairs, which contain the role and content attributes.</p> <ul style="list-style-type: none">• role indicates the role in a dialogue. The value can be system or user. If you want the model to answer questions as a specific persona, set role to system. If you do not use a specific persona, set role to user. In a dialogue request, you need to set role only once.• content indicates the content of a dialogue, which can be any text. <p>The messages parameter helps the model to generate a proper response based on the context in the dialogue.</p>
model	Yes	String	ID of the model to be used. Set this parameter based on the deployed model. The value can be DeepSeek-R1 or DeepSeek-V3.
stream	No	boolean	<p>Whether to enable the streaming mode. The streaming protocol is Server-Sent Events (SSE).</p> <p>If the streaming mode is enabled, set the value to true. After the streaming mode is enabled, this API sends the generated text to the client in real time, instead of sending all text at a time after the text generation is complete.</p> <p>Default value: false</p>

Parameter	Mandatory	Type	Description
temperature	No	Float	<p>Used to control the diversity and creativity of the generated text.</p> <p>A floating-point number that controls the randomness of sampling. A lower temperature produces more deterministic outputs. A higher temperature, for example, 0.9, produces more creative outputs. The value 0 indicates greedy sampling. If the value is greater than 1, the effect is very likely to be unavailable.</p> <p>temperature is one of the key parameters that affect the output quality and diversity of an LLM. Other parameters, like top_p, can also be used to adjust the behavior and preferences of the LLM. However, do not use temperature and top_p at the same time.</p> <p>Minimum value: 0. It is recommended that the value be greater than or equal to 1e-5.</p> <p>Maximum value: 1.0</p> <p>Default value: 1.0</p>

Parameter	Mandatory	Type	Description
top_p	No	Float	<p>Nucleus sampling parameter. As an alternative to adjusting the sampling temperature, the model will consider the results of the top_p probability tokens. 0.1 means that only tokens included in the top 10% probability will be considered. You are advised to change the value or temperature, but not both.</p> <p>Value range: (0.0, 1.0]</p> <p>Default value: 0.8</p> <p>NOTE</p> <p>A token is the smallest unit of text a model can work with. A token can be a word or part of characters. An LLM turns input and output text into tokens, generates a probability distribution for each possible word, and then samples tokens according to the distribution.</p>
max_tokens	No	Integer	<p>Maximum number of output tokens for the generated text.</p> <p>The total length of the input text plus the generated text cannot exceed the maximum length that the model can process.</p> <p>Minimum value: 1</p> <p>Maximum value: 8192</p> <p>Default value: 4096</p>

Parameter	Mandatory	Type	Description
presence_penalty	No	Float	<p>Controls how the LLM processes new tokens. If a token has already appeared in the generated text, the model will penalize this token when generating text. If the value of presence_penalty is a positive number, the model tends to generate new tokens that have not appeared before. That is, the model tends to talk about new topics.</p> <p>Minimum value: -2 Maximum value: 2 Default value: 0 (indicating that this parameter does not take effect)</p>
frequency_penalty	No	Float	<p>Controls how the model penalizes new tokens based on their existing frequency. If a token appears frequently in the training set, the model will penalize this token when generating text. A positive value penalizes new tokens that have already been used frequently, making it less likely to repeat words or phrases exactly.</p> <p>Minimum value: -2 Maximum value: 2 Default value: 0 (indicating that this parameter does not take effect)</p>

Table 3-7 ChatCompletionMessageParam

Parameter	Mandatory	Type	Description
role	Yes	String	<p>Role in a dialog. The default value range is as follows: system, user, assistant, tool, function. Customization is supported.</p> <p>If you want the model to answer questions as a specific persona, set role to system. If you do not use a specific persona, set role to user.</p> <p>When the parameter is returned, the value is fixed to assistant.</p> <p>In a dialogue request, you need to set role only once.</p>
content	Yes	String	<p>Content of a dialogue, which can be any text in tokens.</p> <p>A multi-turn dialogue cannot contain more than 20 content fields in the message parameter.</p> <p>Minimum length: 1</p> <p>Maximum length: token length supported by different models.</p> <p>Default value: None</p>

Response Parameters

Non-streaming

Status code: 200

Table 3-8 Response body parameters

Parameter	Type	Description
id	String	Uniquely identifies each response. The value is in the format of "chatcmpl-{random_uuid()}".
object	String	The value must be chat.completion .
created	Integer	Time when a response is generated, in seconds.
model	String	Request model ID.

Parameter	Type	Description
choices	Array of ChatCompletionResponseChoice objects	List of generated text.
usage	UsageInfo object	Token usage for the dialogue. This parameter helps you learn about model usage, preventing the model from generating excessive tokens.
prompt_logprobs	Object	Logarithmic probability information of the input text and the corresponding tokens. Default value: null

Table 3-9 ChatCompletionResponseChoice

Parameter	Type	Description
message	ChatMessage object	Generated text
index	Integer	Index of the generated text, starting from 0
finish_reason	String	Reason why the model stops generating tokens. Value: stop, length, content_filter, tool_calls, or insufficient_system_resource stop: The model stops generating text after the task is complete or when a pre-defined stop sequence is encountered. length: The output length reaches the context length limit of the model or the max_tokens limit. content_filter: The output content is filtered due to filter conditions. tool_calls: The model determines to call an external tool (function/API) to complete the task. insufficient_system_resource: Generation is interrupted because system inference resources are insufficient. Default value: stop
logprobs	Object	Evaluation metric, indicating the confidence value of the inference output. Default value: null

Parameter	Type	Description
stop_reason	Union[Integer, String]	Token ID or character string that instructs the model to stop generating. If the EOS token is encountered, the default value is returned. If the string or token ID in the stop parameter specified in the user request is encountered, the corresponding string or token ID is returned. This parameter is not a standard field of the OpenAI API but is supported by the vLLM API. Default value: None

Table 3-10 UsageInfo

Parameter	Type	Description
prompt_tokens	Number	Number of tokens contained in the user prompt.
total_tokens	Number	Number of all tokens in a dialog request.
completion_tokens	Number	Number of answers tokens generated by the inference model.

Table 3-11 ChatMessage

Parameter	Type	Description
role	String	Role that generates the message. The value must be assistant .
content	String	Content of a dialogue Minimum length: 1 Maximum length: token length supported by different models.
reasoning_content	String	Reasoning steps that led to the final conclusion (thinking process of the model). NOTE This parameter is available only for the DeepSeek-R1 model.

Streaming (with stream set to true)

Status code: 200

Table 3-12 Data units output in streaming mode

Parameter	Type	Description
data	CompletionStreamResponse object	If stream is set to true , messages generated by the model will be returned in streaming mode. The generated text is returned incrementally. Each data field contains a part of the generated text until all data fields are returned.

Table 3-13 CompletionStreamResponse

Parameter	Type	Description
id	String	Unique identifier of the dialogue.
created	Integer	Unix timestamp (in seconds) when the chat was created. The timestamps of each chunk in the streaming response are the same.
model	String	Name of the model that generates the completion.
object	String	Object type, which is chat.completion.chunk .
choices	ChatCompletionResponseStreamChoice	A list of completion choices generated by the model.

Table 3-14 ChatCompletionResponseStreamChoice

Parameter	Type	Description
index	Integer	Index of the completion in the completion choice list generated by the model.

Parameter	Type	Description
finish_reason	String	Reason why the model stops generating tokens. Value: stop , length , content_filter , tool_calls , or insufficient_system_resource stop: The model stops generating text after the task is complete or when a pre-defined stop sequence is encountered. length: The output length reaches the context length limit of the model or the max_tokens limit. content_filter: The output content is filtered due to filter conditions. tool_calls: The model determines to call an external tool (function/API) to complete the task. insufficient_system_resource: Generation is interrupted because system inference resources are insufficient.

