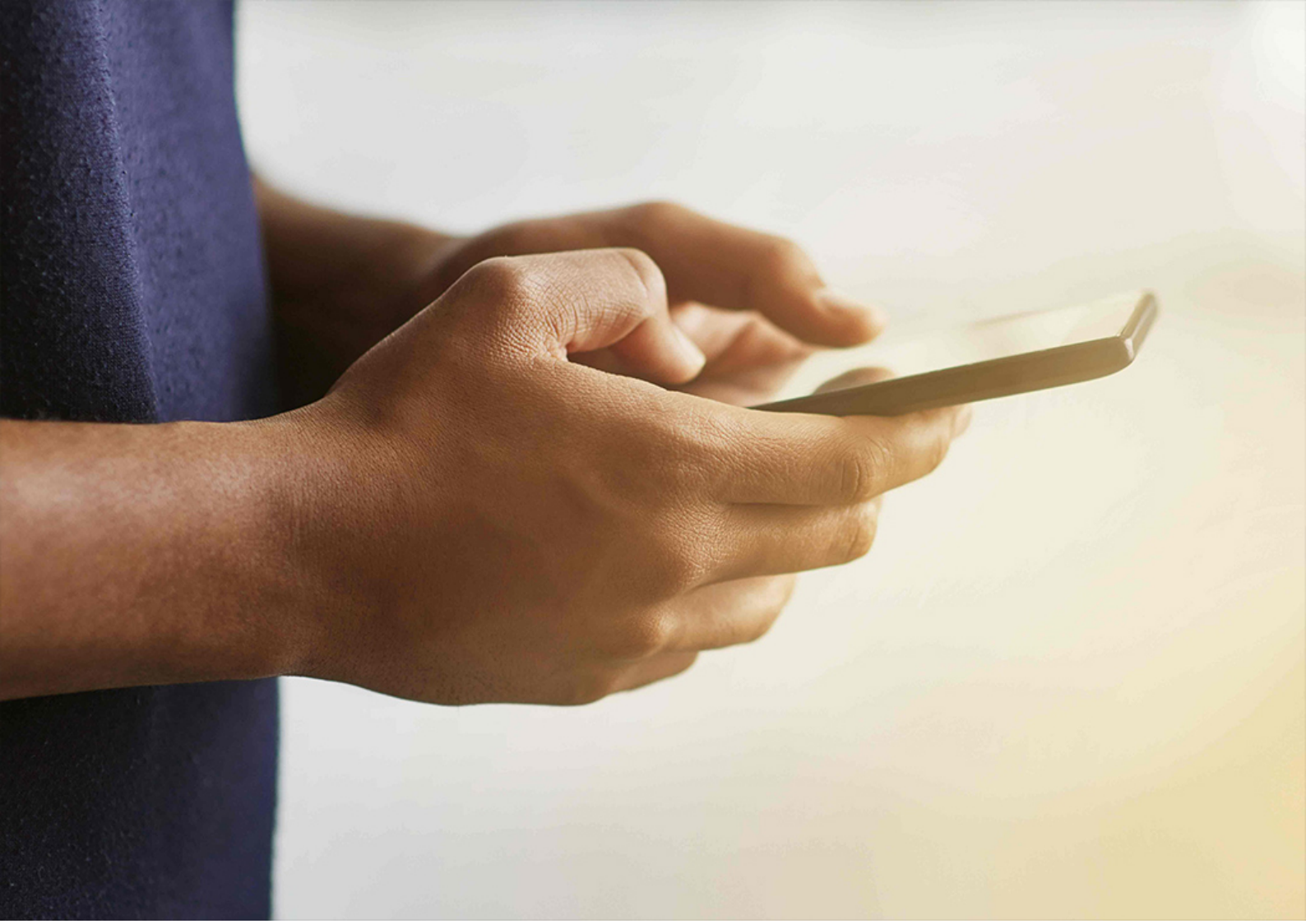


INTERNAL
2023-11-23

# Generative AI Hub

© 2023 SAP SE or an SAP affiliate company. All rights reserved.

THE BEST RUN SAP

# Content


**1 Generative AI Hub in SAP AI Core. . . . 4**

**2 Service Plans. . . . 7**

2.1 Metering and Pricing. . . . 7

2.2 Add the Internal Plan for LLMs to the Global Account. . . . 8

2.3 Update an Existing Plan of SAP AI Core to Include LLMs. . . . 9

**3 Initial Setup. . . . 11**

3.1 Create a Subaccount. . . . 12

3.2 Enable Cloud Foundry. . . . 14

3.3 Create a Space. . . . 15

3.4 Add a Service Plan. . . . 16

3.5 Create a Service Instance. . . . 18

3.6 Create a Service Key. . . . 22

3.7 Use a Service Key. . . . 24

Using SAP AI Launchpad. . . . 25

Using Postman. . . . 26

Using Curl. . . . 27

**4 Tutorials. . . . 29**

**5 Activating the Generative AI Hub. . . . 30**

5.1 Activate the Generative AI Hub for SAP AI Core. . . . 30

Using the API. . . . 30

5.2 Activate the Generative AI Hub for SAP AI Launchpad. . . . 35

5.3 Model Lifecycle. . . . 42

**6 Consuming Large Language Models. . . . 43**

6.1 Consume Large Language Models Using SAP AI Core. . . . 43

Summarizing. . . . 44

Inferencing. . . . 47

Transformations. . . . 53

Expansions. . . . 58

Chatbot. . . . 60

6.2 Consume Large Language Models Using SAP AI Launchpad. . . . 62

Question Answering. . . . 64

Summarizing. . . . 64

Inferencing. . . . 67

Transformations. . . . . 73
Expansions. . . . . 80

**7 Stopping or Deleting a Virtual Deployment. . . . . 83**
7.1 Stop or Delete a Virtual Deployment in SAP AI Core. . . . . 83
Stop a Virtual Deployment. . . . . 83
Delete a Virtual Deployment. . . . . 83
7.2 Stop or Delete a Virtual Deployment in SAP AI Launchpad. . . . . 84
Stop a Virtual Deployment. . . . . 84
Delete a Virtual Deployment. . . . . 84

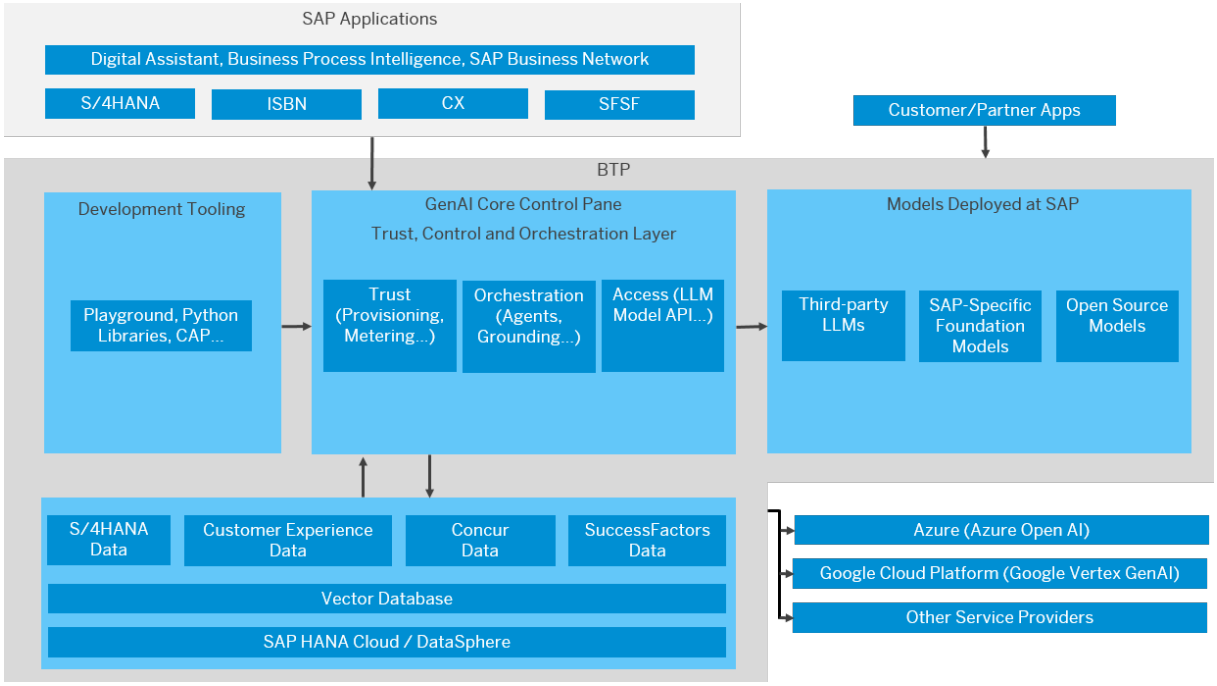
# 1 Generative AI Hub in SAP AI Core

The generative AI hub incorporates large language models (LLMs) into your AI activities in SAP AI Core and SAP AI Launchpad.

LLMs are self-supervised, deep learning models that have been trained on vast amounts of unlabeled data. They leverage AI technology and industrial-scale computational resources to learn complex language patterns and semantic knowledge bases for natural language processing (NLP) tasks. They parse input, such as prompts, and by predicting a target word, can return contextually relevant responses written in natural language. A single LLM can perform multiple NLP tasks by using different input formats and output modes.

LLMs are general models but can be fine-tuned with additional embeddings for specialized or domain-specific use cases.

SAP AI Core and the generative AI hub help you to integrate LLMs and AI into new business processes in a cost-efficient manner.




Generative AI Hub Architecture Overview

## Access

You can use the generative AI hub through the API, using a tool such as curl or Postman, or through SAP AI Launchpad.

## Scenarios

Access to the large language models is provided under the global AI scenario `foundation-models`, which is managed by SAP AI Core. Individual models are provided as executables in the form of serving templates, and accessed by choosing the corresponding template for the desired model.

The following scenarios are available:

| Scenario Number | Global Scenario | Executable ID | Description |
|---|---|---|---|
| 1 | foundation-models | azure-openai | The Azure OpenAI Service provides REST API access to OpenAI's LLMs. |
| 2 | foundation-models | aicore-opensource | Opensource models hosted and accessed via SAP AI Core. |

## Models

The following models are supported:

| Executable ID | Model Name | Model Version | Deprecation (as Specified by Model Provider) | Region | Request Limit (Requests per Minute) |
|---|---|---|---|---|---|
| azure-openai | gpt-35-turbo | 0613 | 2025-07-05 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 120 |
| azure-openai | gpt-35-turbo-16k | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 96 |
| azure-openai | gpt-4 | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 18 |
| azure-openai | gpt-4-32k | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 78 |

| Executable ID | Model Name | Model Version | Deprecation (as Specified by Model Provider) | Region | Request Limit (Requests per Minute) |
|---|---|---|---|---|---|
| azure-openai | text-embedding-ada-002 | 2 | 2025-02-02 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 138 |
| aicore-open-source | tiiuae--falcon-40b-instruct | | | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | TBD |

> **Note**
>
> Rate limits are applied at tenant level.

Models from Azure OpenAI are accessed through a private instance of the `chat-completions API`. For more information, see Azure OpenAI Chat Completions API Documentation .

> **Note**
>
> The following topics are out of the scope of this document:
>
> - Advanced consumption patterns such as working with a textual knowledge base (such as embeddings)
> - Complex orchestration of LLM calls
> - Training own models

## Related Information

Azure Chat Completions Documentation
Tiiuae Falcon 40b Instruct Documentation

# 2 Service Plans

The SAP AI Core service plan you choose determines pricing, conditions of use, resources, available services, and hosts. The generative AI hub in SAP AI Core is available only through the `sap-internal` service plan.

> ⚠ Caution
>
> The `sap-internal` service plan is available only for internal consumption and on eu10 canary.

The `sap-internal` service plan enhances the capabilities provided in the `standard` service plan. Specifically, it provides access to the `foundation-models` global AI scenario. This scenario, which is managed by SAP AI Core, includes serving templates for deployments with integrated LLM access.

If you're new to SAP AI Core, choose the `sap-internal` service plan during your initial setup. For more information, see Add the Internal Plan for LLMs to the Global Account [page 8].

If you already have an SAP AI Core tenant on a `standard` or `free` tier service plan, you can update the service plan to `sap-internal`. For more information, see Update an Existing Plan of SAP AI Core to Include LLMs [page 9].

> → Tip
>
> If you update to the `sap-internal` service plan, you can still use your original service key.

## 2.1 Metering and Pricing

The use of large language models (LLMs) in the generative AI hub is metered using GenAI tokens and capacity units.

A GenAI token corresponds to a block of 1,000 tokens from the LLM service provider. Its cost varies depending on the model used and the type of token (input or output).

A capacity unit is the number of GenAI tokens multiplied by a fixed amount, and is used to calculate the monetary value of your LLM usage. The fixed amount is currently 2.6925 (subject to change).

The following table provides the conversion rates between tokens from the LLM service provider and GenAI tokens. The rates apply to blocks of 1,000 input and output tokens. You can refer to these values to calculate the total number of GenAI tokens that you consume. You then multiply the number of consumed tokens by the fixed-rate capacity unit to obtain the monetary value.

> ℹ Note
>
> Values indicated are subject to change.

| Model | GenAI Input Tokens (for 1,000 Tokens) | GenAI Output Tokens (for 1,000 Tokens) |
|---|---|---|
| GPT-35-Turbo | 0.00094 | 0.00122 |
| GPT-35-Turbo-16K | 0.00180 | 0.00238 |
| GPT-4 | 0.01735 | 0.03462 |
| GPT-4-32K | 0.03462 | 0.06917 |
| text-embedding-ada-002 | 0.00013 | 0.00000 |
| tiiuae--falcon-40b-instruct | TBD | TBD |

Example

Model: GPT-35-Turbo
LLM provider input tokens: 1,000
LLM provider output tokens: 100,000

To calculate the GenAI **input** tokens, we divide the total number of input tokens from our LLM provider (1,000) by 1,000 and multiply by the conversion rate (0.00094): (1,000/1,000) * 0.00094 = 0.00094.

To calculate the GenAI **output** tokens, we divide the total number of output tokens from our LLM provider (100,000) by 1,000 and multiply by the given conversion rate (0.00122): (100,000/1,000) * 0.00122 = 0.122.

This gives us the total number of consumed GenAI tokens as 0.00094 + 0.122 = 0.12294. By multiplying this value with the capacity value (2.6925), we obtain the number of capacity units: 0.12294 * 2.6925 = 0.331.

If the price for one capacity unit is €1, the total amount consumed equates to €0.331.

## 2.2 Add the Internal Plan for LLMs to the Global Account

Large language models (LLMs) are available only with the `sap-internal` service plan, which you configure in your global account during the initial setup phase.

### Prerequisites

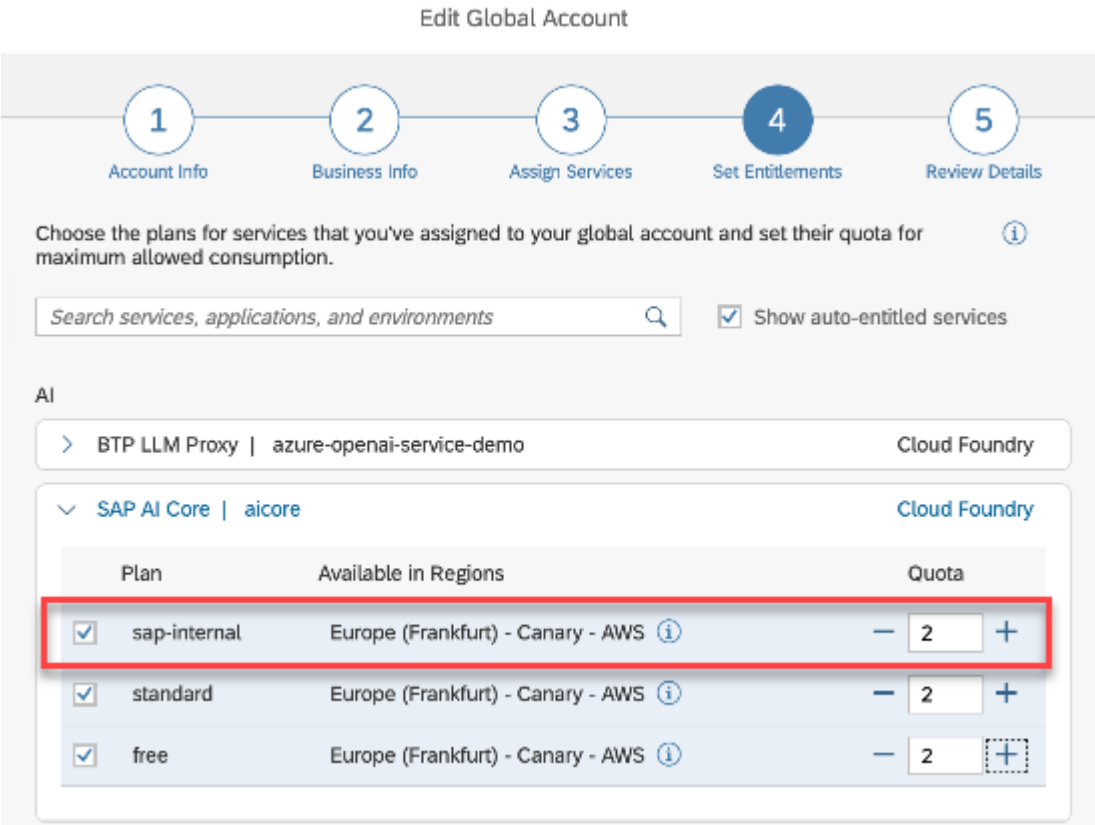
- You're a new user of SAP AI Core.

### Context

To add the `sap-internal` plan, set the quota in your global account.

## Procedure

Complete the initial setup for SAP AI Core as described in Add the Internal Plan for LLMs to the Global Account [page 8]. When you set your entitlements in step 5 [page 16], include the `sap-internal` service plan and set the applicable quota.




> **i Note**
>
> The `sap-internal` plan includes the features of the `standard` plan, with the addition of the LLM capabilities.

# 2.3 Update an Existing Plan of SAP AI Core to Include LLMs

Large language models (LLMs) are available only with the `sap-internal` service place. You can switch to this service plan by updating your instance of SAP AI Core.