Status code: 400

Table 3-15 Response body parameters

Parameter	Type	Description
error_code	String	Error code
error_msg	String	Error message

Example Request

- Non-streaming
V1 inference API:
POST https://{endpoint}/v1/{project_id}/alg-infer/3rdnlp/service/{deployment_id}/v1/chat/completions

Request Header:
Content-Type: application/json
X-Auth-Token:
MIINRwYJKoZIhvcNAQcCoIIINODCCDTQCAQExDTALBgIghkgBZQMEAgEwgguVBgkqhkiG...

Request Body:
{
 "model": "DeepSeek-V3",
 "messages": [
 {
 "role": "user",
 "content": "Hello"
 }
]
}
V2 inference API:
POST https://{endpoint}/api/v2/chat/completions

```
Request Header:  
Content-Type: application/json  
Authorization: Bearer 201ca68f-45f9-4e19-8fa4-831e...
```

```
Request Body:  
{  
  "model": "DeepSeek-V3",  
  "messages": [  
    {  
      "role": "user",  
      "content": "Hello"  
    }  
  ]  
}
```

- **Streaming (with stream set to true)**

V1 inference API:

POST https://{endpoint}/v1/{project_id}/alg-infer/3rdnlp/service/{deployment_id}/v1/chat/completions

```
Request Header:  
Content-Type: application/json  
X-Auth-Token:  
MIINRwYJKoZlIhvcNAQcCoIINODCCDTQCAQExDTALBglghkgBZQMEAgEwgguVBgkqhkiG...
```

```
Request Body:  
{  
  "model": "DeepSeek-V3",  
  "messages": [  
    {  
      "role": "user",  
      "content": "Hello"  
    }  
  ],  
  "stream": true  
}
```

V2 inference API:

POST <https://{endpoint}/api/v2/chat/completions>

```
Request Header:  
Content-Type: application/json  
Authorization: Bearer 201ca68f-45f9-4e19-8fa4-831e...
```

```
Request Body:  
{  
  "model": "DeepSeek-V3",  
  "messages": [  
    {  
      "role": "user",  
      "content": "Hello"  
    }  
  ],  
  "stream": true  
}
```

Example Response

Status code: 200

OK

- **Non-streaming Q&A response**

```
{  
  "id": "chat-9a75fc02e45d48db94f94ce38277beef",  
  "object": "chat.completion",  
  "created": 1743403365,  
  "model": "DeepSeek-V3",  
  "choices": [  
    {  
      "index": 0,  
      "message": {  
        "role": "assistant",
```

```
        "content": "Hello. How can I help you?",
        "tool_calls": []
      },
      "finish_reason": "stop"
    }
  ],
  "usage": {
    "prompt_tokens": 64,
    "total_tokens": 73,
    "completion_tokens": 9
  }
}
```

- Non-streaming Q&A response with chain of thought

```
{
  "id": "81c34733-0e7c-4b4b-a044-1e1fcd54b8db",
  "model": "deepseek-r1_32k",
  "created": 1747485310,
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "\n\nHello. Nice to meet you. Is there anything I can do for you?",
        "reasoning_content": "Hmm, the user just sent a short \"Hello\", which is a greeting in Chinese. First, I need to confirm their needs. They might want to test the reply or have a specific question. Also, I need to consider whether to respond in English, but since the user used Chinese, it's more appropriate to reply in Chinese.\n\nThen, I need to ensure the reply is friendly and complies with the guidelines, without involving sensitive content. The user may expect further conversation or need help with a question. At this point, I should maintain an open-ended response, inviting them to raise specific questions or needs. For example, I could say, \"Hello! Nice to meet you, is there anything I can help you with?\" This is both polite and proactive in offering assistance.\n\nAdditionally, avoid using any format or markdown, keeping it natural and concise. The user might be new to this platform and unfamiliar with how to ask questions, so a more encouraging tone might be better. Check for any spelling or grammatical errors to ensure the reply is correct and error-free.\n"
      },
      "tool_calls": [
        ]
      },
      "finish_reason": "stop"
    }
  ],
  "usage": {
    "completion_tokens": 184,
    "prompt_tokens": 6,
    "total_tokens": 190
  }
}
```

- Streaming Q&A response

Response body of the V1 inference API

```
data:
{"id":"chat-97313a4bc0a342558364345de0380291","object":"chat.completion.chunk","created":1743404317,"model":"DeepSeek-V3","choices":[{"index":0,"message":{"role":"assistant"},"logprobs":null,"finish_reason":null]}
```

```
data:
{"id":"chat-97313a4bc0a342558364345de0380291","object":"chat.completion.chunk","created":1743404317,"model":"DeepSeek-V3","choices":[{"index":0,"message":{"content":"Hello"},"logprobs":null,"finish_reason":null}]}
```

```
data:
{"id":"chat-97313a4bc0a342558364345de0380291","object":"chat.completion.chunk","created":1743404317,"model":"DeepSeek-V3","choices":[{"index":0,"message":{"content":", can I help you?"},"logprobs":null,"finish_reason":"stop","stop_reason":null}]}
```

data:[DONE]

Response body of the V2 inference API

```
data:
{"id":"chat-97313a4bc0a342558364345de0380291","object":"chat.completion.chunk","created":1743404317,"model":"DeepSeek-V3","choices":[{"index":0,"delta":
```

```
{"role":"assistant"},"logprobs":null,"finish_reason":null}]

data:
{"id":"chat-97313a4bc0a342558364345de0380291","object":"chat.completion.chunk","created":1743404317,"model":"DeepSeek-V3","choices":[{"index":0,"delta":{"content":"Hello"},"logprobs":null,"finish_reason":null}}]

data:
{"id":"chat-97313a4bc0a342558364345de0380291","object":"chat.completion.chunk","created":1743404317,"model":"DeepSeek-V3","choices":[{"index":0,"delta":{"content":". Can I help you?"},"logprobs":null,"finish_reason":"stop","stop_reason":null}}]

data:[DONE]
```

- Streaming Q&A response with chain of thought

Response body of the V1 inference API

```
data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"role":"assistant","content":"","logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":6,"completion_tokens":0}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"reasoning_content":"Hmm"},"logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":7,"completion_tokens":1}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"reasoning_content":"","logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":8,"completion_tokens":2}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"reasoning_content":"user sent"},"logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":10,"completion_tokens":4}}]

...

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"reasoning_content":"Generate"},"logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":185,"completion_tokens":179}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"reasoning_content":"final"},"logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":186,"completion_tokens":180}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"reasoning_content":"reply.\n"},"logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":188,"completion_tokens":182}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"content":"\n\nHello"},"logprobs":null,"finish_reason":null}},{"usage":{"prompt_tokens":6,"total_tokens":191,"completion_tokens":185}}]

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"message":{"content":"."}}
```

```
Nice"},"logprobs":null,"finish_reason":null},"usage":
{"prompt_tokens":6,"total_tokens":193,"completion_tokens":187}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"message":{"content":"to
meet"},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":194,"completion_tokens":188}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"message":{"content":"you"},"logprobs
":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens
":195,"completion_tokens":189}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"message":{"content":"
What"},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":197,"completion_tokens":191}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"message":{"content":"can I
do"},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":199,"completion_tokens":193}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"message":{"content":"for
you"},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":201,"completion_tokens":195}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"message":{"content":"?
"},"logprobs":null,"finish_reason":"stop","stop_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":203,"completion_tokens":197}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[],"usage":{"prompt_tokens":6,"total_tokens":203,"completion_tokens":197}}

data:[DONE]
Response body of the V2 inference API
data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"delta":
{"role":"assistant","content":""},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":6,"completion_tokens":0}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"delta":
{"reasoning_content":"Hmm"},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":7,"completion_tokens":1}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"delta":
{"reasoning_content":"","logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":8,"completion_tokens":2}}

data:{"id":"chat-
cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model"
:"DeepSeek-R1","choices":[{"index":0,"delta":{"reasoning_content":"user
sent"},"logprobs":null,"finish_reason":null}],"usage":
{"prompt_tokens":6,"total_tokens":10,"completion_tokens":4}}
```

```
...

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"reasoning_content":"Generate"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":185,"completion_tokens":179}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"reasoning_content":"final"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":186,"completion_tokens":180}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"reasoning_content":"reply.\n"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":188,"completion_tokens":182}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"\n\nHello"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":191,"completion_tokens":185}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":". Nice"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":193,"completion_tokens":187}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"to meet"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":194,"completion_tokens":188}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"you"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":195,"completion_tokens":189}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"."},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":197,"completion_tokens":191}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"Can I help"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":199,"completion_tokens":193}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"your"},"logprobs":null,"finish_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":201,"completion_tokens":195}}

data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[{"index":0,"delta":{"content":"?"},"logprobs":null,"finish_reason":"stop","stop_reason":null}],"usage":{"prompt_tokens":6,"total_tokens":203,"completion_tokens":197}}
```

```
data:{"id":"chat-cc897cfa872a4fc993a803bbddf9268a","object":"chat.completion.chunk","created":1747485542,"model":"DeepSeek-R1","choices":[],"usage":{"prompt_tokens":6,"total_tokens":203,"completion_tokens":197}}
```

```
data:[DONE]
```

- Streaming Q&A response when the content is not approved
event:moderation data:{"suggestion":"block","reply":"As an AI language model, my goal is to provide help and information in a positive, proactive, and safe manner. Your question is beyond my answer range."}

```
data:[DONE]
```

Status Codes

For details, see [Status Codes](#).

Error Codes

For details, see [Error Codes](#).

3.2 Data Engineering APIs

3.2.1 Querying Data Lineages

Function

This API allows you to import raw datasets from OBS. You can use the OBS path to query all raw datasets created based on the path and the subsequent lineage information.

URI

GET /v1/{project_id}/workspaces/{workspace_id}/data-management/lineages

Request Parameters

Table 3-16 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. Used to obtain the permission required to call APIs. The token is the value of X-Subject-Token in the response header in Authentication .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .

Table 3-17 Request query parameters

Parameter	Mandatory	Type	Description
limit	Yes	integer	Definition: Maximum number of lineages can be returned by the API. Constraints: N/A Value range: [1, 1000] Default value: 100
from_path	Yes	string	Definition: Source OBS path. Constraints: Full OBS path of the end tenant. Value range: N/A Default value: N/A

Response Parameters

Parameter	Type	Description
lineages	array	Definition: Dataset lineage list. Constraints: The type of items in the list is Lineage . Value range: N/A Default value: N/A

Example Request

GET https://{endpoint}/v1/{project_id}/workspaces/{workspace_id}/data-management/lineages?
limit=100&from_path=bucket/folder1/folder2

Request Header:
Content_Type: application/json
X-Auth-Token: MIIVV...

Request Params:
limit: 1000
from_path: bucket/folder1/folder2

Example Response

```
{
  "lineages": [
    {
      "id": null,
      "from_id": null,
      "from_name": null,
      "from_catalog": null,
      "from_type": "OBS",
      "to_id": "1352299121133883392",
      "to_name": null,
      "to_catalog": "ORIGINAL",
      "to_type": "DATASET",
      "process_id": null,
      "process_name": null,
      "process_type": null,
      "train_job_name": null,
      "model_type": null,
      "train_type": null,
      "create_time": null,
      "from_path": "bucket/folder",
      "from_path_existed": null
    },
    {
      "id": "1352299380551585793",
      "from_id": "1352299121133883392",
      "from_name": " time series-regression-test",
      "from_catalog": "ORIGINAL",
      "from_type": "DATASET",
      "to_id": "1352299379473649664",
      "to_name": "pub_time series regression",
      "to_catalog": "PUBLISH",
      "to_type": "DATASET",
      "process_id": "lt_97a2aa4cca744775aa5c7cfe3cb36121",
      "process_name": "pub_time series regression",
      "process_type": "PUBLISH",
      "train_job_name": null,
      "model_type": null,
      "train_type": null,
      "create_time": null,
      "from_path": null,
      "from_path_existed": null
    }
  ]
}
```

Status Codes

For details, see [Status Codes](#).

Error Codes

For details, see [Error Codes](#).

3.2.2 Permanently Deleting a Dataset

Function

For data uploaded from OBS, you need to delete the associated raw data in OBS when deleting the datasets.

URI

POST /v1/{project_id}/workspaces/{workspace_id}/data-management/dataset/
permanent-delete:

Request Parameters

Table 3-18 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. Used to obtain the permission required to call APIs. The token is the value of X-Subject-Token in the response header in Authentication .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .

Table 3-19 Request body parameters

Parameter	Mandatory	Type	Description
dataset_name	Yes	string	Definition: Dataset name. Constraints: The name length ranges from [1,128]. Value range: N/A Default value: N/A

Parameter	Mandatory	Type	Description
catalog	No	CatalogEnum	Definition: Dataset form. Constraints: N/A Value range: <ul style="list-style-type: none">• ORIGINAL: A type of dataset generated during data importing.• PROCESS: A type of the dataset generated during data processing.• PUBLISH: A type of dataset generated during data publishing. Default value: N/A
delete_obs	No	boolean	Definition: Whether to delete OBS data. Constraints: N/A Value range: <ul style="list-style-type: none">• true: Delete OBS data.• false: Do not delete OBS data. Default value: N/A

Response Parameters

Parameter	Type	Description
dataset_name	string	Definition: Dataset name. Constraints: N/A Value range: The name length ranges from [1,128]. Default value: N/A

Parameter	Type	Description
catalog	CatalogEnum	Definition Dataset form. Constraints: N/A Value range: <ul style="list-style-type: none">• ORIGINAL: A type of dataset generated during data importing.• PROCESS: A type of the dataset generated during data processing.• PUBLISH: A type of dataset generated during data publishing. Default value: N/A
result	boolean	Definition: Operation result. Constraints: N/A Value range: <ul style="list-style-type: none">• true: Deletion success.• false: Deletion failure. Default value: N/A

Example Request

Permanently delete the original OBS data corresponding to the dataset.

```
POST https://{endpoint}/v1/{project_id}/workspaces/{workspace_id}/data-management/dataset/permanent-delete?dataset_name=pub_345135233&catalog=PROCESS&delete_obs=true

Request Header:
Content_Type: application/json
X-Auth-Token: MIIVV...

Request Params:
dataset_name: pub_345135233
catalog: PROCESS
delete_obs:true
```

Example Response

```
{
  "DatasetOperationResp": [
    {
      "dataset_name": pub_345135233,
      "catalog": PROCESS,
      "result": true,
```

```
    },  
  }  
}
```

Status Codes

For details, see [Status Codes](#).

Error Codes

For details, see [Error Codes](#).

3.3 Agent APIs

3.3.1 Calling an Application

Function

This API allows you to call the API of a created application and input a question to obtain the application execution result.

URI

POST /v1/{project_id}/agents/{agent_id}/conversations/{conversation_id}

For details about how to obtain the URI, see [Request URI](#).