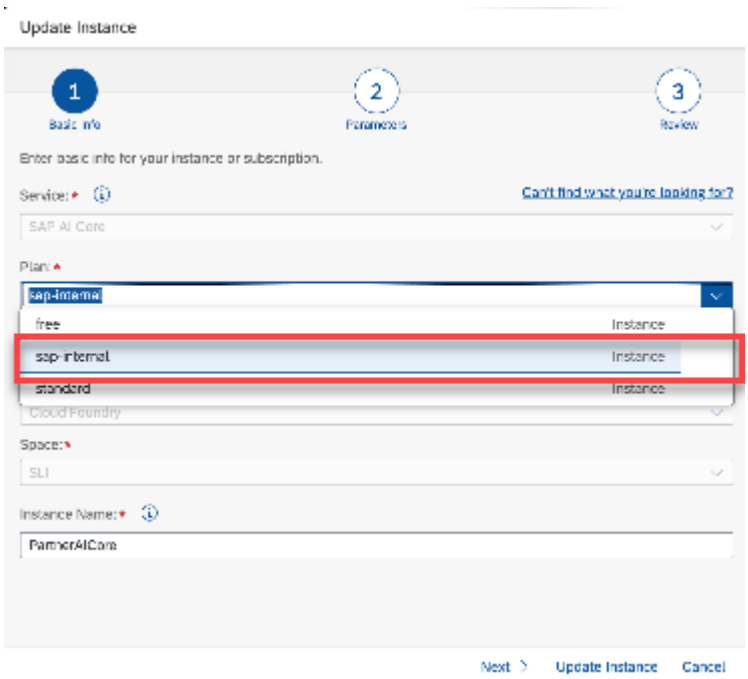
## Prerequisites

You're an existing user of SAP AI Core.

> **! Restriction**
>
> If you have an existing `standard` plan, it is **not** possible to create a new instance with the `sap-internal` plan. You must update your standard instance to`sap-internal`.

## Procedure

1. In your SAP BTP cockpit, choose *Instances and Subscriptions*.
2. Under *Environments*, choose *Update* and add the `sap_internal` service plan.



# 3 Initial Setup

You provision SAP AI Core from the SAP BTP cockpit in SAP Business Technology Platform. After provisioning, you will have your service key, which provides URLs and credentials for accessing the SAP AI Core instance

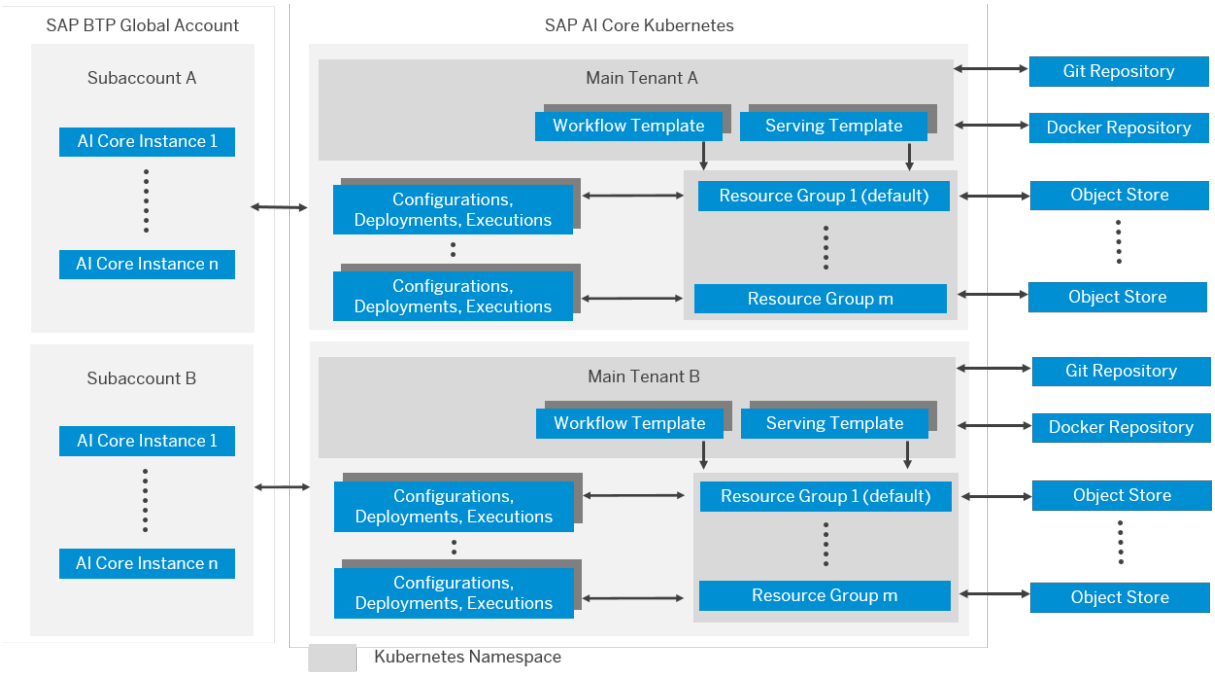
## Prerequisites

- Your SAP BTP administrator has access to a global account on SAP Business Technology Platform. For more information, see Getting a Global Account.
- Your SAP BTP administrator has set the entitlement to a subaccount.

## Context

The SAP AI Core service is a tenant-aware reuse service. It isolates tenants based on the ID of the zone (representing the subaccount). The SAP AI Core service instance is created within a subaccount. Each subaccount represents an SAP AI Core tenant.

> **i Note**
>
> The SAP AI Core service does not isolate tenants based on the service instance ID. If you create multiple service instances within the same subaccount, all of them will reference the same SAP AI Core tenant.



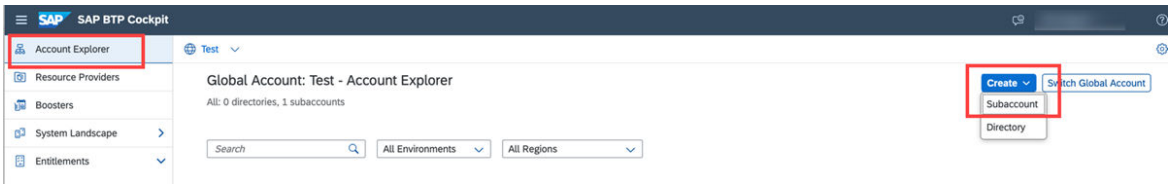
The steps below guide you through the provisioning procedure, Alternatively, a booster is available for both SAP AI Core and SAP AI Launchpad. For more information, see AI Boosters Tutorial. If you choose to use the booster, you can skip the remaining steps to SAP AI Core Starter Tutorials.


1. Create a Subaccount [page 12]
2. Enable Cloud Foundry [page 14]
3. Create a Space [page 15]
4. Add a Service Plan [page 16]
5. Create a Service Instance [page 18]
6. Create a Service Key [page 22]
7. Use a Service Key [page 24]

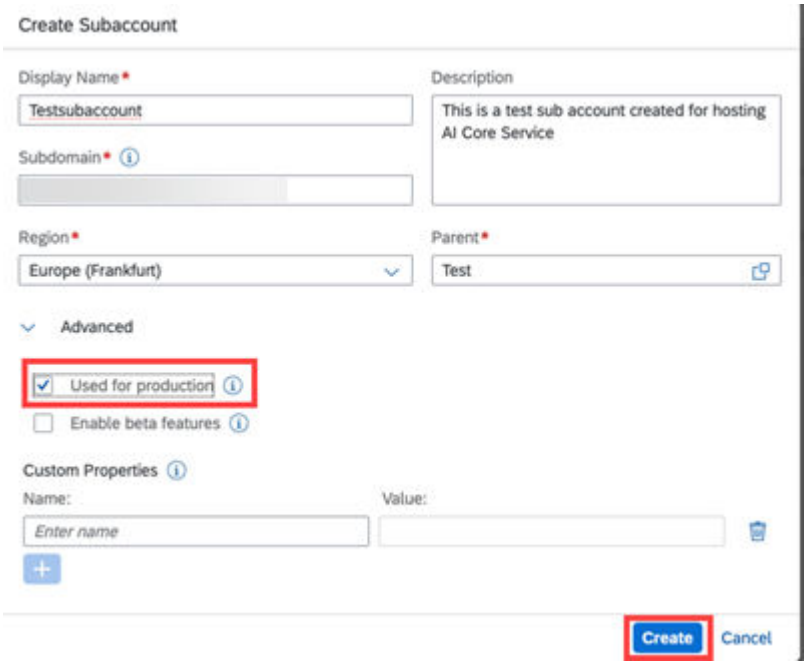

## 3.1 Create a Subaccount

### Procedure

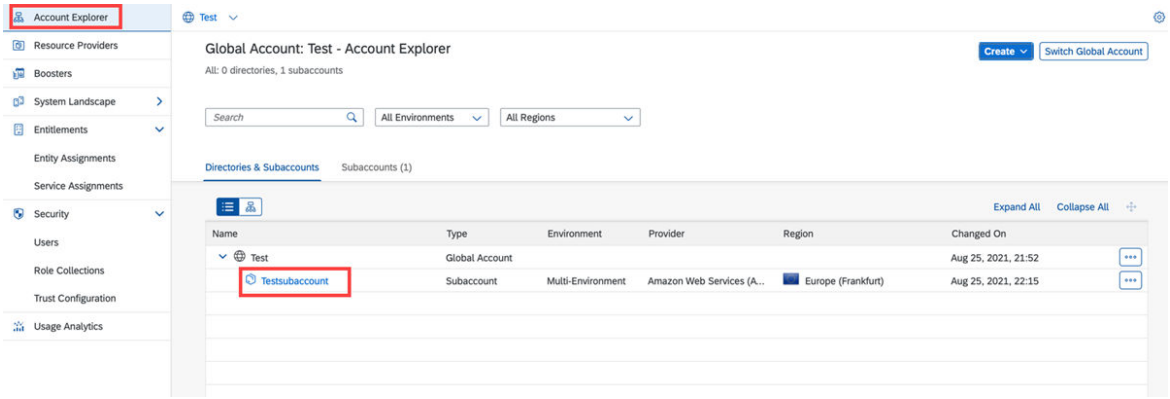
1. In SAP Business Technology Platform cockpit, choose *Account Explorer* and then click *Create* > *Subaccount* >.




2. In the *Create Subaccount* dialog, complete the mandatory information, select the *Used for production* checkbox under *Advanced*, and click *Create*.




3. Return to the *Account Explorer* to view your subaccount.



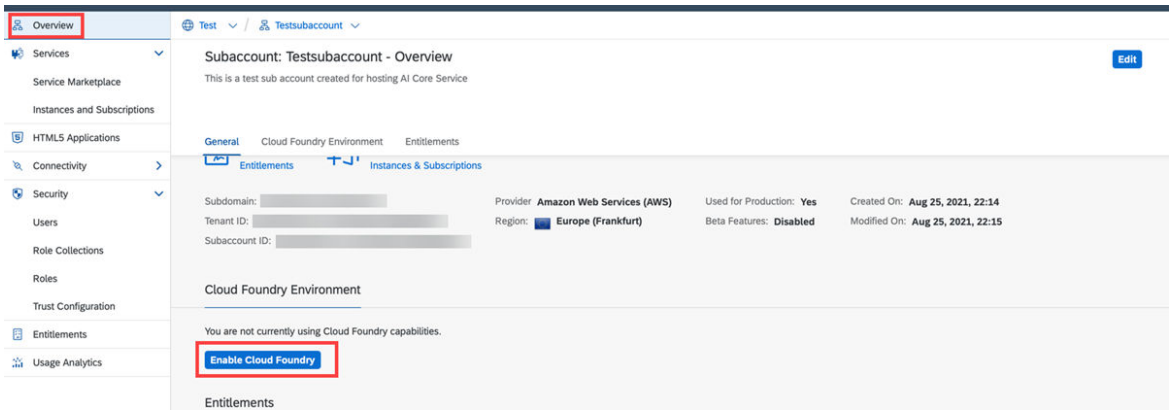

**Task overview:** Initial Setup [page 11]

**Next task:** Enable Cloud Foundry [page 14]

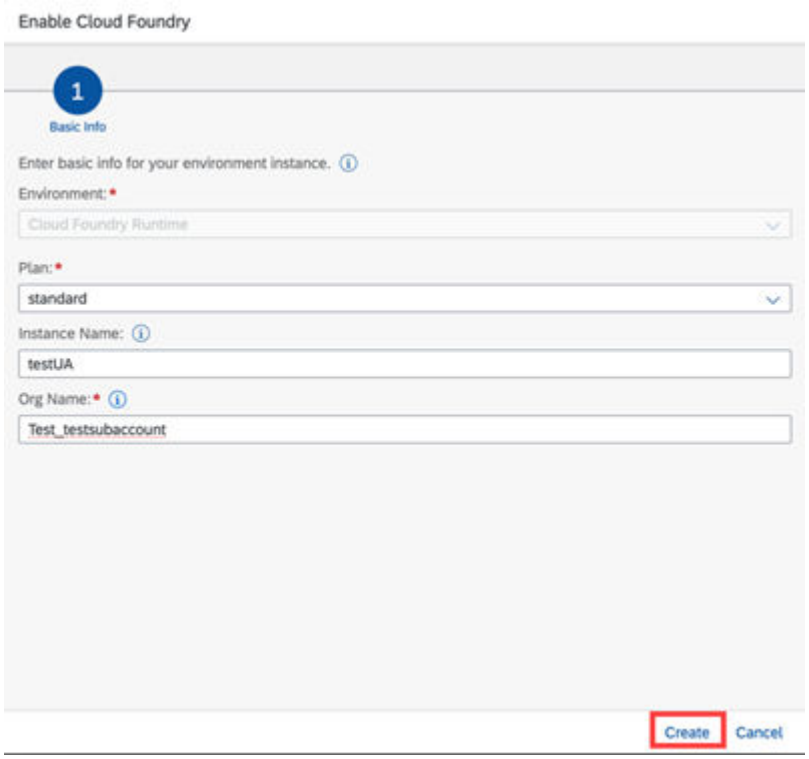
# 3.2 Enable Cloud Foundry

## Procedure

1. Choose your subaccount and on the *Overview* page, choose *Enable Cloud Foundry*.




2. Enter the basic information for your Cloud Foundry environment instance and click *Create*.




**Task overview:** Initial Setup [page 11]

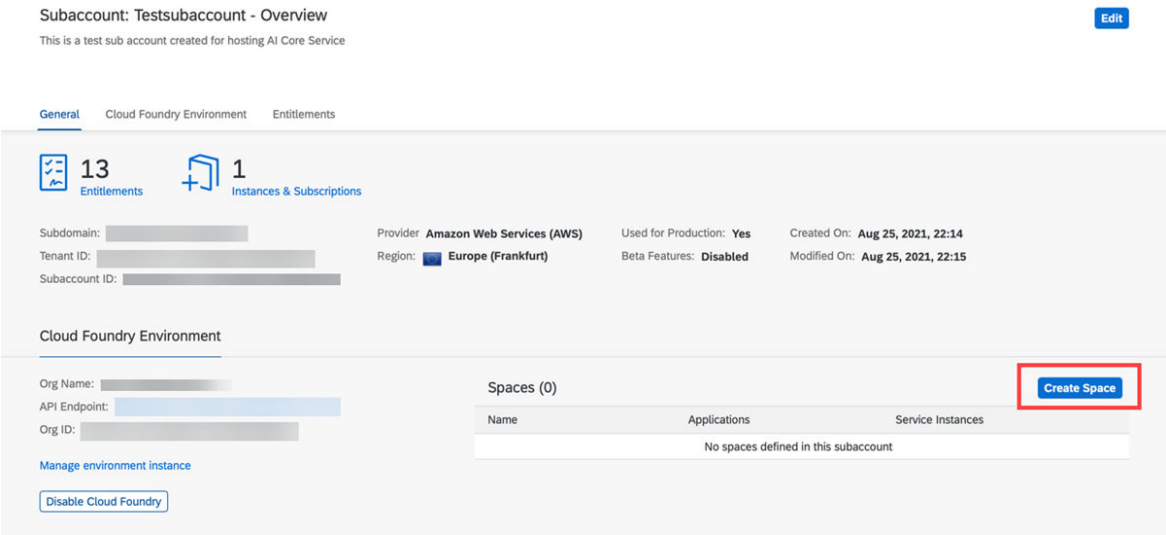
**Previous task:** Create a Subaccount [page 12]

**Next task:** Create a Space [page 15]

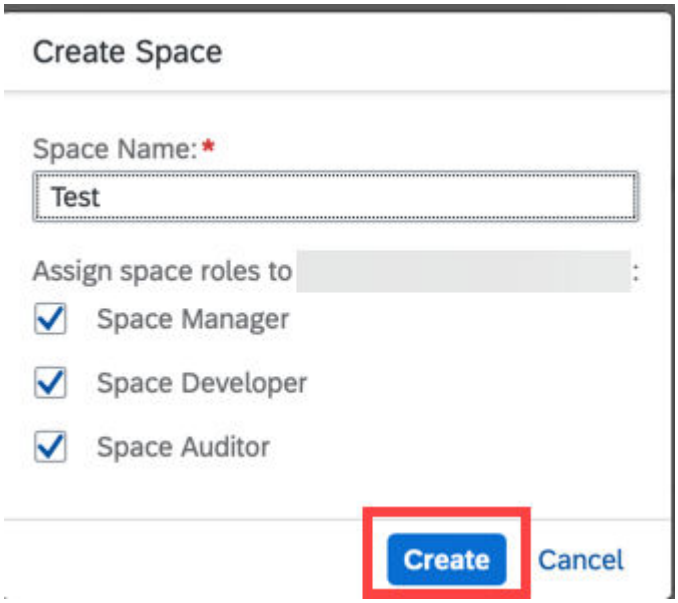
## 3.3 Create a Space

### Procedure

1. On the overview page for your subaccount, choose *Create Space*.




2. Enter a name for your space and assign space roles by selecting the checkboxes..



3. Click *Create*.

**Task overview:** Initial Setup [page 11]

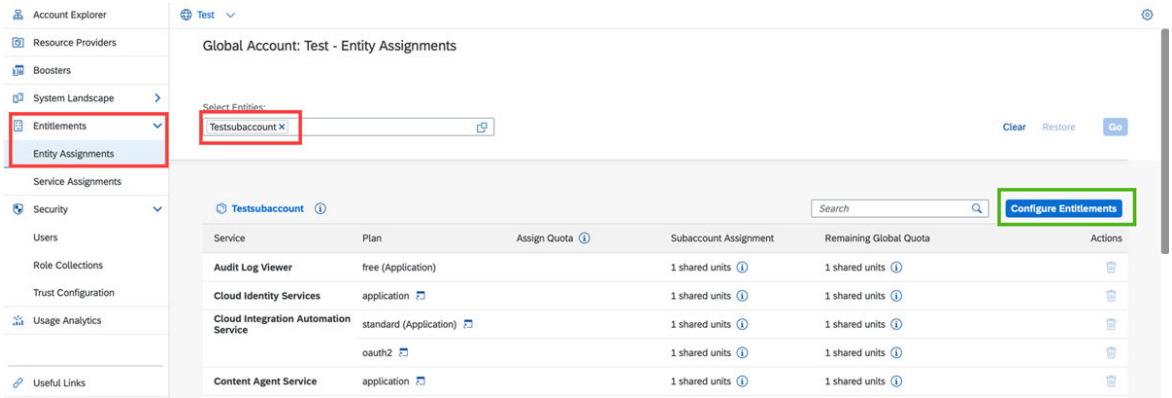
**Previous task:** Enable Cloud Foundry [page 14]