Table 3-20 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain a project ID, see Obtaining the Project ID .
agent_id	Yes	String	Agent ID. For details about how to obtain the agent ID, perform the following steps: On the Agent App Dev page, choose Workstation > Application in the navigation pane. Locate the target agent, click *** , and select Replication ID .
conversation_id	Yes	String	Conversation ID, which uniquely identifies a conversation. The conversation ID can be set to any value in the standard UUID format.

Request Parameters

Table 3-21 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. Used to obtain the permission required to call APIs. The token is the value of X-Subject-Token in the response header in Authentication .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .
stream	Yes	Boolean	Whether to enable streaming calling. This function is enabled by default. <ul style="list-style-type: none">• true: Enable• false: Disable NOTE Currently, the agent supports only streaming calling. Therefore, set this parameter to true .

Table 3-22 Request body parameters

Parameter	Mandatory	Type	Description
query	Yes	String	User's question, used as the input to the agent.

Response Parameters

Streaming (with stream set to true in the header)

Status code: 200

Table 3-23 Data units output in streaming mode

Parameter	Type	Description
data	String	<ul style="list-style-type: none">• If stream is set to true, agent execution messages will be returned in streaming mode.• The generated text is returned incrementally. Each data field contains a part of the generated text until all data fields are returned.

Table 3-24 Data units output in streaming mode

Parameter	Type	Description
event	String	Data unit type. The options are as follows: <ul style="list-style-type: none">• start: start node, indicating that the model is called to start a conversation.• message: message node, indicating the message returned by the model.• plugin_start: plug-in request node, indicating the request for calling a plug-in.• plugin_end: plug-in response node, indicating the response to calling a plug-in.• statistic_data: execution data node, including the time consumed by the current call.• summary_response: message summary node, including the full response information of the current call.• done: end node, indicating that the streaming call ends.
content	Object	Message block content, which varies depending on the event value.
createdTime	long	Timestamp for returning the message block, for example, 1733817348963 .
latency	Object	Time consumed, including the following elements: <ul style="list-style-type: none">• plugin: time consumed for calling the plug-in• model: time consumed for calling the model• overall: total consumed time

Parameter	Type	Description
plugin	Object	Plug-in request, including the following elements: <ul style="list-style-type: none">● name: plug-in name● arguments: input parameters of the plug-in

Example Request

Streaming (with stream set to true in the header)

```
POST https://{endpoint}/v1/{project_id}/agent-run/agents/{agent_id}/conversations/{conversation_id}

Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZlHvcNAQcCoIINODCCDTQCAQExDTALBglgghkgBZQMEAgEwgguVBgkqhkiG...
stream: true

Request Body:
{
  "query": "Query the status of meeting room A12 from 09:00 to 10:00."
}
```

Example Response

```
data:{"event": "start", "createdTime": 1735558575017}

data:{"event": "message", "content": "OK", "createdTime": 1735558576300}

data:{"event": "message", "content": "", "createdTime": 1735558576301}

data:{"event": "message", "content": "I will", "createdTime": 1735558576301}

data:{"event": "message", "content": "call the", "createdTime": 1735558576302}

data:{"event": "message", "content": "query", "createdTime": 1735558576302}

data:{"event": "message", "content": "_", "createdTime": 1735558576302}

data:{"event": "message", "content": "meeting", "createdTime": 1735558576302}

data:{"event": "message", "content": "_", "createdTime": 1735558576302}

data:{"event": "message", "content": "room", "createdTime": 1735558576303}

data:{"event": "message", "content": "_status", "createdTime": 1735558576303}

data:{"event": "message", "content": "tool", "createdTime": 1735558576303}

data:{"event": "message", "content": "to", "createdTime": 1735558576304}

data:{"event": "message", "content": "query", "createdTime": 1735558576304}

data:{"event": "message", "content": "A", "createdTime": 1735558576304}

data:{"event": "message", "content": "12", "createdTime": 1735558576304}

data:{"event": "message", "content": "meeting room", "createdTime": 1735558576305}

data:{"event": "message", "content": "from", "createdTime": 1735558576305}

data:{"event": "message", "content": "9", "createdTime": 1735558576305}
```

```
data:{ "event": "message", "content": ".", "createdTime": 1735558576305 }
data:{ "event": "message", "content": "to", "createdTime": 1735558576306 }
data:{ "event": "message", "content": "10", "createdTime": 1735558576306 }
data:{ "event": "message", "content": ".00", "createdTime": 1735558576306 }
data:{ "event": "message", "content": "status", "createdTime": 1735558576306 }
data:{ "event": "message", "content": ".", "createdTime": 1735558576306 }
data:{ "event": "message", "content": "Please", "createdTime": 1735558576307 }
data:{ "event": "message", "content": "wait", "createdTime": 1735558576307 }
data:{ "event": "message", "content": "a moment", "createdTime": 1735558576307 }
data:{ "event": "message", "content": ".", "createdTime": 1735558576307 }
data:{ "event": "message", "content": " ", "createdTime": 1735558576307 }
data:{ "event": "message", "content": " query", "createdTime": 1735558576307 }
data:{ "event": "message", "content": "_", "createdTime": 1735558576308 }
data:{ "event": "message", "content": "meeting", "createdTime": 1735558576308 }
data:{ "event": "message", "content": "_", "createdTime": 1735558576308 }
data:{ "event": "message", "content": "room", "createdTime": 1735558576308 }
data:{ "event": "message", "content": "_status", "createdTime": 1735558576308 }
data:{ "event": "message", "content": "|", "createdTime": 1735558576308 }
data:{ "event": "message", "content": "{", "createdTime": 1735558576309 }
data:{ "event": "message", "content": "meeting", "createdTime": 1735558576309 }
data:{ "event": "message", "content": "Room", "createdTime": 1735558576309 }
data:{ "event": "message", "content": "\:", "createdTime": 1735558576309 }
data:{ "event": "message", "content": "{", "createdTime": 1735558576309 }
data:{ "event": "message", "content": "number", "createdTime": 1735558576310 }
data:{ "event": "message", "content": "\.", "createdTime": 1735558576310 }
data:{ "event": "message", "content": " 12", "createdTime": 1735558576310 }
data:{ "event": "message", "content": "}", "createdTime": 1735558576310 }
data:{ "event": "message", "content": "\:", "createdTime": 1735558576310 }
data:{ "event": "message", "content": "start", "createdTime": 1735558576310 }
data:{ "event": "message", "content": "\:\:", "createdTime": 1735558576311 }
data:{ "event": "message", "content": "9", "createdTime": 1735558576311 }
data:{ "event": "message", "content": ".00", "createdTime": 1735558576311 }
data:{ "event": "message", "content": "\:", "createdTime": 1735558576311 }
data:{ "event": "message", "content": "end", "createdTime": 1735558576311 }
data:{ "event": "message", "content": "\:\:", "createdTime": 1735558576311 }
```

```
data:{\"event\":\"message\",\"content\":\"10\",\"createdTime\":1735558576311}
data:{\"event\":\"message\",\"content\":\":00\",\"createdTime\":1735558576312}
data:{\"event\":\"message\",\"content\":\"\\\"\",\"createdTime\":1735558576312}
data:{\"event\":\"message\",\"content\":\" \",\"createdTime\":1735558576312}
data:{\"event\":\"plugin_start\",\"type\":\"plugin\",\"latency\":{\"overall\":1.3},\"plugin\":
{\"name\":\"query_meeting_room_status\",\"arguments\":{\"meetingRoom\":{\"number\": 12}, \"start\":
\\\"9:00\\\", \\\"end\\\": \\\"10:00\\\"}\"},\"createdTime\":1735558576316}
data:{\"event\":\"plugin_end\",\"content\":{\"result\":\"Idle\"},\"role\":\"function\",\"latency\":
{\"overall\":1.51,\"plugin\":0.0},\"createdTime\":1735558576521}
data:{\"event\":\"start\",\"createdTime\":1735558576522}
data:{\"event\":\"message\",\"content\":\"A\",\"createdTime\":1735558576976}
data:{\"event\":\"message\",\"content\":\"12\",\"createdTime\":1735558576977}
data:{\"event\":\"message\",\"content\":\"meeting room\",\"createdTime\":1735558576977}
data:{\"event\":\"message\",\"content\":\"from\",\"createdTime\":1735558576977}
data:{\"event\":\"message\",\"content\":\"9\",\"createdTime\":1735558576978}
data:{\"event\":\"message\",\"content\":\":00\",\"createdTime\":1735558576978}
data:{\"event\":\"message\",\"content\":\"to\",\"createdTime\":1735558576978}
data:{\"event\":\"message\",\"content\":\"10\",\"createdTime\":1735558576978}
data:{\"event\":\"message\",\"content\":\":00\",\"createdTime\":1735558576978}
data:{\"event\":\"message\",\"content\":\"in\",\"createdTime\":1735558576978}
data:{\"event\":\"message\",\"content\":\"this\",\"createdTime\":1735558576979}
data:{\"event\":\"message\",\"content\":\"during\",\"createdTime\":1735558576979}
data:{\"event\":\"message\",\"content\":\"is\",\"createdTime\":1735558576979}
data:{\"event\":\"message\",\"content\":\"idle\",\"createdTime\":1735558576979}
data:{\"event\":\"message\",\"content\":\".\",\"createdTime\":1735558576979}
data:{\"event\":\"message\",\"content\":\" \",\"createdTime\":1735558576980}
data:{\"event\":\"statistic_data\",\"latency\":{\"overall\":1.97},\"createdTime\":1735558576986}
data:{\"event\":\"summary_response\",\"content\":\"A12 meeting room from 09:00 to 10:00 in this duration is
idle. \",\"role\":\"assistant\",\"createdTime\":1735558576987}
data:{\"event\":\"done\",\"createdTime\":1735558577011}
```

Status Codes

For details, see [Status Codes](#).

Error Codes

For details, see [Error Codes](#).

3.3.2 Calling a Workflow

Function

This API allows you to call the API of a created workflow and input a question to obtain the workflow execution result.

URI

POST /v1/{project_id}/workflows/{workflow_id}/conversations/{conversation_id}

For details about how to obtain the URI, see [Request URI](#).

Table 3-25 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain a project ID, see Obtaining the Project ID .
workflow_id	Yes	String	Workflow ID. For details about how to obtain the workflow ID, perform the following steps: On the Agent App Dev page, choose Workstation > Workflow in the navigation pane on the left. Locate the target workflow, click ... , and select Replication ID .
conversation_id	Yes	String	Conversation ID, which uniquely identifies a conversation. The conversation ID can be set to any value in the standard UUID format.

Request Parameters

Table 3-26 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. Used to obtain the permission required to call APIs. The token is the value of X-Subject-Token in the response header in Authentication .

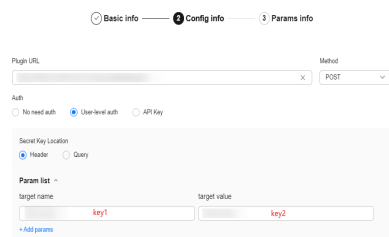
Parameter	Mandatory	Type	Description
Content-Type	Yes	String	MIME type of the request body. The value is application/json .
stream	No	Boolean	Whether to enable streaming calling. <ul style="list-style-type: none">• true: Enable• false: Disable

Table 3-27 Request body parameters

Parameter	Mandatory	Type	Description
inputs	Yes	Map<String, Object>	User's question, which is used as the input to the workflow and corresponds to the WORKFLOW_STARTED parameter of the workflow. The default field query is entered by the user.
plugin_configs	No	List<PluginConfig>	Plug-in configurations. When a user-defined plug-in node is configured in a workflow, authentication information may be required. For details about the structure, see Table 3-28 .

Table 3-28 PluginConfig parameters

Parameter	Mandatory	Type	Description
plugin_id	Yes	String	Plug-in ID. To obtain the ID, perform the following steps: On the Agent App Dev page, choose Workstation > Plugin in the navigation pane. Locate the target plug-in, click *** , and select Replication ID .

Parameter	Mandatory	Type	Description
config	Yes	Map<String, String>	<p>Plug-in configurations.</p> <p>When the workflow is associated with a plug-in node and the plug-in requires user-level authentication, you need to configure the authentication information. For example, you can set config to {"key2": "value"} for the following plug-in.</p>  <p>In other cases, this parameter does not need to be set. Specify an empty array for plugin_configs.</p>

Response Parameters

Non-streaming (with stream set to false in the header)

Status code: 200

Table 3-29 Data units output in non-streaming mode

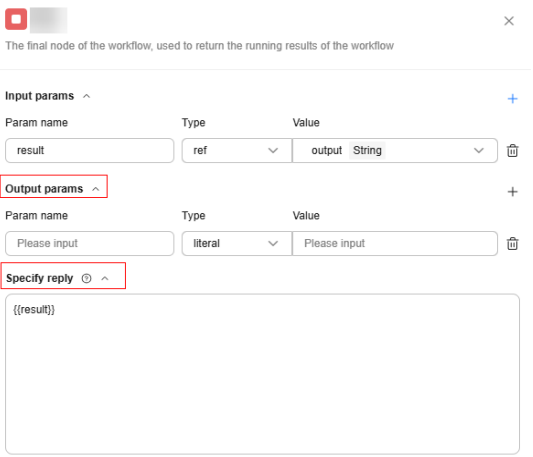
Parameter	Type	Description
outputs	Map<String, Object>	<p>Final output of the workflow. Multiple parameters are supported.</p> <p>NOTE The following is an example of outputs:</p> <ul style="list-style-type: none">The responseContent parameter is available by default. The value is the Specify reply content of the workflow's end node.You can customize parameters in the Output params module of the workflow's end node. Customized parameters are displayed in the user_fields parameter. <pre>"outputs":{"user_fields":{"aaa":"1","vvv":[{"role":"user","content":"1"}]},"responseContent":" Hello! \uD83D\uDE0A You entered 1. Is there anything I can help you with? If you have any specific questions or requirements, feel free to let me know!"}</pre> 
messages	List<Message >	Replies of the workflow assistant, such as the messages returned in the questioner node. For details, see Table 3-30 .
status	Map<String, Object>	Status, including the status code and description.
start_time	Long	Start time.
end_time	Long	End time.

Table 3-30 Message

Parameter	Type	Description
role	String	Conversation role, which can be user or assistant
content	String	Conversation content

Streaming (with stream set to true or not specified in the header)**Status code: 200****Table 3-31** Data units output in streaming mode

Parameter	Type	Description
data	String	If stream is set to true , workflow execution messages will be returned in streaming mode. The generated text is returned incrementally. Each data field contains a part of the generated text until all data fields are returned.

Table 3-32 Data units output in streaming mode

Parameter	Type	Description
event	String	Data unit type. The options are as follows: <ul style="list-style-type: none">• workflow_started: workflow start event, indicating that the workflow starts running.• workflow_finished: workflow end event, indicating that the workflow ends.• message: message event, indicating the message returned in streaming mode during workflow execution.• error: error event, indicating the workflow execution error information.• end: end event, indicating the request ends.
data	Object	Message block content, which varies depending on the event value.

Table 3-33 Data unit of the workflow_started event

Parameter	Type	Description
start_time	Long	Workflow start time.

Table 3-34 Data unit of the workflow_finished event

Parameter	Type	Description
start_time	Long	Workflow start time.