**Next task:** Add a Service Plan [page 16]

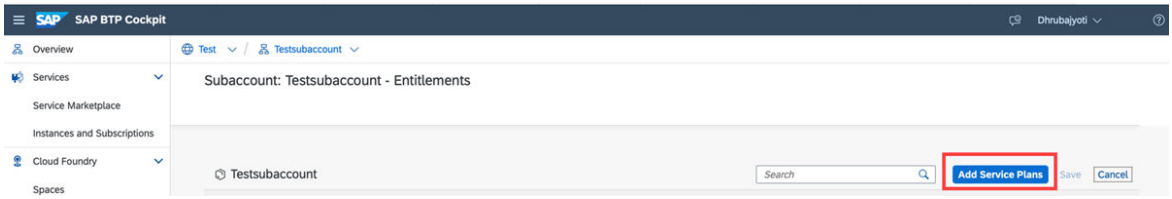
# 3.4 Add a Service Plan

## Procedure

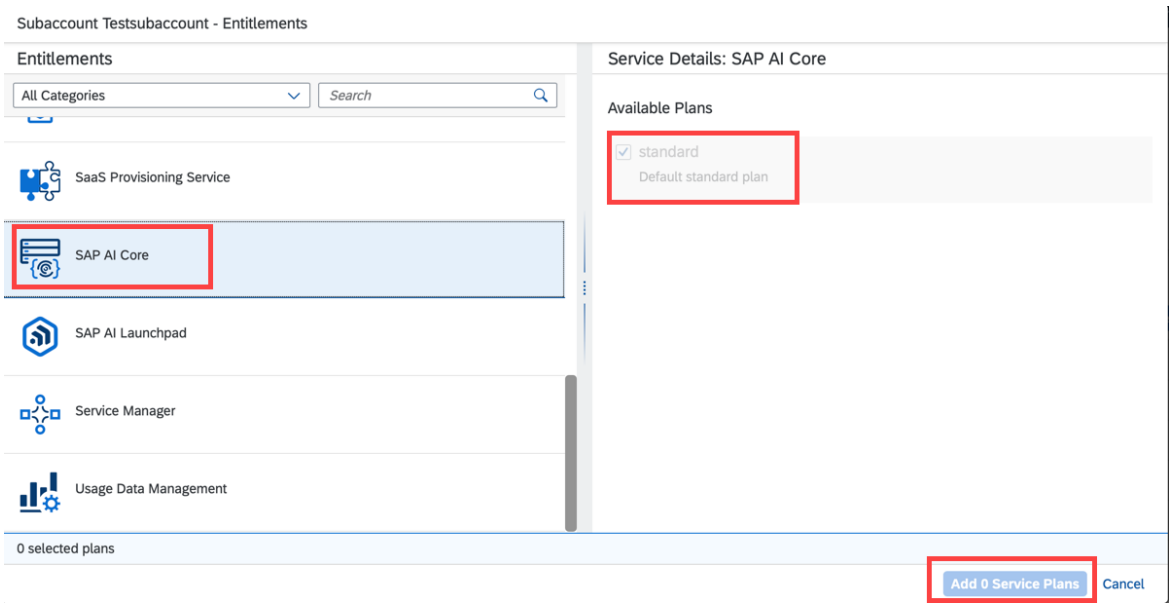
1. In SAP BTP cockpit, navigate to your global account and choose *Entitlements* and then *Entity Assignments*.
2. From the *Select Entities* box, select your subaccount and click *Select*.
3. Choose *Configure Entitlements*.



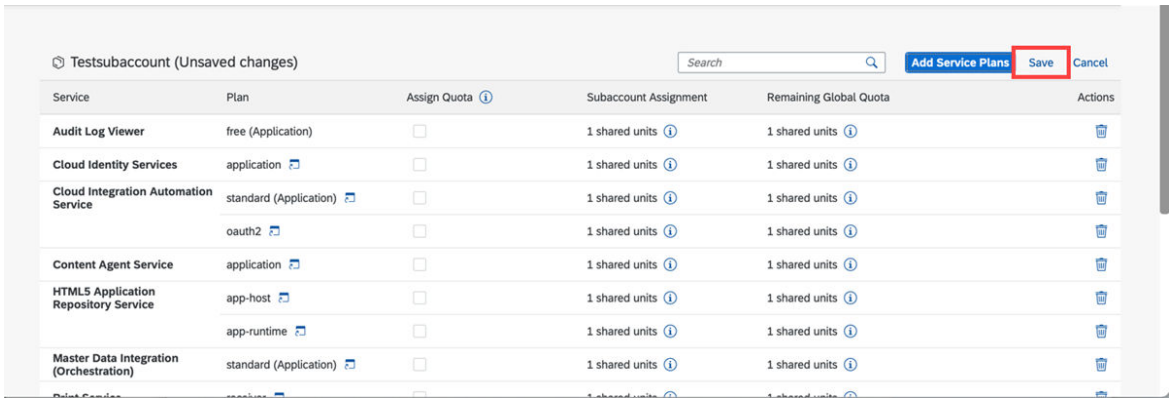

4. Choose *Add Service Plans*.




5. Select SAP AI Core and pick a service plan.




6. Save your changes.




**Task overview:** Initial Setup [page 11]

**Previous task:** Create a Space [page 15]

**Next task:** Create a Service Instance [page 18]

## Related Information

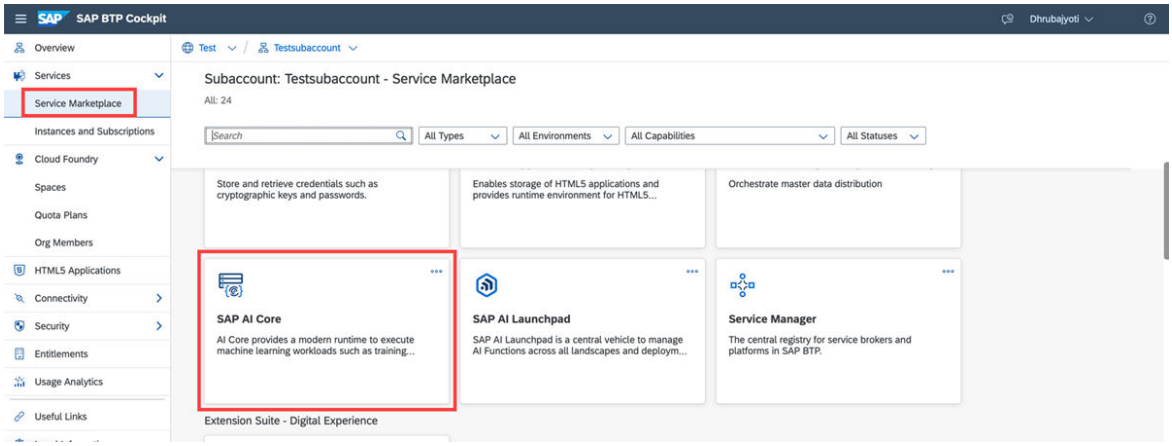
Service Plans

# 3.5 Create a Service Instance

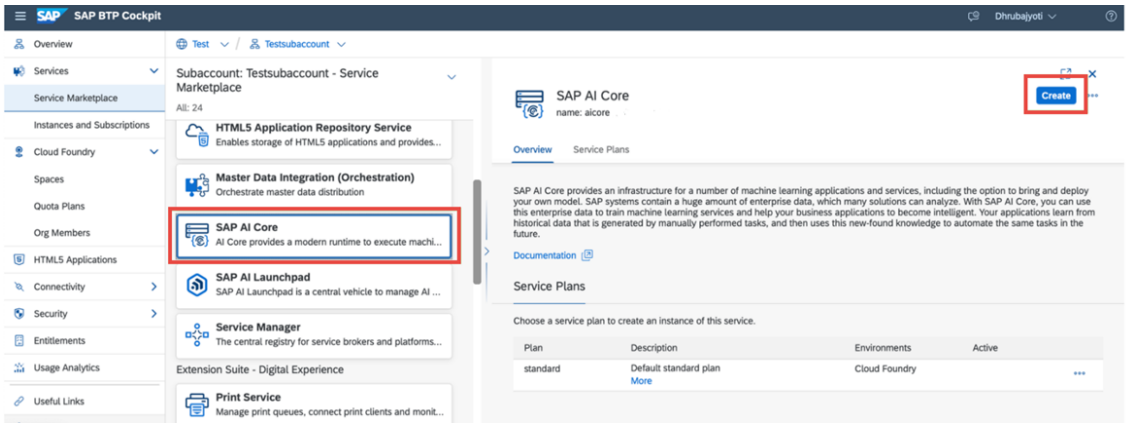
## Procedure

1. SAP BTP cockpit, navigate to your global account, and choose *Service Marketplace*.

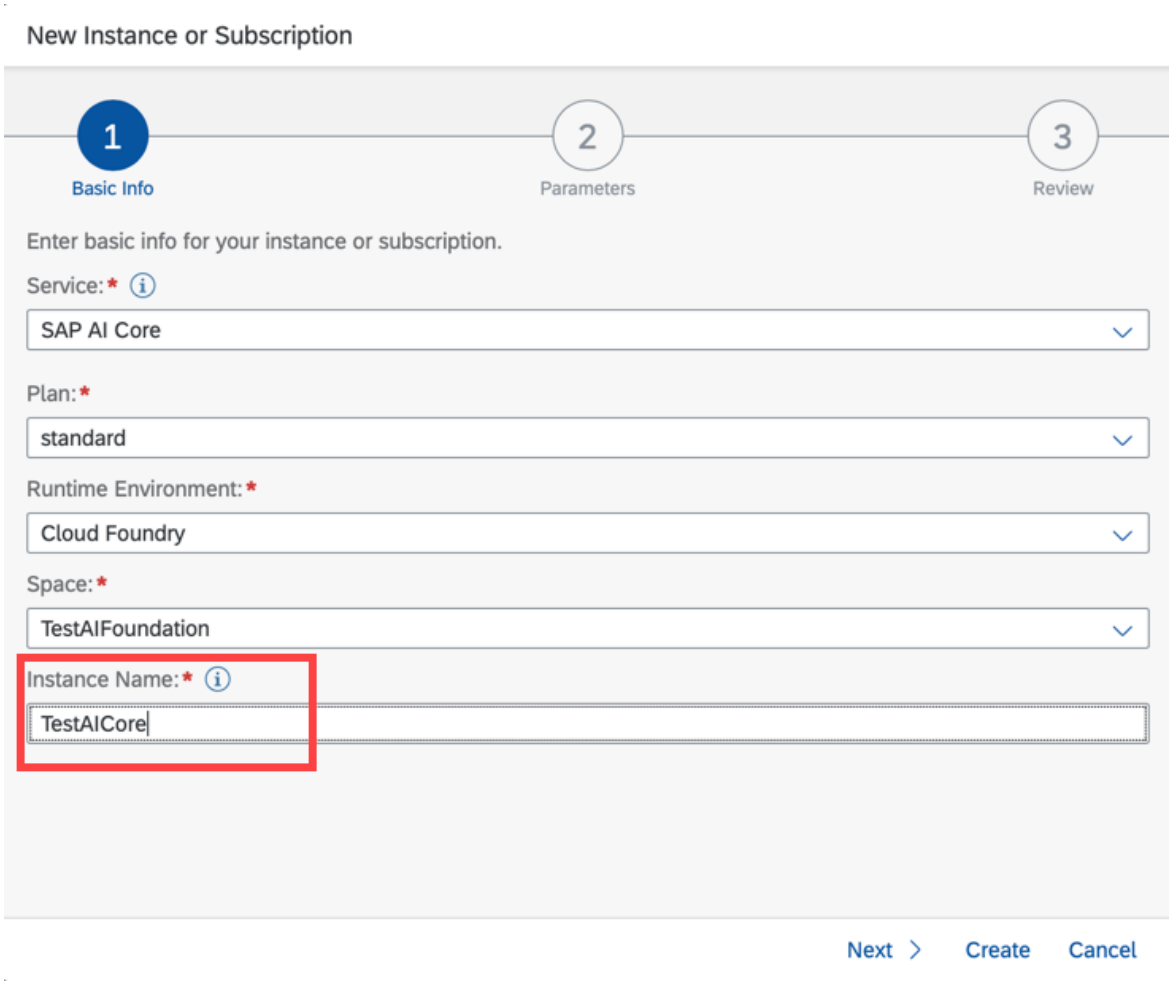
   You will see a tile for SAP AI Core.