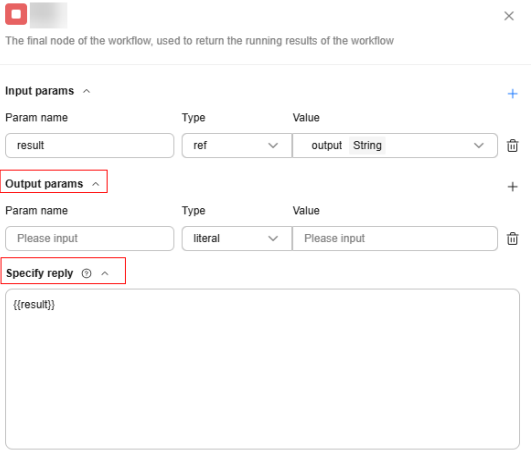
Parameter	Type	Description
end_time	Long	Workflow end time.
outputs	Map<String, Object>	<div>Final output of the workflow. Multiple parameters are supported.</div> <div>NOTE</div> <div>The following is an example of outputs:</div> <div><ul style="list-style-type: none">The responseContent parameter is available by default. The value is the Specify reply content of the workflow's end node.You can customize parameters in the Output params module of the workflow's end node. Customized parameters are displayed in the user_fields parameter.</div> <div><pre>"outputs":{"user_fields":{"aaa":"1","vvv":[{"role":"user","content":"1"}]},"responseContent":" Hello! \uD83D\uDE0A You entered 1. Is there anything I can help you with? If you have any specific questions or requirements, feel free to let me know!"}</pre></div> <div></div>
status	Map<String, Object>	Status, including the status code and description.

Table 3-35 Data unit of the message event

Parameter	Type	Description
text	String	Message block of the workflow output
index	Integer	Index of a message block
node_id	String	Node ID
node_type	String	Node type
node_name	String	Node name

Table 3-36 Data unit of the error event

Parameter	Type	Description
code	String	Workflow execution error code
message	String	Workflow execution error message
node_id	String	Node ID
node_type	String	Node type
node_name	String	Node name

Example Request

```
POST https://{endpoint}/v1/{project_id}/agent-run/workflows/{workflow_id}/conversations/{conversation_id}

Request Header:
Content-Type: application/json
X-Auth-Token: MIINRwYJKoZlHvcNAQcCoIINODCCDTQCAQExDTALBgIghkgBZQMEAgEwgggVBgkqhkiG...
stream: true
Request Body:
{
  "inputs": {
    "query": "Hello"
  },
  "plugin_configs": [
    {
      "plugin_id": "xxxxxxxx",
      "config": {
        "key": "value"
      }
    }
  ]
}
```

Example Response

Non-streaming (with stream set to false in the header)

Messages returned in the input node

```
{
  "conversation_id": "2c90493f-803d-431d-a197-57543d414317",
  "messages": [
    {
      "role": "assistant",
      "content": "{\n  \"inputs\": [\n    {\n      \"actualType\": \"string\", \"sourceType\": \"null\", \"description\": \"name\", \"name\": \"name\", \"type\": \"string\", \"required\": true\n    }\n  ]\n}",
      "nodeId": "node_1745928389632",
      "nodeType": "Input",
      "nodeName": "Input"
    }
  ],
  "status": {
    "code": 3,
    "desc": "waiting"
  },
  "start_time": 1734336269313,
  "end_time": 1734336270908
}
```

Messages returned in the questioner node

```
{
  "conversation_id": "f9a5540f-0c92-4f28-bd6e-f96ce04f5cc81",
  "messages": [
    {
      "role": "assistant",
      "content": "Please provide your name and age.",
      "nodeId": "node_1745929628452",
      "nodeType": "Questioner",
      "nodeName": "Questioner"
    }
  ],
  "status": {
    "code": 3,
    "desc": "waiting"
  },
  "start_time": 1745929778250,
  "end_time": 1745929779951
}
```

Messages returned in the end node

```
{
  "conversation_id": "2c90493f-803d-431d-a197-57543d414317",
  "outputs": {
    "responseContent": "Hello. How can I help you?"
  },
  "messages": [],
  "status": {
    "code": 1,
    "desc": "succeeded"
  },
  "start_time": 1734337068533,
  "end_time": 1734337082545
}
```

Streaming (with stream set to true or not specified in the header)

Messages returned in the input node

```
data:{"event":"workflow_started","data":{"start_time":1745929087614}}

data:{"event":"message","data":{"text":{"inputs":{"actualType":"string","sourceType":"null","description":"name","name":"name","type":"string","required":true}},"index":0,"node_id":"node_1745928389632","node_type":"Input","node_name":"Input"}}

data:{"event":"message","data":{"text":"","node_id":"node_1745928389632","node_type":"Input","node_name":"Input","is_finished":true}}

data:{"event":"end"}
```

Messages returned in the questioner node

```
data:{"event":"workflow_started","data":{"start_time":1745929709955}}

data:{"event":"message","data":{"text":"Please provide your name and age","index":0,"node_id":"node_1745929628452","node_type":"Questioner","node_name":"Questioner"}}

data:{"event":"message","data":{"text":"","node_id":"node_1745929628452","node_type":"Questioner","node_name":"Questioner","is_finished":true}}

data:{"event":"end"}
```

Messages returned in the end node

```
data:{"event":"workflow_started","data":{"start_time":1745929897770}}

data:{"event":"message","data":{"text":"","index":0,"node_id":"node_end","node_type":"End","node_name":"End"}}
```

```
data:{"event":"message","data":
{"text":"Hello","index":1,"node_id":"node_end","node_type":"End","node_name":"End"}}

data:{"event":"message","data":
{"text":"!", "index":2,"node_id":"node_end","node_type":"End","node_name":"End"}}

data:{"event":"message","data":{"text":"What can I do for
you?","index":3,"node_id":"node_end","node_type":"End","node_name":"End"}}

data:{"event":"message","data":{"text":"","node_id":"node_end","node_type":"End","node_name":"
end","is_finished":true}}

data:{"event":"workflow_finished","data":{"status":{"code":1,"desc":"succeeded"},"outputs":
{"responseContent":"","Hello! Is there anything I can help you
with?","start_time":1745929897770,"end_time":1745929898600}}

data:{"event":"end"}
```

Status Codes

For details, see [Status Codes](#).

Error Codes

For details, see [Error Codes](#).

3.4 Token Calculator

Function

To help you manage tokens more effectively and optimize token usage, the platform provides a token calculator. The token calculator evaluates the number of tokens in the specified text before model inference, estimate the cost, and optimize the data preprocessing policy.

URI

POST /v1/{project_id}/deployments/{deployment_id}/caltokens

For details about how to obtain the URI, see [Request URI](#).

Table 3-37 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain a project ID, see Obtaining the Project ID .
deployment_id	Yes	String	Model deployment ID. For details about how to obtain the deployment ID, see Obtaining the Model Deployment ID .

Request Parameters

Table 3-38 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. Used to obtain the permission required to call APIs. The token is the value of X-Subject-Token in the response header in Authentication .
Content-Type	Yes	String	MIME type of the request body. The value is application/json .

Table 3-39 Request body parameters

Parameter	Mandatory	Type	Description
data	Yes	List<String>	Character string of the number of tokens to be counted. The list length must be an odd number.
with_prompt	No	Boolean	Whether to calculate only the number of tokens for input characters. true : Only the number of tokens for the input character string is calculated. false : Total number of tokens for input character string and characters generated during inference.

Response Parameters

Table 3-40 Response body parameters

Parameter	Type	Description
tokens	List<String>	Token list split from the text.
token_number	Integer	Total number of tokens.