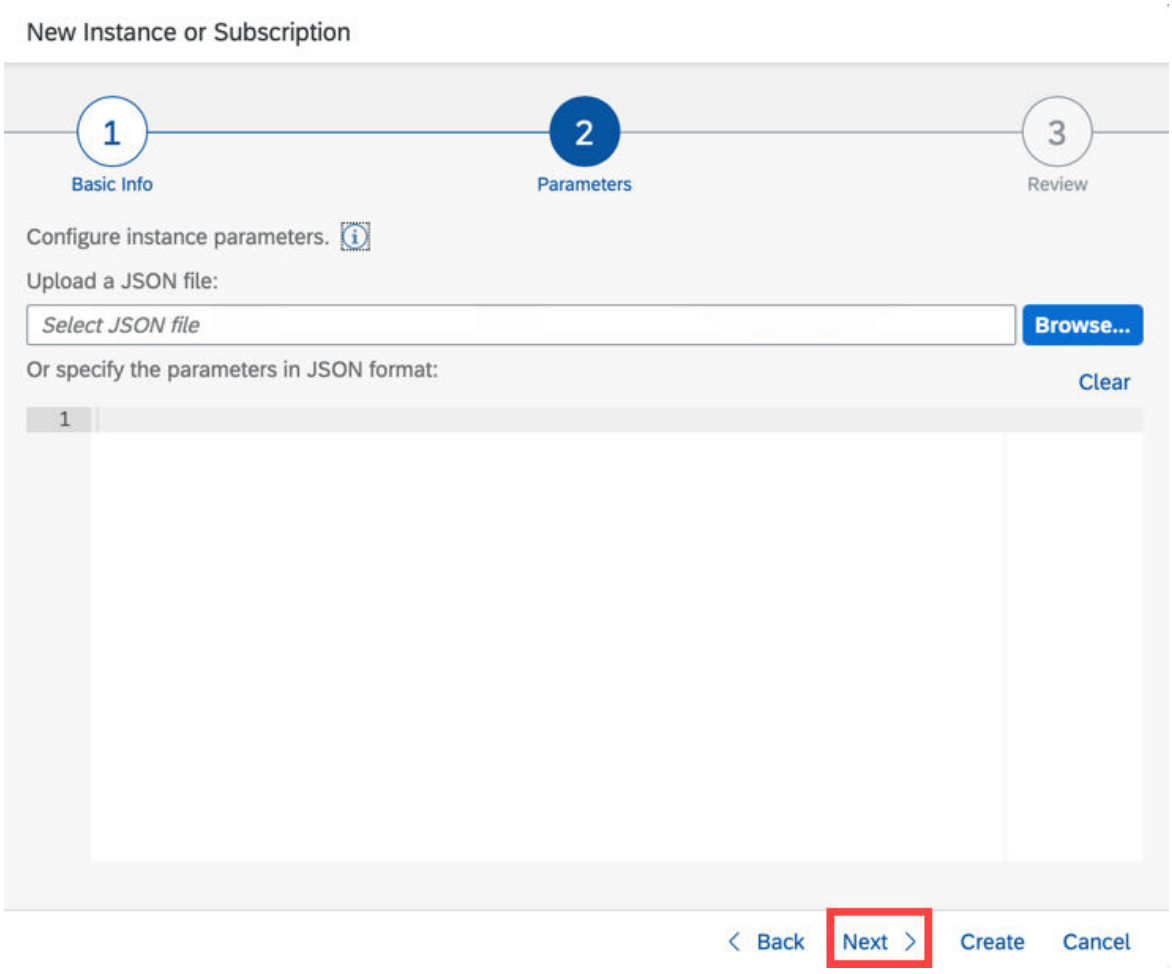

2. Open the tile and click *Create*.



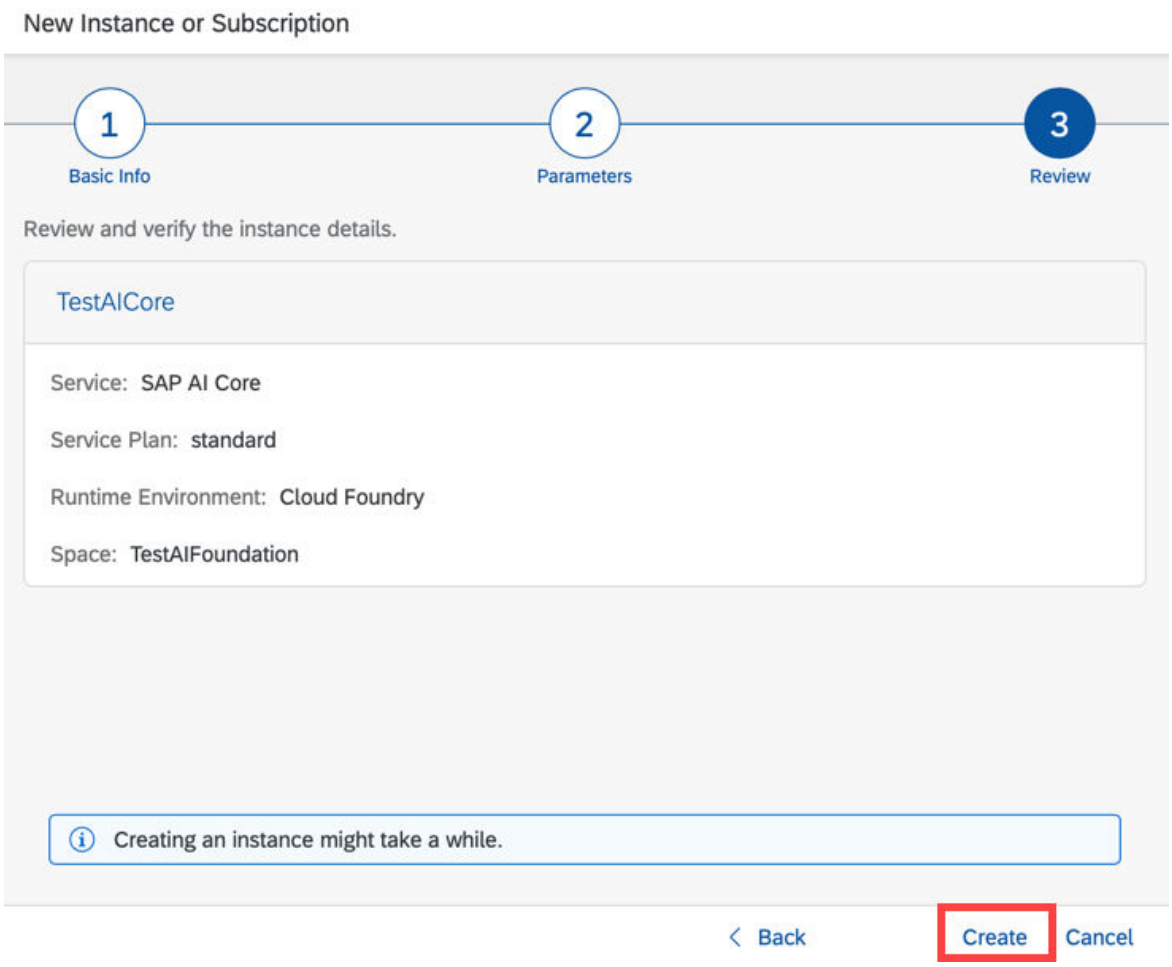

3. Enter a name for your instance and choose *Next* (all other details will be filled by default).




4. Parameters can be uploaded using a JSON file, this is optional. Choose *Next* to proceed.

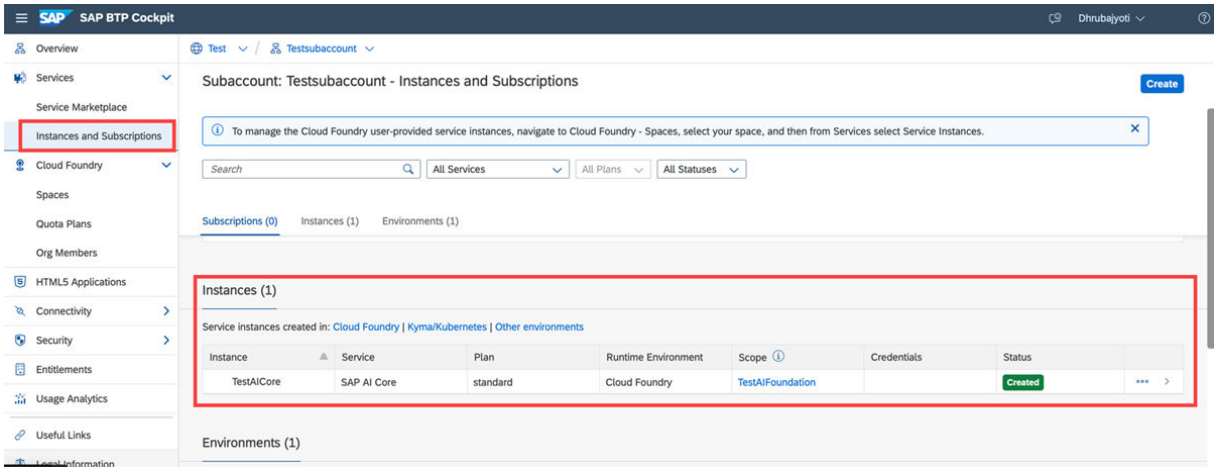



5. Review and verify the instance, and then choose *Create*.




## Results

When your service instance is created, you can view it on the *Instances and Subscriptions* page of your subaccount.



**Task overview:** Initial Setup [page 11]

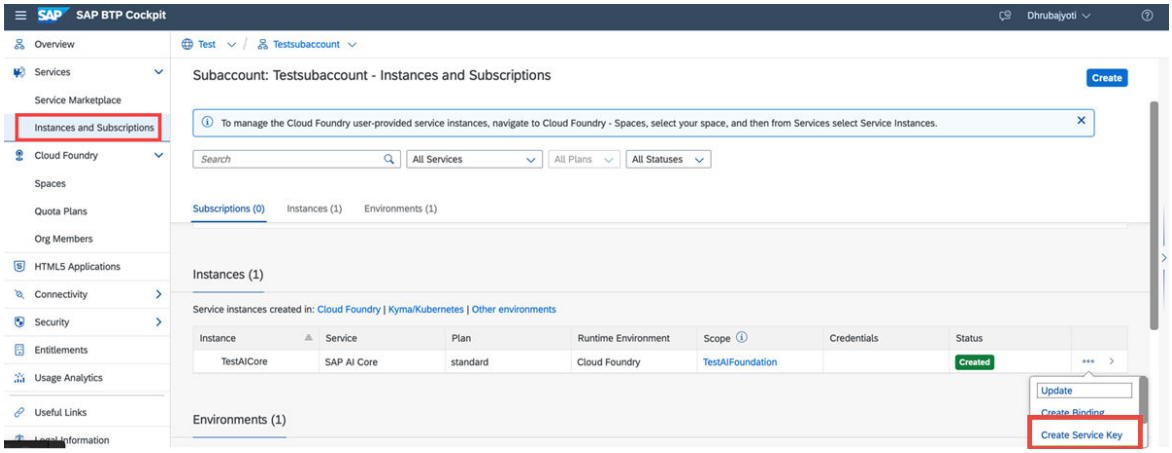
**Previous task:** Add a Service Plan [page 16]

**Next task:** Create a Service Key [page 22]

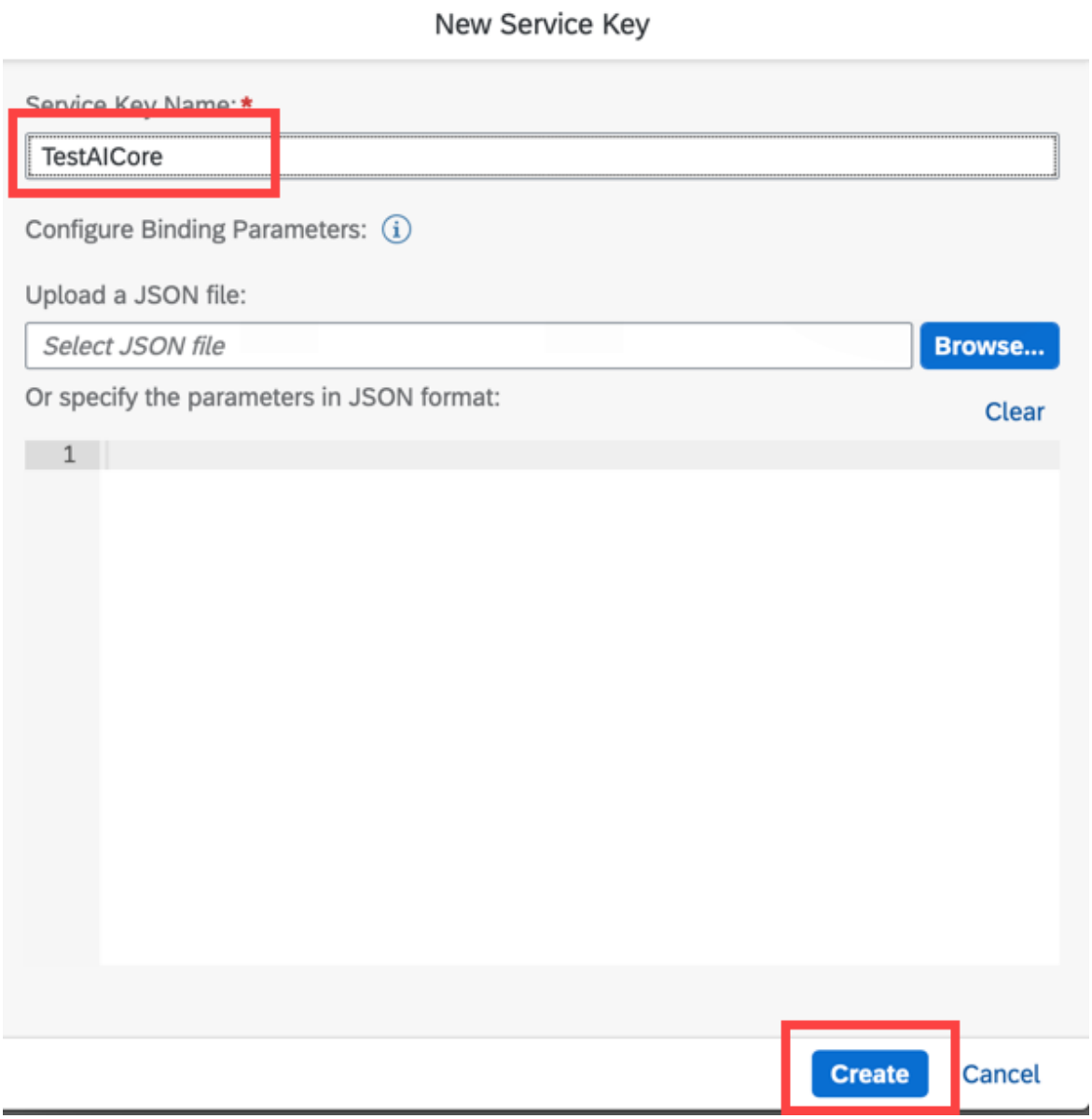
# 3.6 Create a Service Key

## Procedure

1. On the *Instances and Subscriptions* page, find your new instance and choose *Create Service Key* from the dropdown.




2. Enter a name for your service key and click *Create*.




3. Download your service key to save it.

## Results

You now have your service key, which provides URLs and credentials for accessing the SAP AI Core instance through SAP AI Launchpad, SAP AI Core toolkit Postman, or curl.

Credentials

aicore-servicekey

Form | JSON


```
1  {
2      "clientid": "sb-4376bdb7-                                    b313",
3      "clientsecret": "                                            =",
4      "url": "https://          j.authentication.sap.hana.ondemand.com",
5      "identityzone": "ai                  ",
6      "identityzoneid": "1eb727a0-                                 a",
7      "appname": "4                                                e!b313",
8      "serviceurls": {
9          "AI_API_URL": "https://api.ai.i                          s.ml.hana.ondemand.com"
10     }
11 }
```


Copy JSON  Download  Close

- `clientid`, `clientsecret`, and `url` can be used to generate your authentication token.
- `identityzone` and `identityzoneid` represent your tenant ID.
- `appname` provides the service instance details if service instance isolation is implemented.
- `serviceurls` allow you to interact with SAP AI Core once your authentication token has been generated.
  - `AI_API_URL`: Unified AI API to handle ML artifacts (such as training, data, models, and deployments) across multiple hyperscalers.

**Task overview:** Initial Setup [page 11]

**Previous task:** Create a Service Instance [page 18]

**Next task:** Use a Service Key [page 24]

## 3.7 Use a Service Key

**Task overview:** Initial Setup [page 11]

**Previous task:** Create a Service Key [page 22]

# Using SAP AI Launchpad

## Prerequisites

You have the `connections_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

## Procedure

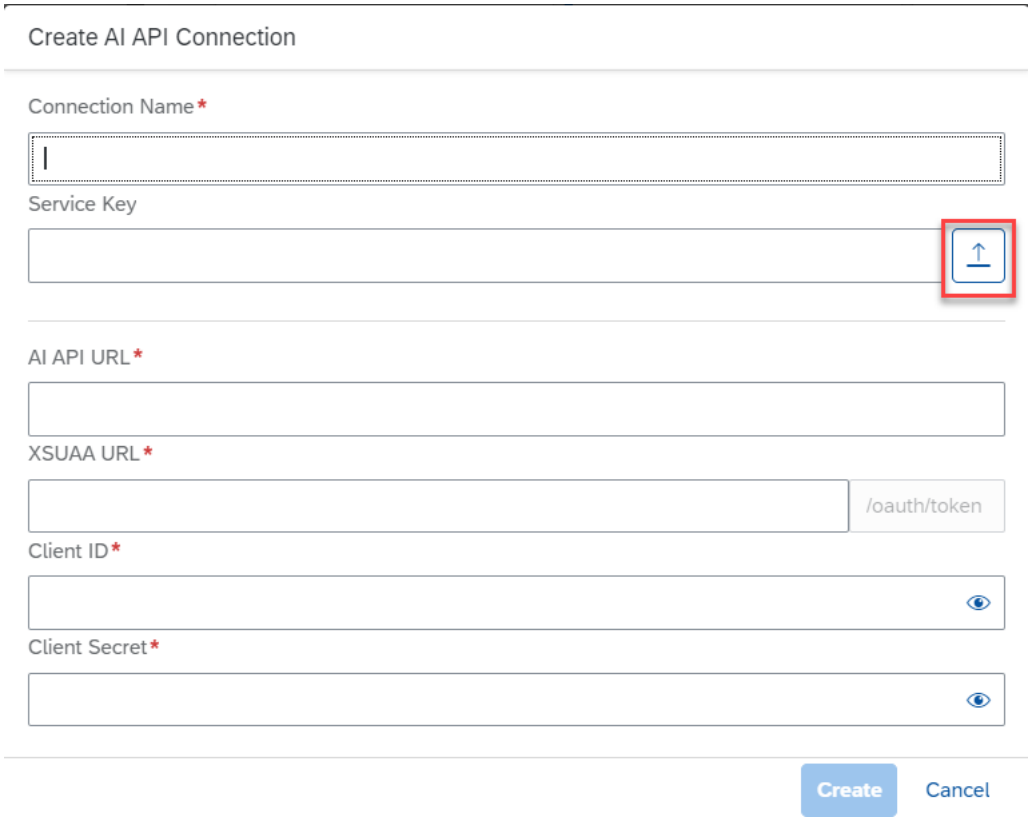
1. Open the *Workspaces* app and choose *Add*. The *Create AI API Connection* dialog appears.
2. Enter a name for your new connection.

   Connection names must comply with the following criteria:

   - Contain only lowercase alphanumeric characters, hyphens (-), or periods (.)
   - Start with an alphanumeric character
   - End with an alphanumeric character
3. Upload the service key file for your SAP AI Core instance, if available.

   To upload the service key as a `.TXT` or `JSON` file, choose ↑ *(Upload)* . Search for and choose the local service key file.

   Service key data then defaults to the remaining fields.



4. If a service key file is not available, complete the following:
   a. Enter the `AI_API_URL` from your service key for your SAP AI Core instance.
   b. Enter the `url` from your service key for your SAP AI Core instance.
   c. Enter the `clientid` from your service key for your SAP AI Core instance.
   d. Enter the `clientsecret` from your service key for your SAP AI Core instance.

   Sensitive fields are masked. You can unmask to show your entry if needed.
5. Choose *Create*.

### Results

The new connection appears in the *Workspaces* app.

## Using Postman

### Prerequisites

- You have downloaded and installed the Postman client from https://www.postman.com/ .
- You have familiarized yourself with the Postman documentation and interface.

### Procedure

1. Download the JSON collection from the https://api.sap.com/api/AI_CORE_API/overview.
2. In Postman, press *Import* and choose the JSON file download. Double-click, then choose *Import* to start the import.
3. After the import is complete, highlight the collection, then select the *Authorization* tab.
4. Navigate to *Configure New Token*, enter the credentials from your service key and choose save.

   > **i Note**
   >
   > - The *Token Name* field is your choice of descriptive identifier.
   > - The *Access Token URL* is labelled *url* in your service key. Add **`/oauth/token`** to the end of the url.
   > - The *Grant Type* should be **`Client Credentials`**.
   >
   > You might see an alert relating to the characters in your credentials, you can ignore this.
5. Select the *Variables* tab, and set your `baseUrl` from your credentials.

   The `baseUrl` is labelled *AI_API_URL* in your service key.
6. Choose *Save*.
7. In the *Authorization* tab, choose *Get New Access Token*. and wait for the authentication process.

When the authentication process is comeplete, select *Use Token* to finish. Check that the token is stored in your environment variables. If it is not stored automatically, you can copy and paste it to the *Token* field manually.

# Using Curl

## Prerequisites

curl is likely to be installed on your operating system by default. To check, open a command prompt and enter **`curl -V`**. If curl isn't installed, download and install it from https://curl.se/ .

> **Note**
>
> On macOS, you may need to install `jq` so that you can follow the curl commands.
>
> 1. Install brew from https://brew.sh/ .
> 2. In a Terminal session, run `brew install jq` to install `jq` in your shell environment.

## Procedure

1. Set up your environment as follows:

```
# XSUAA details
# URLs should be without trailing slash '/'
export CLIENTID=<clientid> from service key
export CLIENTSECRET=<clientsecret> from service key
export XSUAA_URL=<url> from service key
export AI_API_URL=<AI_API_URL> from service key
```

> **Note**
>
> The `export` command sets the values of your keys to your environment variable, meaning that they will be retained after you close your terminal session. It is possible to set the environment variables without the `export`, for the current session only.

2. Get the XSUAA OAuth Token using `clientid` and `clientsecret` from the service key to call the APIs.

   The XSUAA OAuth token is required for authentication when making AI API calls.

```
SECRET=`echo -n '$CLIENTID:$CLIENTSECRET' | base64 -i - `
TOKEN=`curl --location --request POST "$XSUAA_URL/oauth/token?
grant_type=client_credentials" \
 --header "Authorization: Basic $SECRET" | jq -r '.access_token'`
```

> **Note**
>
> The token is valid for a limited time. Once it has expired, create a new token, using the same code snippet.

3. Verify that the token has been fetched properly:

```
echo $TOKEN
```

You should see a long string of alphnumeric characters:


```
eyJhbGciOiJSUzI1NiIsImprdSI6Imh0dHBzOi8vYWktYWxwaGEtdmFsaWRhdGlvbi0yLmF1dGhlbn
RpY2F0aW9uLnNhcC5oYW5hLm9uZGVtYW5kLmNvbS90b2tlbl9rZXlzIiwia2lkIjoiZGVmYXVsdC1q
d3Qta2V5LTMyODMxMjg2NCIsInR5cCI6IkpXVCJ94ZGU5YjAxNmQ0MDk5YjlmM...
.............................................
...ALdfbMsHoYTtF6fNFbf3ZQ
```

# 4 Tutorials

The activation and consumption steps are also available online in a step by step tutorial.

For more information, see Prompt LLMs in the Generative AI Hub in SAP AI Core and SAP AI Launchpad.

# 5 Activating the Generative AI Hub

## 5.1 Activate the Generative AI Hub for SAP AI Core

You make a large language model (LLM) available for use by creating a virtual deployment. You can do so one time for each model and model version, and for each resource group that you want to use with Generative AI Hub. The virtual deployment URL that is generated can be reused.

### Prerequisites

- You have an SAP AI Core service instance and service key. For more information, see Initial Setup [page 11].
- You're using the `sap-internal` service plan. For more information, see Service Plans [page 7].
- You have completed the client authorization for your preferred user interface. For more information, see Use a Service Key [page 24]

### Context

You make an LLM available for use by creating a virtual deployment. You can do so one time for each model and model version. The LLM virtual deployment includes the `modelName` and `version` of the model you want to access. After the virtual deployment is complete, you have a `deploymentUrl`, which can be used across your organization to access the LLM model version.

## Using the API

### Procedure

1. Decide which LLM you want to deploy and note the following information:
   - Executable ID
   - Model name
   - Model version

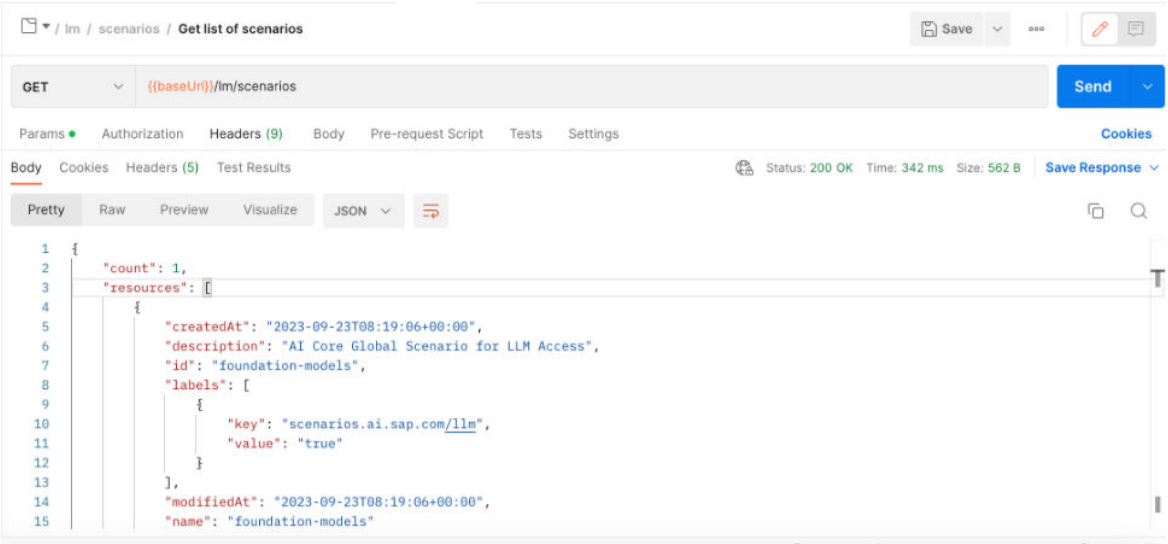
> **i Note**
>
> - Where model version is not listed, it is not applicable.
> - Instead of specifying a model version, using "latest" will use the latest version of the model available in SAP AI Core. For more information, see Model Lifecycle [page 42].

| Executable ID | Model Name | Model Version | Deprecation (as Specified by Model Provider) | Region | Request Limit (Requests per Minute) |
|---|---|---|---|---|---|
| azure-openai | gpt-35-turbo | 0613 | 2025-07-05 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 120 |
| azure-openai | gpt-35-turbo-16k | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 96 |
| azure-openai | gpt-4 | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 18 |
| azure-openai | gpt-4-32k | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 78 |
| azure-openai | text-embedding-ada-002 | 2 | 2025-02-02 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 138 |
| aicore-open-source | tiiuae--falcon-40b-instruct | | | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | TBD |

2. Check that you have access to the scenario containing LLMs by sending a GET request to `{{apiurl}}/v2/lm/scenarios`.

   Set the *Authorization* header with **`Bearer $TOKEN`** and set your resource group.

   > **i Note**
   >
   > You must use the same resource group for all of you LLM activities. To use a different resource group, these steps must be repeated for each resource group.




The scenarios listed contain a scenario with the id `foundation-models`.

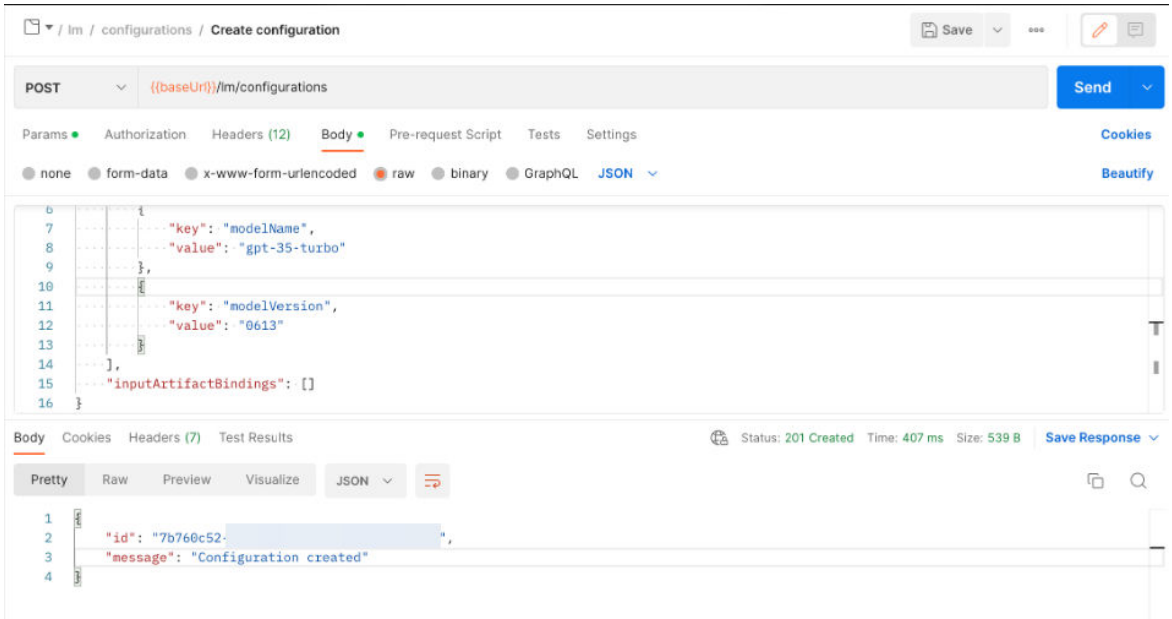
3. Create a configuration by sending a POST request to the endpoint `{{apiurl}}/v2/lm/configurations`.

   Include details of the model to which you want to provide access by passing in the following parameters:

   - `name` is your free choice of identifier.
   - `executableId`, `modelName`, and `modelVersion` are provided in the table above.
   - `scenarioId` must be `foundation-models`.
   - `versionId` is your own version reference.

Sample Code

```
{
    "name": "yourNameChoice",
    "executableId": "azure-openai",
    "scenarioId": "foundation-models",
    "versionId": "0.0.1",
    "parameterBindings": [
        {
            "key":"modelName",
            "value":"gpt-35-turbo"
        },
        {
            "key": "modelVersion",
            "value": "0613"
        }
    ],
    "inputArtifactBindings": []
}
```



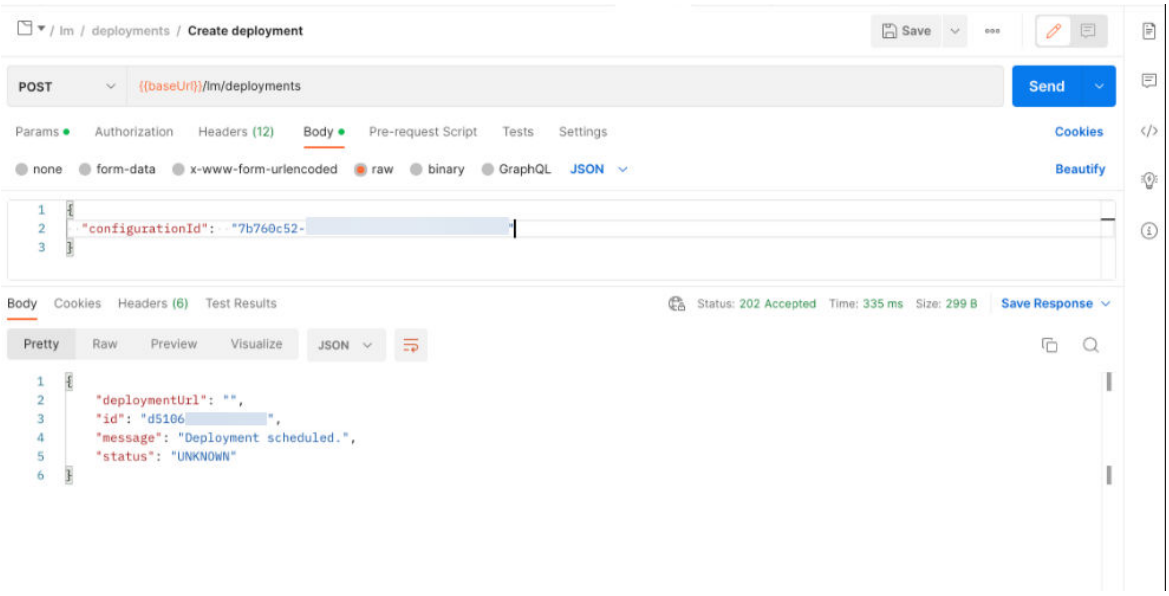

> **→ Tip**
>
> You can specify the value **`latest`** for the `modelVersion` to use the most recent model version available in SAP AI Core.

You receive a unique `configuationId` in the response.

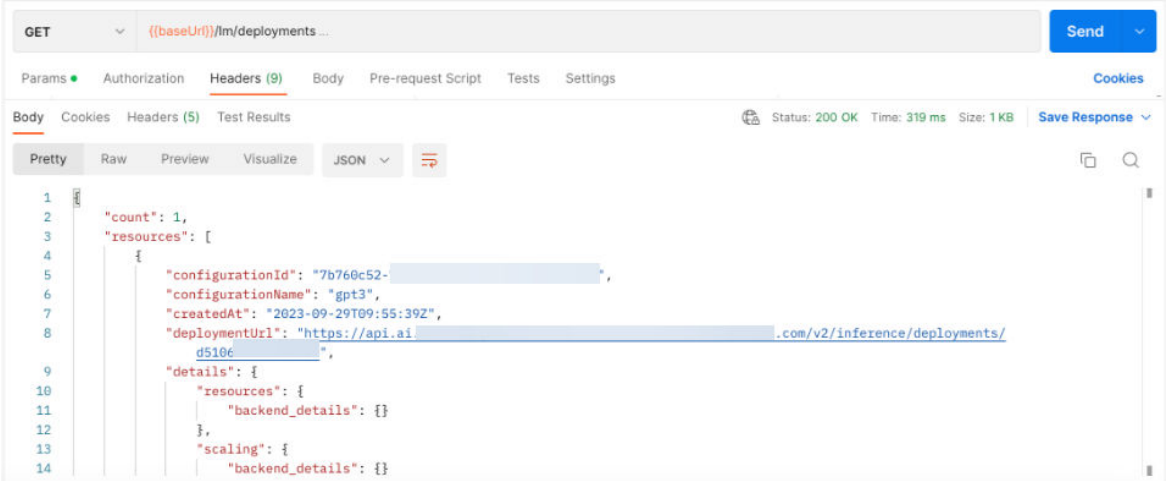
4. Create a virtual deployment by sending a POST request to the endpoint `{{apiurl}}/v2/lm/deployments`.

   Include the `configuationId` from the previous step in your request.

> **Sample Code**
>
> ```
> {
>  "configurationId": "yourConfigurationId"
> }
> ```




5. Retrieve the details of your virtual deployment by sending a GET request to the endpoint`{{apiurl}}/v2/lm/deployments`.




## Next Steps

When the virtual deployment is running, the LLM can be accessed using the `deploymentUrl` provided in the response. For more information, see Consume Large Language Models Using SAP AI Core [page 43].

## 5.2 Activate the Generative AI Hub for SAP AI Launchpad

You make a large language model (LLM) available for use by creating a virtual deployment. You can do so one time for each model and model version, and for each resource group that you want to use with Generative AI Hub.

### Prerequisites

- You have an SAP AI Launchpad service instance and service key. For more information, see Initial Setup [page 11].
- You're using the `sap-internal` service plan. For more information, see Service Plans [page 7].

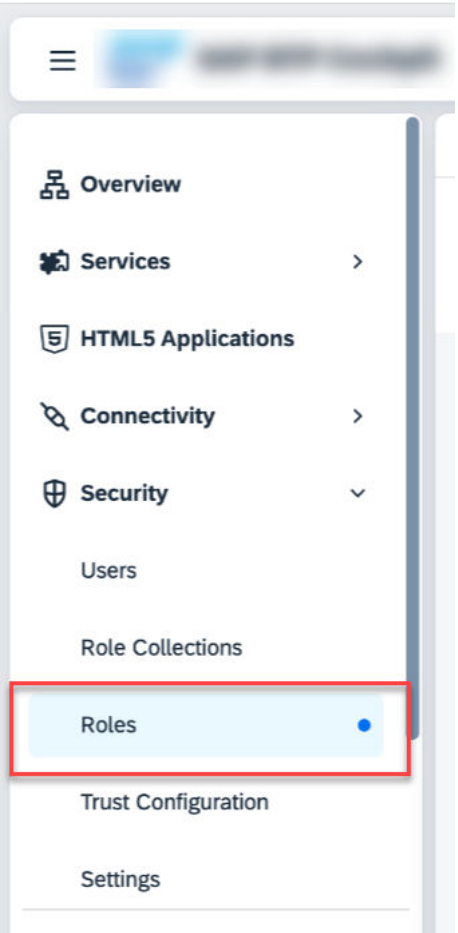
### Context

You access the Generative AI app within SAP AI Launchpad via the side navigation panel.

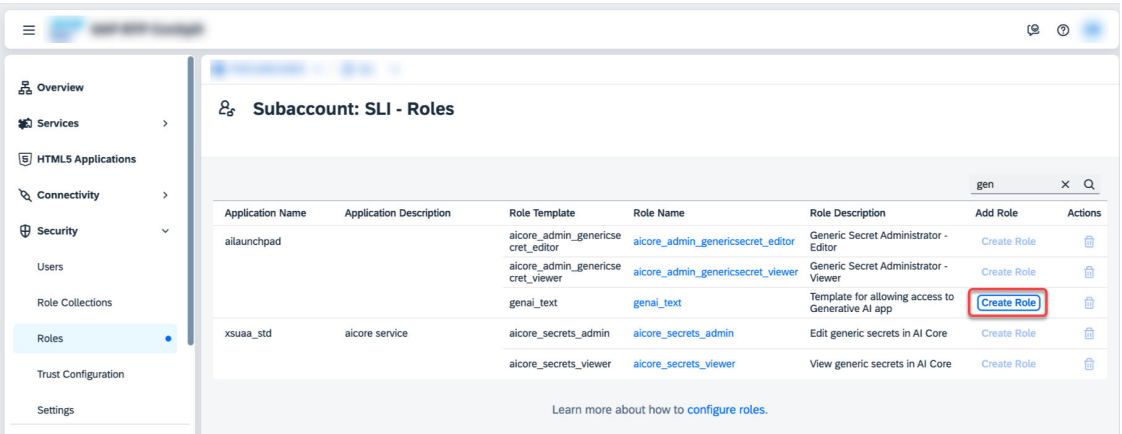
You make an LLM available for use by creating a virtual deployment. You can do so one time for each model and model version.

### Procedure

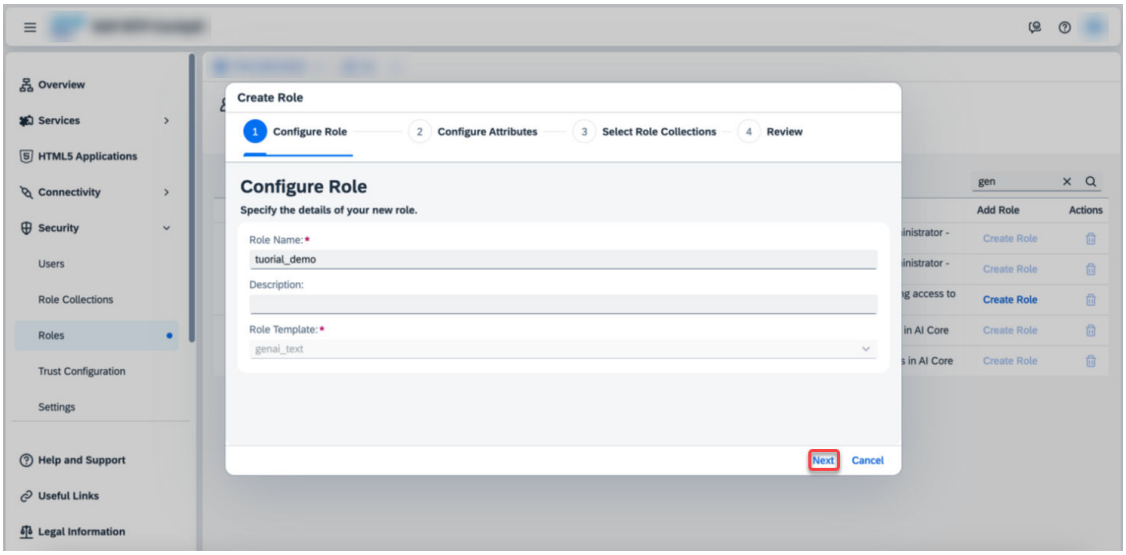
1. Assign the `genai_text` role template:
   a. In SAP BTP cockpit, navigate to your subaccount and choose *Roles* to access the list of roles.



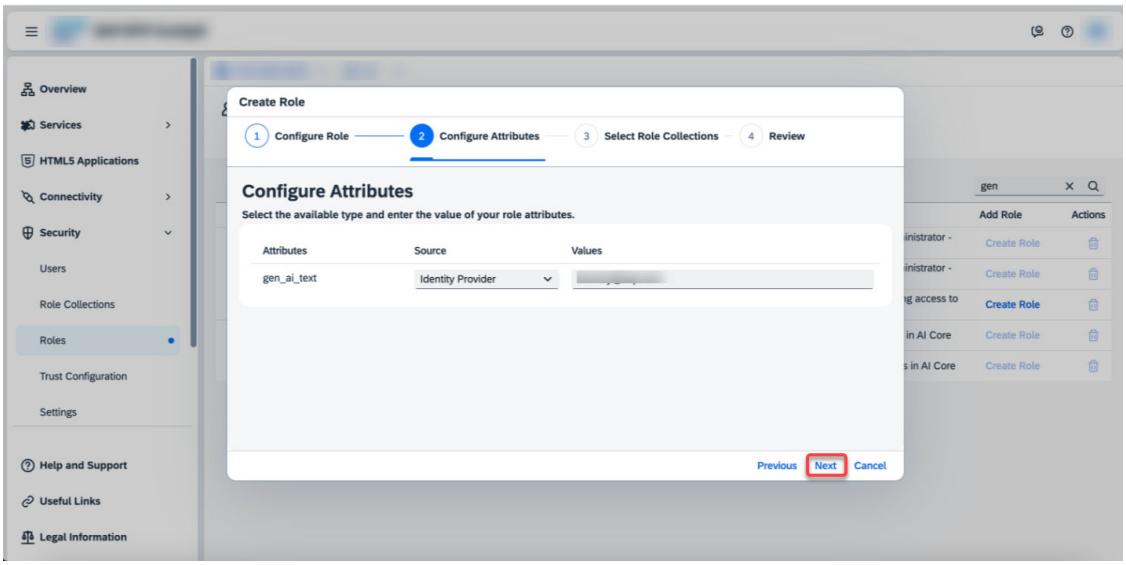

b. Within the `ailaunchpad` application, find the `genai_text` role template and choose *Create Role*.