Example Request

```
{
  "data": [
    "Hello, please introduce Xi'an."
  ],
  "with_prompt": true
}
```

Example Response

```
{
  "tokens": [
    "Hello",
    ".",
    "Please",
    "introduce",
    "Xi'an",
    "."
  ],
  "token_number": 6
}
```

Status Codes

For details, see [Status Codes](#).

Error Codes

For details, see [Error Codes](#).

4 Appendix

4.1 Status Codes

HTTP status codes are three-digit codes that can be categorized into five classes: 1xx (informational responses), 2xx (successful responses), 3xx (redirection), 4xx (client errors), and 5xx (server errors).

The following table lists the common status codes.

Status Code	Message	Description
100	Continue	The client should proceed with the request. This provisional response informs the client that part of the request has been received and has not yet been rejected by the server.
101	Switching Protocols	Switching protocols. The target protocol must be more advanced than the source protocol. For example, the current HTTPS protocol is switched to a later version.
200	OK	The server has successfully processed the request.
201	Created	The request has been fulfilled and has resulted in one or more new resources being created.
202	Accepted	The request has been accepted, but the processing has not been completed.
203	Non-Authoritative Information	Non-authoritative information. The request is successful.

Status Code	Message	Description
204	No Content	The request has been fulfilled, but the HTTP response does not contain a response body. The status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed parts of the GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which a user client (such as a browser) can choose the most appropriate one.
301	Moved Permanently	The requested resource has been assigned a new permanent URI, and the new URI is contained in the response.
302	Found	The requested resource has been temporarily moved to a new location.
303	See Other	The response to the request can be found under a different URI, and should be retrieved using the GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, it does not return any resources.
305	Use Proxy	The requested resource must be accessed through a proxy.
306	Unused	This HTTP status code is no longer used.
400	Bad Request	Invalid request. The client should not repeat the request without modifications.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.

Status Code	Message	Description
403	Forbidden	The request has been rejected. The server understood your request but refuses to authorize it, meaning you do not have permission to access the requested resource. The client should not repeat the request without modifications.
404	Not Fou	The requested resource could not be found. The client should not repeat the request without modifications.
405	Method Not Allowed	The method received in the request is not supported by the target resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server cannot fulfill the request according to the content characteristics of the request.
407	Proxy Authentication Required	The request cannot be fulfilled because the client needs to authenticate with a proxy server before accessing the desired resource. This is similar to 401.
408	Request Timeout	The server didn't receive a complete request message within the time that it was prepared to wait. The client may repeat the request without modifications at any time later.
409	Conflict	The server cannot fulfill the request due to a conflict with the current state of the resource. This status code indicates that the resource that the client attempts to create already exists, or the requested update failed due to a conflict.
410	Gone	The requested resource is no longer available. The status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server refuses to process the request without a defined Content-Length .
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.

Status Code	Message	Description
413	Request Entity Too Large	The server refuses to process the request because the payload size is too large. The server may close the connection to prevent the client from continuously sending the request. If the server cannot process the request temporarily, the response will contain a Retry-After header field.
414	Request URI Too Long	The request URI is too long for the server to process.
415	Unsupported Media Type	The server is unable to process the media format in the request.
416	Requested Range Not Satisfiable	The requested range is invalid.
417	Expectation Failed	The server fails to meet the requirements of the Expect request-header field.
422	Unprocessable Entity	The request is well-formed but cannot be processed due to semantic errors.
429	Too Many Requests	The client has sent excessive number of requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server has received an excessive number of requests within a given time (beyond its processing capability). In this case, the client should resend the request after the time specified in the Retry-After header of the response has elapsed.
500	Internal Server Error	The server is able to receive the request but unable to understand it.
501	Not Implemented	The server does not support the functionality required to fulfill the request.
502	Bad Gateway	The server is acting as a gateway or proxy and receives an invalid request from a remote server.
503	Service Unavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	Gateway Timeout	The request cannot be fulfilled within a given time. This status code is returned to the client only when the Timeout parameter is specified in the request.

Status Code	Message	Description
505	HTTP Version Not Supported	The server does not support the HTTPS protocol version used in the request.

4.2 Error Codes

If an error code starting with **APIGW** is returned after you call an API, resolve the problem by referring to [Error Codes](#). If an error code starts with **APIG**, rectify the fault by referring to this document.

Table 4-1 Error Codes

Module	Error Code	Error Message	Description	Solution
Model inference	PANGU.0010	parameter illegal.	The request parameter is incorrect.	Enter correct request parameters by referring to the API document and debug the API again.
	PANGU.0011	Authentication failed.	Authentication failed.	Authentication failed. For details, see Authentication in the API document.
	PANGU.0012	The authentication information is missing.	Identity authentication information is unavailable.	Check whether the authentication information is provided when the API is called.
	PANGU.0031	Inner service exception.	Internal error.	Contact technical support for assistance.
	PANGU.3254	The requested inference service does not exist.	The resource does not exist.	Check whether projectId and deploymentId are correctly set when the API is called and whether the inference service is available.
	PANGU.3267	The number of service invoking requests exceeds the project limit.	Too frequent API requests.	Reduce your request frequency.

Module	Error Code	Error Message	Description	Solution
	PANGU.3278	required api parameter is not present.	The request parameter is lost.	Check whether the request parameters are complete, whether the spelling is correct, and whether the values are correct.
	PANGU.3318	The total length of the question should be between 1 and 4096.	The length of Content is invalid.	Check whether the length of the Content parameter value in the request is within the allowed range by referring to the API document and debug the API again.
	PANGU.3320	The parameter [n] can only be 1 or 2 when calling non-streaming.	The value of n used for a non-streaming call of the inference service must be 1 or 2.	Use the correct value 1 or 2.
	PANGU.3321	The parameter [n] can only be 1 when calling streaming.	The value of n used for a streaming call of the inference service must be 1.	Use the correct value 1.
	PANGU.3342	Failed to invoke the inference service. please check the details field.	Failed to call the inference service. Check the error details.	Failed to call the inference service. Check the error details.
	IIT.0201	The input param is invalid!/The input param is invalid, please check your key!	The request parameter is invalid.	Check whether the request parameters are correct.
	IIT.0202	Interval Server Error!	An internal error occurred.	Contact technical support for assistance.

Module	Error Code	Error Message	Description	Solution
	IIT.0203	The input param is invalid, the input data lens is less than the train data lens!	The request parameter is invalid. The data length in the input parameter is less than that used for training.	Check that the feature name and number of features in the request body are the same as those in the training data.
	PREDICT.0102	"Json format is wrong!" or other data-related errors.	The request data is not in JSON format. Other data-related errors occur.	Set the request body to be in JSON format. Adjust the request body based on information about the data-related errors.
	PREDICT.0201	The input param is invalid!/The input param is invalid, please check your key!	The request parameter is invalid.	Check whether the request parameters are correct.
	PREDICT.0202	Interval Server Error!	An internal error occurred.	Contact technical support for assistance.
	PREDICT.0203	The input param is invalid, the input data lens is less than the train data lens!	The request parameter is invalid. The data length in the input parameter is less than that used for training.	Check that the feature name and number of features in the request body are the same as those in the training data.
	APIG.0101	The API does not exist or has not been published in the environment.	The API does not exist or has not been published.	<ul style="list-style-type: none">• Check whether the API URL is correct. For example, check whether the project ID is included in the URL.• Check whether the HTTP request method (such as POST or GET) is correct.

Module	Error Code	Error Message	Description	Solution
	APIG.0201	Backend timeout.	Request timed out.	<ul style="list-style-type: none">• Check whether the API call requests are initiated too frequently. If so, check the return value in the code and resend the requests later (for example, 2 to 5 seconds later). You can also check in the backend whether the result of the previous request is returned. After the result of the previous request is returned, send the next request.• Confirm with technical support to check whether the API has been deployed.