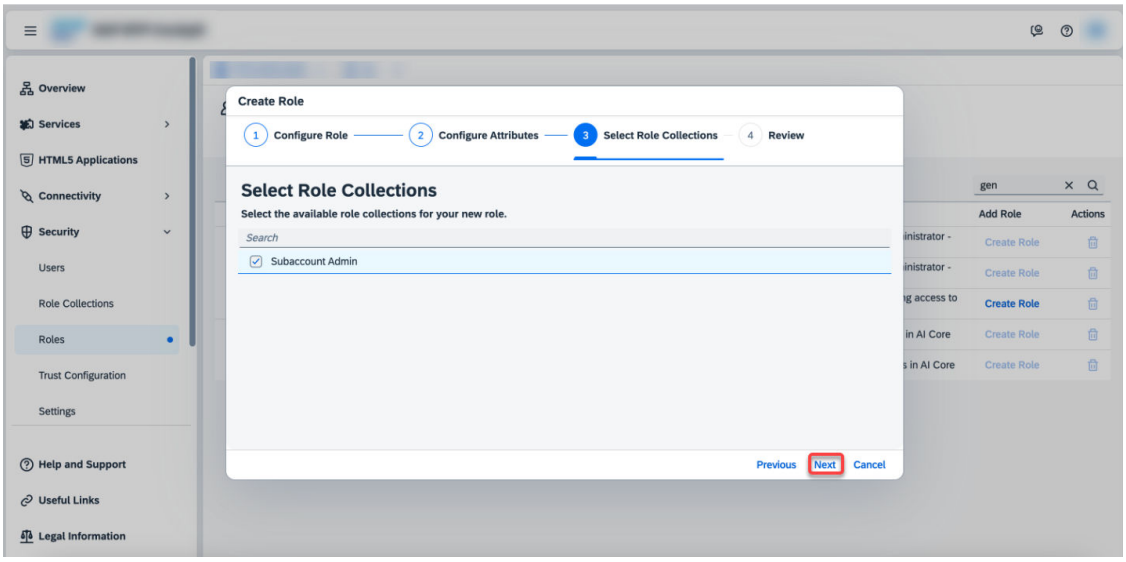

c. In the wizard, give the name of the role. Choose *Next*.



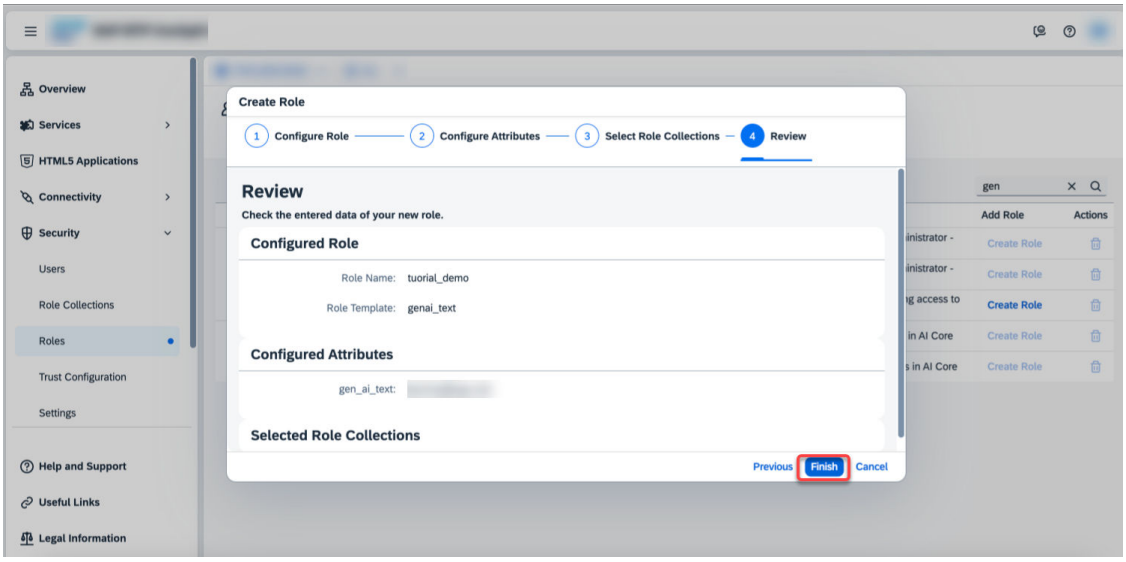

d. Select the available source type and enter the value of your role attribute.




e. Select a role collection and choose *Next*.

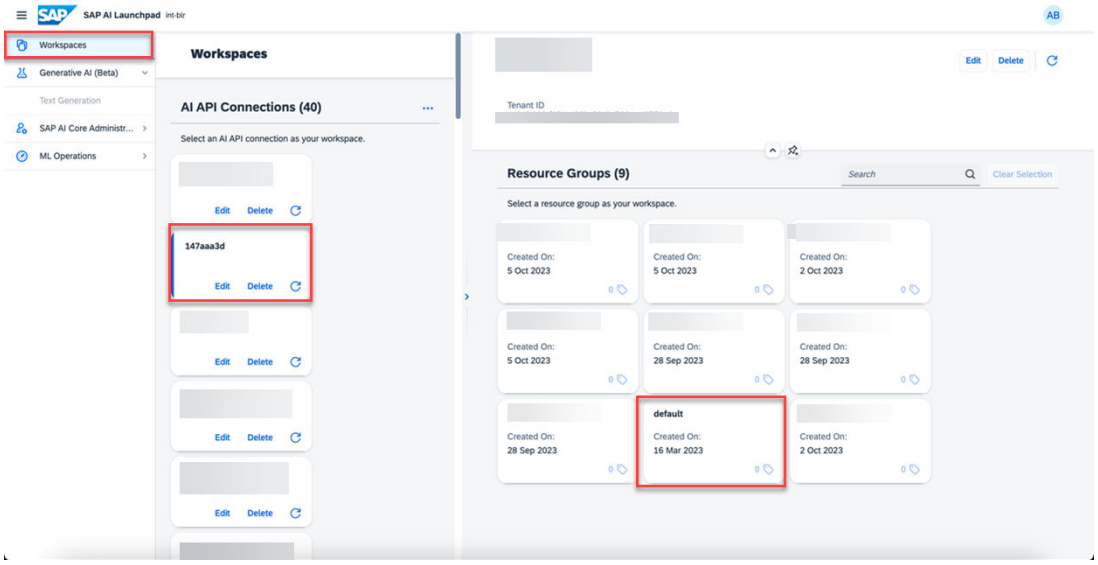



f. Review the data and choose *Finish*.

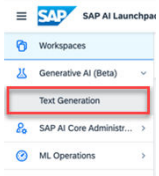



The role will be listed in your SAP BTP cockpit. For more information, see SAP AI Launchpad Role Collection Documentation

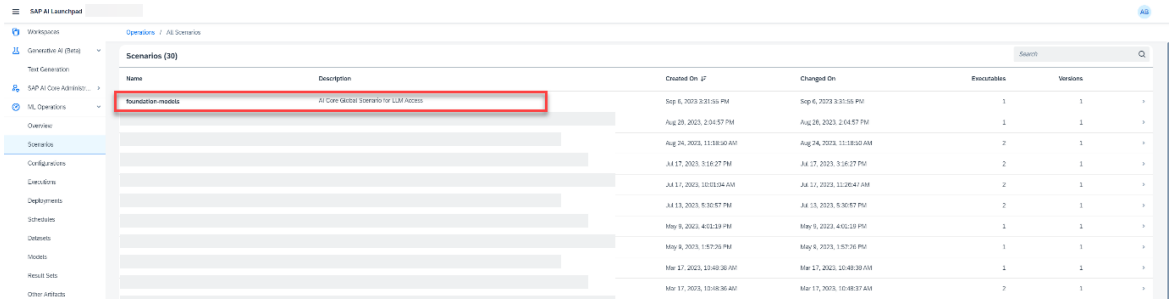
2. Select the connection to your SAP AI Core runtime in the *Workspaces* app and choose the *default* resource group.



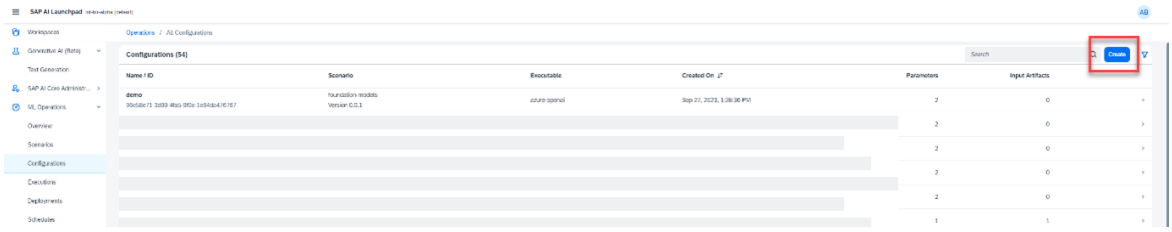

The *Generative AI* app is now clickable in your side navigation panel and resource groups are listed.



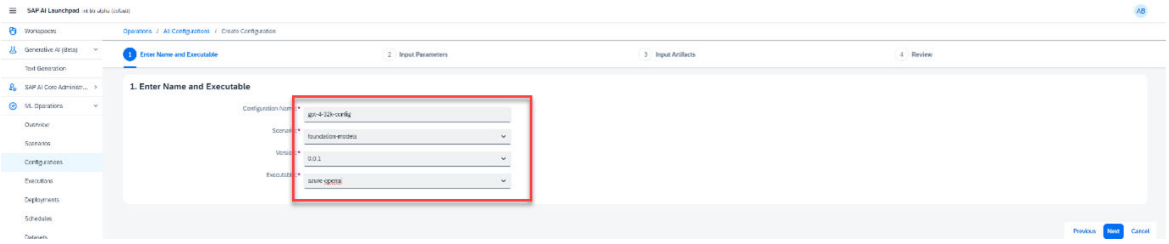

3. In the *ML Ops* app, choose *Scenarios* and check that a scenario called *foundation-models* scenario exists.




4. In the *ML Ops* app, choose *Configurations* and click *Create*.




5. Enter a name for your configuration, choose the *foundation-models* scenario, enter a version number, and select the executable for your chosen model provider.

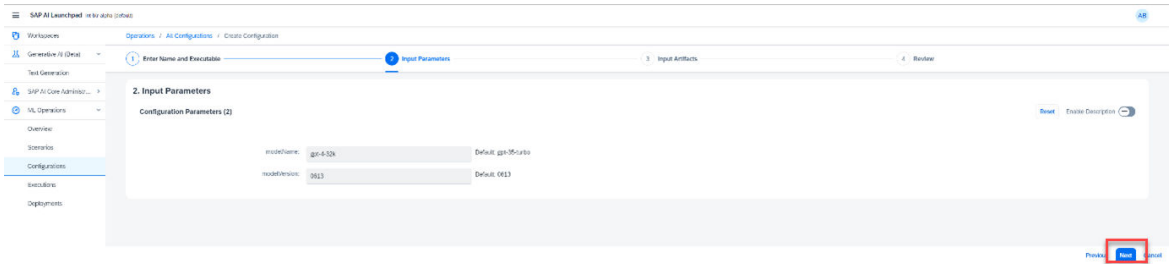



6. Enter the name and version (if applicable) of the model that you want to use.

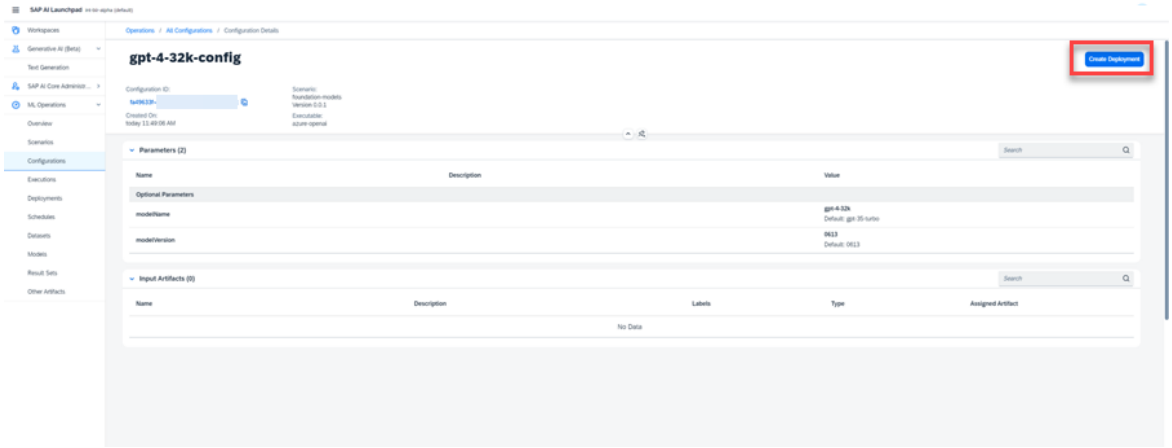
   The following models are available:

| Executable ID | Model Name | Model Version | Deprecation (as Specified by Model Provider) | Region | Request Limit (Requests per Minute) |
|---|---|---|---|---|---|
| azure-openai | gpt-35-turbo | 0613 | 2025-07-05 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 120 |
| azure-openai | gpt-35-turbo-16k | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 96 |
| azure-openai | gpt-4 | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 18 |
| azure-openai | gpt-4-32k | 0613 | 2024-15-01 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 78 |
| azure-openai | text-embedding-ada-002 | 2 | 2025-02-02 | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | 138 |
| aicore-open-source | tiiuae--falcon-40b-instruct | | | • US10 (mapped to Azure US East)<br>• EU10 (mapped to Azure EU Central) | TBD |

> **i Note**
>
> - Where model version is not listed, it is not applicable.
> - Instead of specifying a model version, using “latest” will use the latest version of the model available in SAP AI Core. For more information, see Model Lifecycle [page 42].



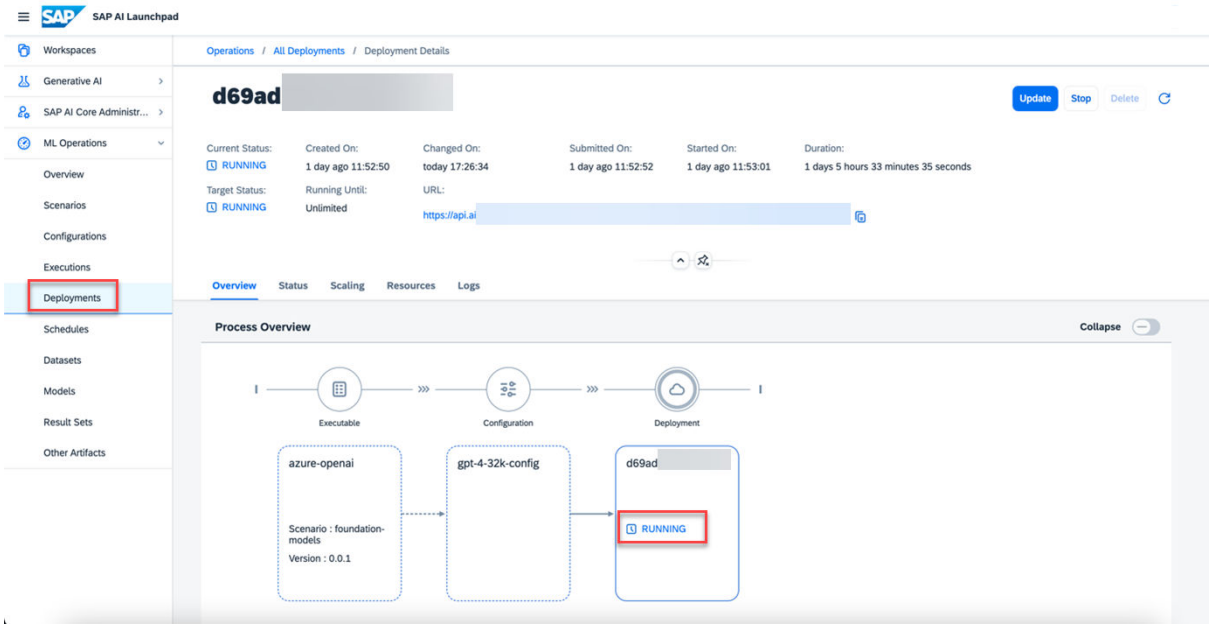

7. After you've created your configuration, select *Create Deployment*.




> **i Note**
>
> You must use the same resource group for all of you LLM activities. To use a different resource group, these steps must be repeated for each resource group.

## Next Steps

When the virtual deployment is running, the LLM can be accessed using the *Text Generation* app. For more information, see Consumption of LLMs Using SAP AI Launchpad [page 62].




## 5.3 Model Lifecycle

Model versions have deprecation dates. Where a model version is specified in a virtual deployment, the virtual deployment will stop working on the deprecation date of that model version.

Implement one of the following model upgrade options:

- **Auto Upgrade:** Create or patch an LLM configuration with `modelVersion latest`. When a new `modelVersion` is supported by SAP AI Core, existing LLM virtual deployments will automatically use the latest version of the given model.
- **Manual Upgrade:** Patch an LLM configuration with your chosen replacement `modelVersion`. This model version will be used in LLM virtual deployments irrespective of updates to the models supported by SAP AI Core.

> **i Note**
>
> If `modelVersion` isn't specified, it will be `latest` by default. To upgrade manually, you **must** specify a `modelVersion`.

# 6 Consuming Large Language Models

## 6.1 Consume Large Language Models Using SAP AI Core

You consume a large language model (LLM) by sending a request to the endpoint `{{deploymentUrl}}/chat/completions?api-version={{api-version}}`. The LLM can carry out natural language-related tasks such as answering questions, summarizing text, and extracting information from a body of text.

### Prerequisites

- You have the virtual deployment URL for your large language model (LLM). For more information, see Activate the Generative AI Hub for SAP AI Core [page 30].

### Context

The body of your request must include the `messages` parameter that defines your query.

Ensure that you have the following headers set:

| Header | Value |
|---|---|
| Authorization | Bearer $TOKEN |
| Resource group | The resource group used in the activation steps |

You can also include optional parameters such as:

| Models | Parameters |
| --- | --- |
| Azure | • `max_tokens`: An integer that defines the maximum number of tokens allowed for the generated answer. The default value is 4,096.<br>• `temperature`: A number between 0 and 2. Higher values make the output more random; lower values make it more focused and deterministic.<br>• `frequency_penalty`: A number between -2.0 and 2.0. Positive values penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim.<br>• `presence_penalty`: A number between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics.<br>• `stop`: A string or array. Up to four sequences. If the output generates one of the stop values, it will stop generating content.<br><br>For more information, see Azure Chat Completions Documentation . |
| Falcon | • `max_tokens`: An integer that defines the maximum number of tokens allowed for the generated answer. The default value is 4,096.<br>• `temperature`: A number between 0 and 2. Higher values make the output more random; lower values make it more focused and deterministic.<br>• `frequency_penalty`: A number between -2.0 and 2.0. Positive values penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim.<br>• `presence_penalty`: A number between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics.<br>• `stop`: A string or array. Up to four sequences. If the output generates one of the stop values, it will stop generating content.<br><br>For more information, see Tiiuae Falcon 40b Instruct Documentation . |

# Summarizing

You can provide the LLM with a text and ask for a summary of it.

## Procedure

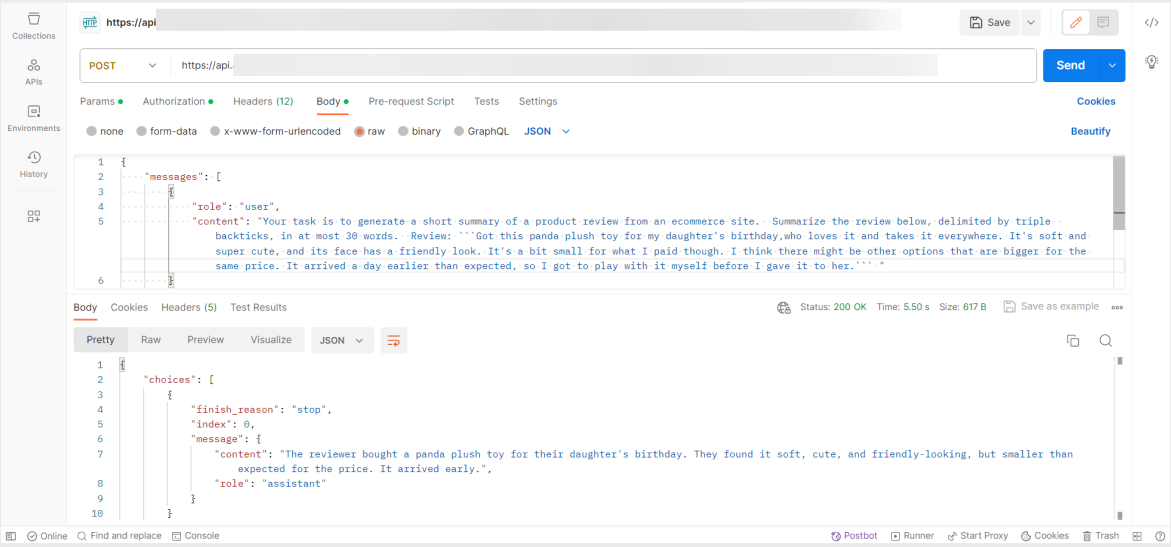
Send a POST request to the endpoint `{{deploymentUrl}}/chat/completions?api-version={{api-version}}`.

Include your query in the body. Mark the text to be summarized with triple back ticks (`).

## Example

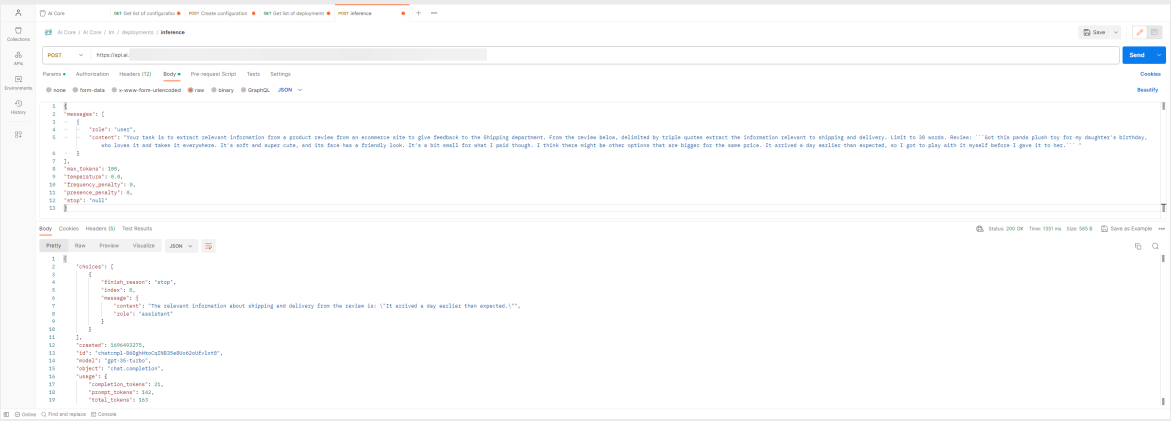
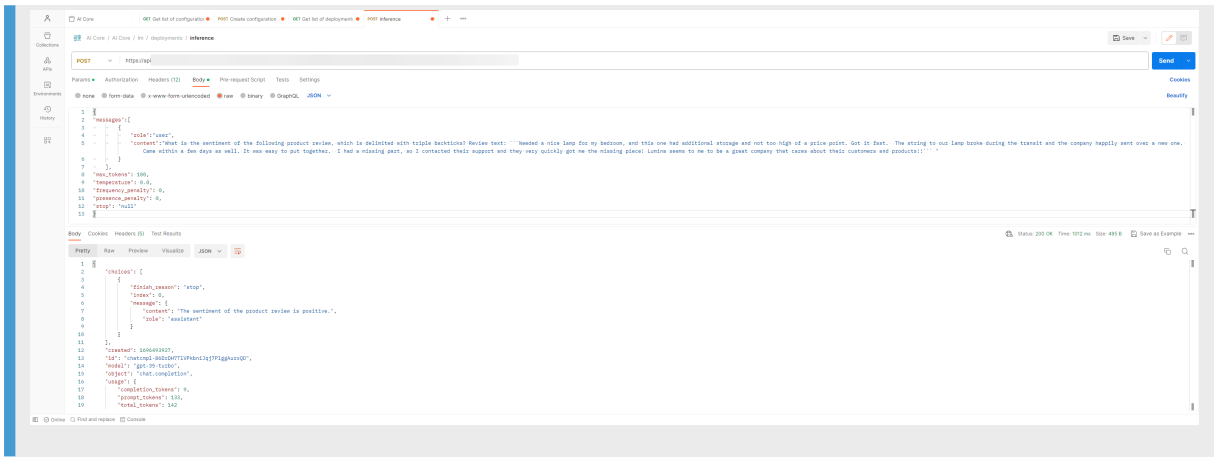
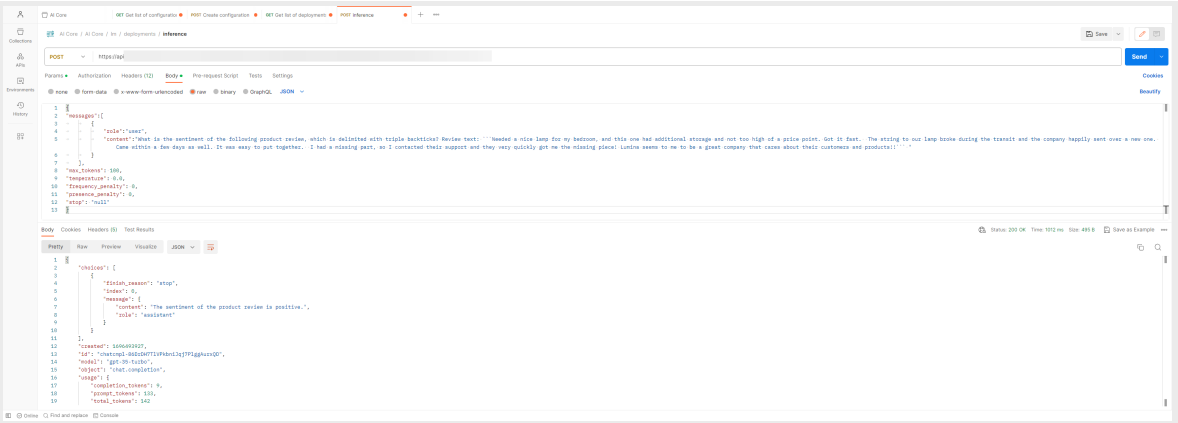
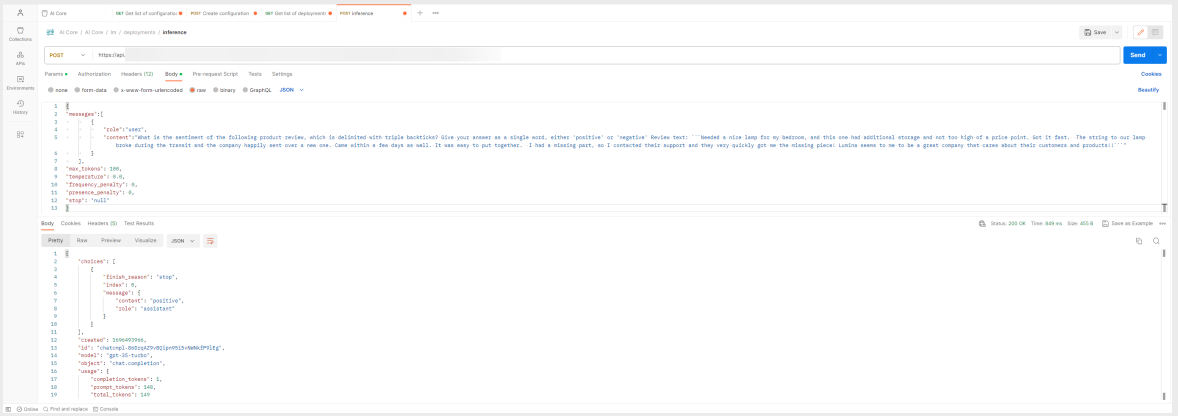
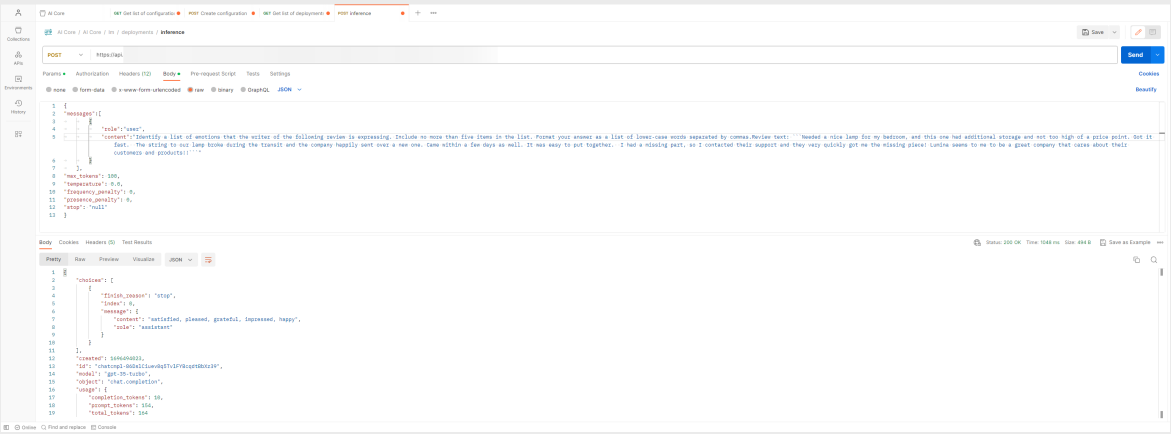
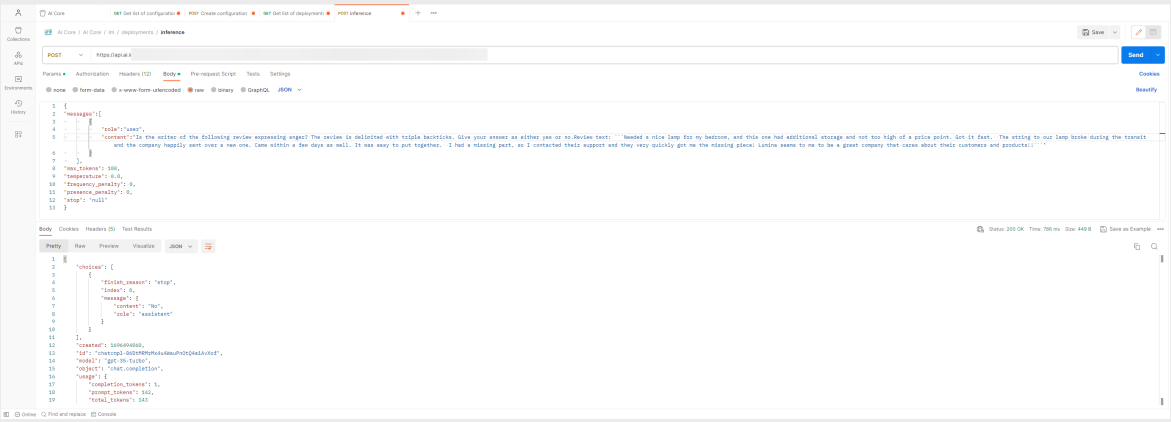
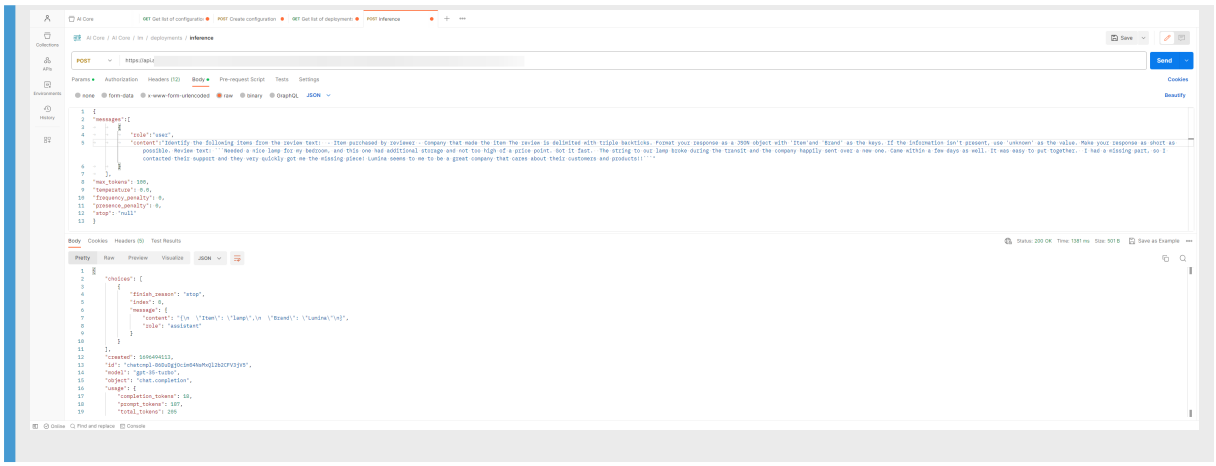
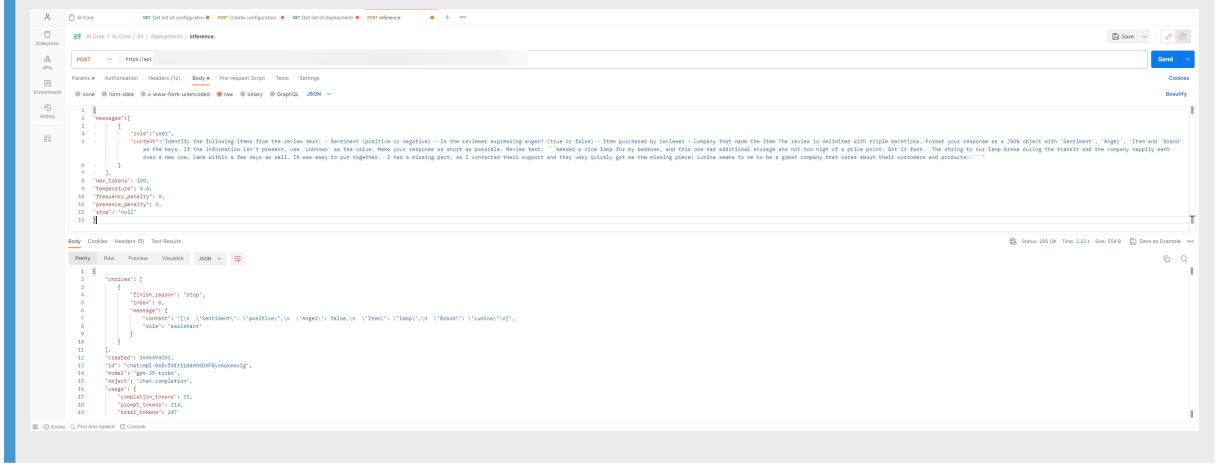
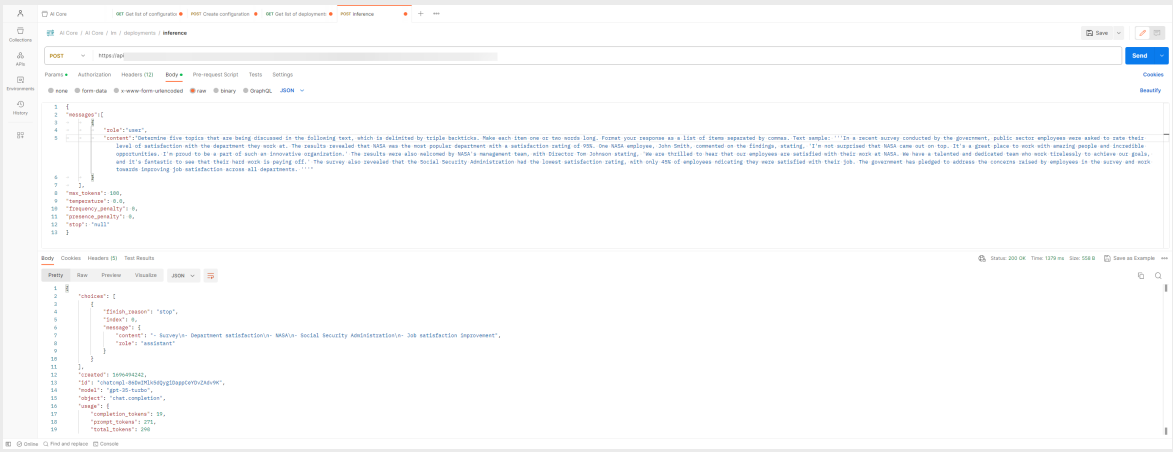
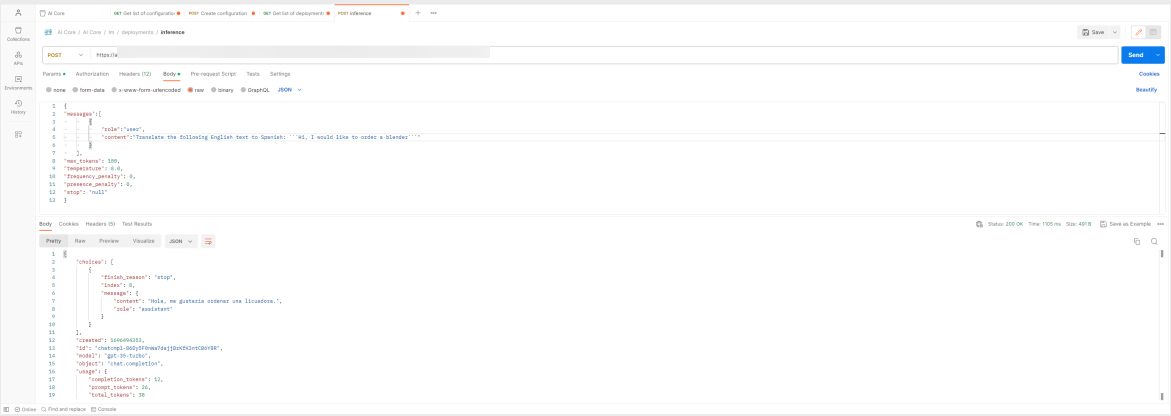
This example generates a summary of a product review. Summaries can include topics that aren't related to the main topic.

```
{
"messages": [
    {
        "role": "user",
        "content": "Your task is to generate a short summary of a product
review from an ecommerce site. Summarize the review below, delimited by
triple backticks, in at most 30 words. Review:```Got this panda plush toy for
my daughter's birthday, who loves it and takes it everywhere. It's soft and
super cute, and its face has a friendly look. It's a bit small for what I
paid though. I think there might be other options that are bigger for the
same price. It arrived a day earlier than expected, so I got to play with it
myself before I gave it to her. ```"
    }
],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```

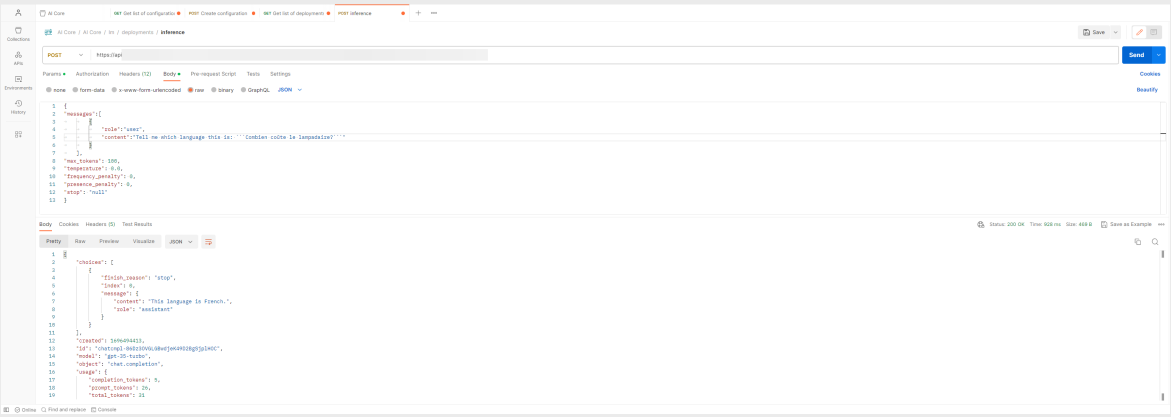



## Example

In this example, the prompt is similar but has been refined by adding the intended recipient of the feedback (the purchasing department) and the reason for requesting it (to determine the price of the product).