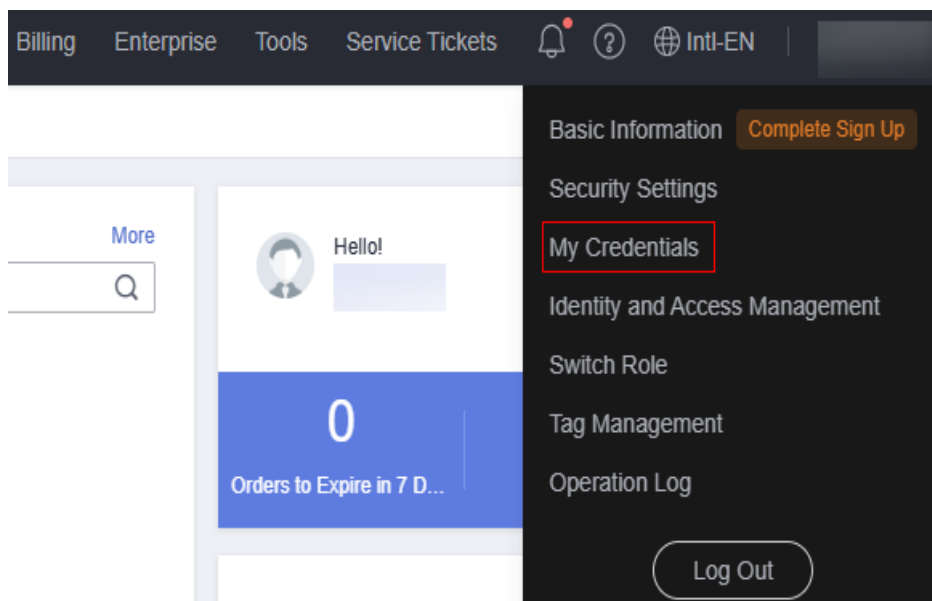
Module	Error Code	Error Message	Description	Solution
	APIG.0301	Incorrect IAM authentication information.	<p>The IAM authentication information is incorrect.</p> <ul style="list-style-type: none">• decrypt token fail: The token fails to be parsed.• token expires: The token has expired.• verify aksk signature fail: The AK/SK authentication fails.• x-auth-token not found: The x-auth-token parameter is not found.	<ul style="list-style-type: none">• If the token fails to be parsed, check the method for obtaining the token, whether the request body is correct, whether the token is correct, and whether the environment for obtaining the token is the same as the environment for calling the API.• If the token has expired, obtain a new token that is valid permanently.• Check whether the AK/SK pair is correct. For example, check whether the SK for the AK is correct and whether an extra space is included in the AK/SK pair.• AK/SK-based authentication errors occur frequently. If an AK/SK pair fails to be authenticated for more than five consecutive times, the AK/SK pair is locked for 5 minutes (the AK/SK-based authentication is considered as an abnormal authentication request within 5 minutes). After 5 minutes, the AK/SK pair is unlocked and re-authenticated.• Check that the spelling of the value for X-Auth-Token in

Module	Error Code	Error Message	Description	Solution
				the request header for an API call is correct.
	APIG.0308	The throttling threshold has been reached: policy user over ratelimit,limit:XX,time:1 minute.	The request exceeds the default rate limit of the service.	<ul style="list-style-type: none">• Use the retry mechanism to rectify the fault by checking the return value in the code and retrying the requests after a short period of time (for example, 2 to 5 seconds).• Check in the backend whether the result of the previous request has been returned. If it has, send the next request. This helps prevent excessively frequent requests.

4.3 Obtaining the Project ID

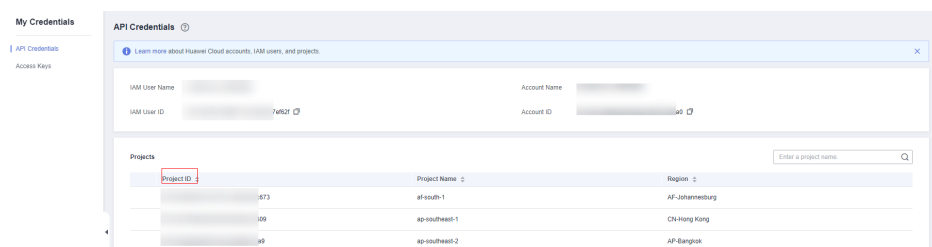
Obtaining the Project ID from the Console

1. Log in to the [management console](#).
2. Click the username in the upper right corner and select **My Credentials** from the drop-down list.

Figure 4-1 My Credentials

3. On the **My Credentials** page, obtain the project ID, account name, account ID, IAM username, and IAM user ID.

When calling a Pangu API, ensure that the project ID you obtained belongs to the region where the Pangu service is deployed. For example, if the PanguLM is deployed in the CN-Hong Kong region, obtain the project ID that belongs to the CN-Hong Kong region.

Figure 4-2 Viewing the project ID

If there are multiple projects, unfold the target region and obtain the project ID from the **Project ID** column.

Obtaining the Project ID by Calling an API

The API for obtaining a project ID is **GET <https://{Endpoint}/v3/projects>**. *{Endpoint}* indicates the endpoint of IAM. For details about API authentication, see [Authentication](#).

Here is an example response, where **id** indicates the project ID.

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "project_name",
      "description": ""
    }
  ]
}
```

```
{
  "links": {
    "next": null,
    "previous": null,
    "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
  },
  "id": "a4a5d4098fb4474fa22cd05f897d6b99",
  "enabled": true
},
{
  "links": {
    "next": null,
    "previous": null,
    "self": "https://www.example.com/v3/projects"
  }
}
```

4.4 Obtaining the Model Deployment ID

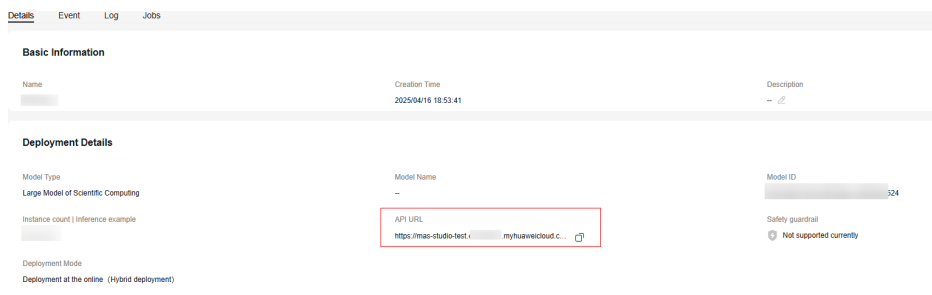
To obtain the model deployment ID, perform the following steps:

Step 1 Log in to ModelArts Studio.

Step 2 Obtain the model deployment ID.

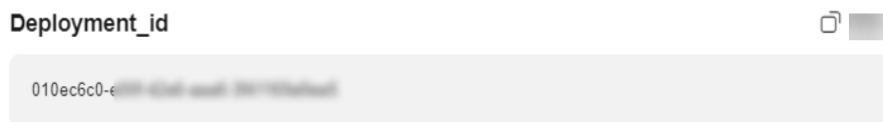
- To call a deployed model, choose **Model Development > Model Deployment** in the navigation pane on the left. On the **My Service** tab page, click the model name in the model deployment list. On the **Details** tab page, obtain the deployment ID of the model.

Figure 4-3 Calling a deployed model



- To call a preset model, choose **Model Development > Model Deployment** in the navigation pane on the left, click **Call Path** in the model list on the **Preset service** page, and obtain the deployment ID of the model.

Figure 4-4 Deployment ID of a preset model



----End