```
{
"messages": [
    {
        "role": "user",
        "content": "Your task is to generate a short summary of a product
review from an ecommerce site to give feedback to the pricing department,
responsible for determining the price of the product. Summarize the review
below, delimited by triple backticks, in at most 30 words, and focusing
on any aspects that are relevant to the price and perceived value. Review:
```Got this panda plush toy for my daughter's birthday,who loves it and takes
it everywhere. It's soft and super cute, and its face has a friendly look.
```
```

```
It's a bit small for what I paid though. I think there might be other options
that are bigger for the same price. It arrived a day earlier than expected,
so I got to play with it myself before I gave it to her.``` "
    }
],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example uses “extract” instead of “summarize”.

```
{
"messages": [
    {
        "role": "user",
        "content": "Your task is to extract relevant information from a
product review from an ecommerce site to give feedback to the Shipping
department. From the review below, delimited by triple backticks extract the
information relevant to shipping and delivery. Limit to 30 words. Review:
```Got this panda plush toy for my daughter's birthday,who loves it and takes
it everywhere. It's soft and super cute, and its face has a friendly look.
It's a bit small for what I paid though. I think there might be other options
that are bigger for the same price. It arrived a day earlier than expected,
so I got to play with it myself before I gave it to her.``` "
    }
],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```
```



# Inferencing

Inferencing uses the information in a given text to draw a conclusion.

## Procedure

Send a POST request to the endpoint `{{deploymentUrl}}/chat/completions?api-version={{api-version}}`.

Include your query in the body. Mark the text to be inferenced with triple back ticks (`).

### Example

This example performs a sentiment analysis on a product review.

```
{
"messages":[
        {
            "role":"user",
            "content":"What is the sentiment of the following product review,
which is delimited with triple backticks? Review text: ```Needed a nice lamp
for my bedroom, and this one had additional storage and not too high of a
price point. Got it fast.  The string to our lamp broke during the transit
and the company happily sent over a new one. Came within a few days as well.
It was easy to put together.  I had a missing part, so I contacted their
support and they very quickly got me the missing piece! Lumina seems to me to
be a great company that cares about their customers and products!!``` "
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example generates the sentiment as a one word response.

```
{
"messages":[
        {
            "role":"user",
            "content": "What is the sentiment of the following product
review, which is delimited with triple backticks? Give your answer as a
single word, either 'positive' or 'negative' Review text: ```Needed a nice
lamp for my bedroom, and this one had additional storage and not too high of
a price point. Got it fast.  The string to our lamp broke during the transit
and the company happily sent over a new one. Came within a few days as well.
It was easy to put together.  I had a missing part, so I contacted their
support and they very quickly got me the missing piece! Lumina seems to me to
be a great company that cares about their customers and products!!```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example analyzes the emotions expressed in the review.

```
{
"messages":[
        {
            "role":"user",
            "content": "Identify a list of emotions that the writer of the
following review is expressing. Include no more than five items in the list.
Format your answer as a list of lower-case words separated by commas.Review
text: ```Needed a nice lamp for my bedroom, and this one had additional
storage and not too high of a price point. Got it fast.  The string to
our lamp broke during the transit and the company happily sent over a new
one. Came within a few days as well. It was easy to put together.  I had
a missing part, so I contacted their support and they very quickly got me
the missing piece! Lumina seems to me to be a great company that cares about
their customers and products!!```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example detects whether anger is present.

```
{
"messages":[
        {
            "role":"user",
            "content": "Is the writer of the following review expressing
anger? The review is delimited with triple backticks. Give your answer as
either yes or no.Review text: ```Needed a nice lamp for my bedroom, and this
one had additional storage and not too high of a price point. Got it fast.
The string to our lamp broke during the transit and the company happily sent
over a new one. Came within a few days as well. It was easy to put together.
I had a missing part, so I contacted their support and they very quickly got
me the missing piece! Lumina seems to me to be a great company that cares
about their customers and products!!```"
        }
    ],
"max_tokens": 100,
```

```
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example detects product and company names from the customer review.

```
{
"messages":[
        {
            "role":"user",
            "content": "Identify the following items from the review text:
- Item purchased by reviewer - Company that made the item The review is
delimited with triple backticks. Format your response as a JSON object
with 'Item'and 'Brand' as the keys. If the information isn't present, use
'unknown' as the value. Make your response as short as possible. Review text:
```Needed a nice lamp for my bedroom, and this one had additional storage and
not too high of a price point. Got it fast.  The string to our lamp broke
during the transit and the company happily sent over a new one. Came within
a few days as well. It was easy to put together.  I had a missing part, so
I contacted their support and they very quickly got me the missing piece!
Lumina seems to me to be a great company that cares about their customers and
products!!```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```
```



## Example

This example performs multiple tasks in a single query.

```
{
"messages":[
        {
            "role":"user",
            "content": "Identify the following items from the review text: -
Sentiment (positive or negative) - Is the reviewer expressing anger? (true or
false) - Item purchased by reviewer - Company that made the item The review
is delimited with triple backticks. Format your response as a JSON object
with 'Sentiment', 'Anger', 'Item'and 'Brand' as the keys. If the information
isn't present, use 'unknown' as the value. Make your response as short as
possible. Review text: ```Needed a nice lamp for my bedroom, and this one
had additional storage and not too high of a price point. Got it fast.  The
string to our lamp broke during the transit and the company happily sent over
a new one. Came within a few days as well. It was easy to put together.  I
had a missing part, so I contacted their support and they very quickly got me
the missing piece! Lumina seems to me to be a great company that cares about
their customers and products!!```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example identifies the five topics discussed in a story.

```
{
"messages":[
        {
            "role":"user",
            "content": "Determine five topics that are being discussed in the
following text, which is delimited by triple backticks. Make each item one or
two words long. Format your response as a list of items separated by commas.
Text sample: '''In a recent survey conducted by the government, public sector
employees were asked to rate their level of satisfaction with the department
they work at. The results revealed that NASA was the most popular department
with a satisfaction rating of 95%. One NASA employee, John Smith, commented
on the findings, stating, 'I'm not surprised that NASA came out on top. It's
a great place to work with amazing people and incredible opportunities. I'm
proud to be a part of such an innovative organization.' The results were
also welcomed by NASA's management team, with Director Tom Johnson stating,
'We are thrilled to hear that our employees are satisfied with their work at
NASA. We have a talented and dedicated team who work tirelessly to achieve
our goals, and it's fantastic to see that their hard work is paying off.' The
survey also revealed that the Social Security Administration had the lowest
satisfaction rating, with only 45% of employees ndicating they were satisfied
with their job. The government has pledged to address the concerns raised by
employees in the survey and work towards improving job satisfaction across
all departments. '''"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



# Transformations

Transformations transform a given text into another language or register.

## Procedure

Send a POST request to the endpoint `{{deploymentUrl}}/chat/completions?api-version={{api-version}}`.

Include your query in the body. Mark the text to be transformed with triple back ticks (`).

### Example

This example translates text from English to Spanish.

```
{
"messages":[
        {
            "role":"user",
            "content": "Translate the following English text to Spanish:
```Hi, I would like to order a blender```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



### Example

This example detects the language that the text is written in.

```
{
"messages":[
        {
            "role":"user",
            "content": "Tell me which language this is: ```Combien coûte le
lampadaire?```"
        }
```
```

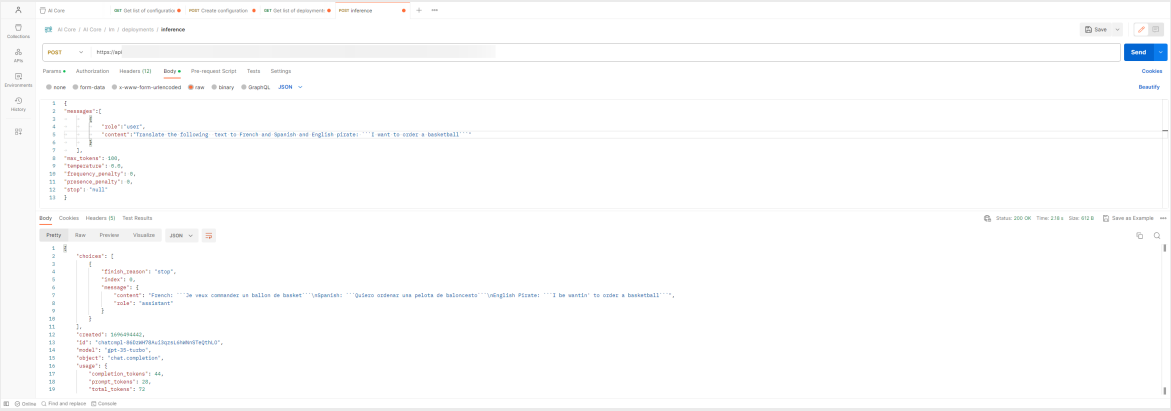
```
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example translates the given text into multiple languages.

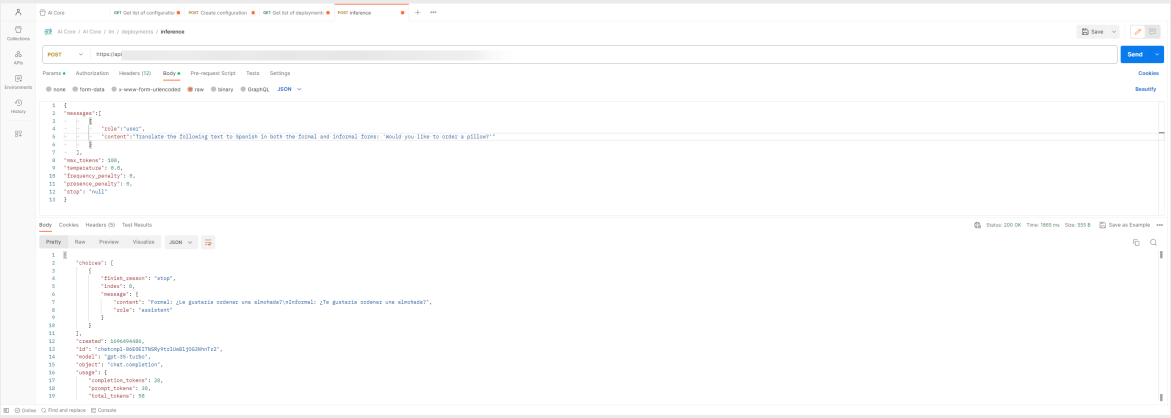
```
{
"messages":[
        {
            "role":"user",
            "content": "Translate the following  text to French and Spanish
and English pirate: ```I want to order a basketball```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example translates both the language and register of the text.

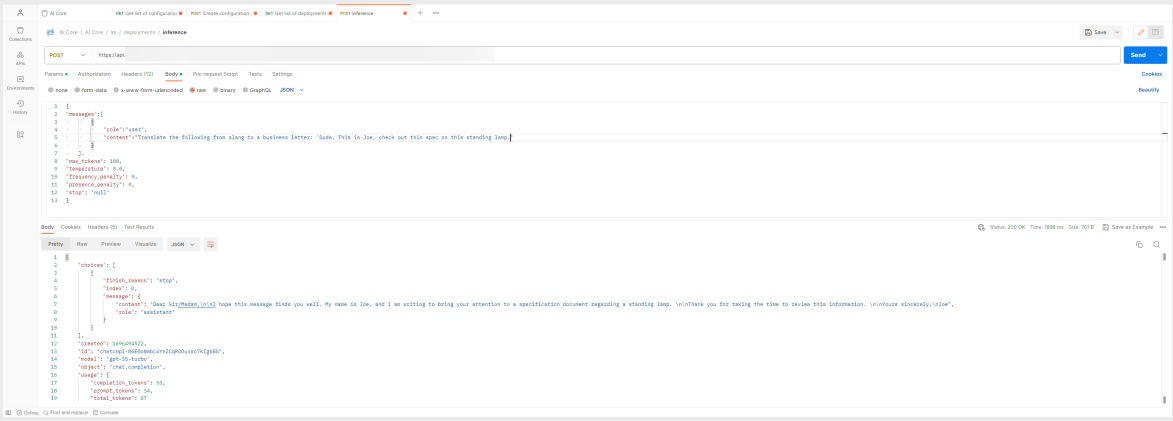
```
{
"messages":[
        {
            "role":"user",
            "content": "Translate the following text to Spanish in both the
formal and informal forms: 'Would you like to order a pillow?'"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example transforms the text to a formal register.

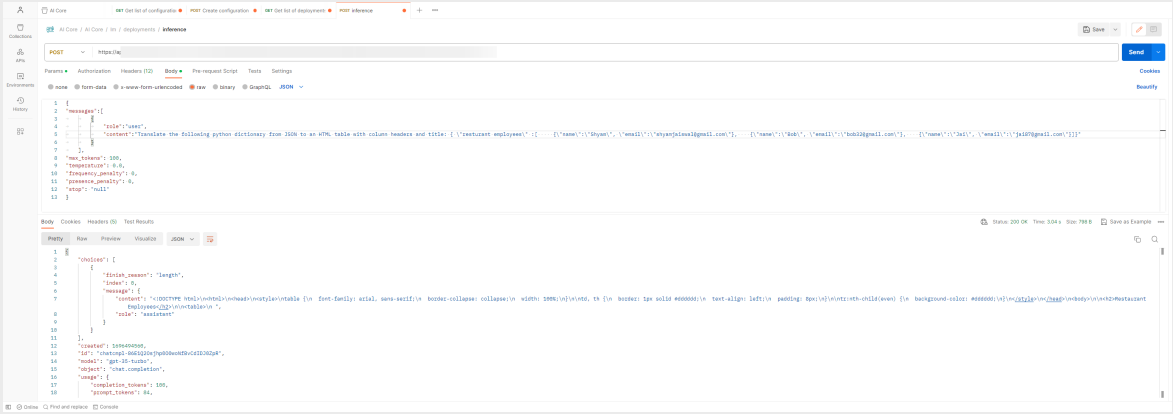
```
{
"messages":[
        {
            "role":"user",
            "content": "Translate the following from slang to a business
letter: 'Dude, This is Joe, check out this spec on this standing lamp.'"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## Example

This example translates between output formats. The prompt describes both the input and output format.

```
{
"messages":[
        {
            "role":"user",
            "content": "Translate the following python dictionary from JSON
to an HTML table with column headers and title: { \"resturant employees\" :
[     {\"name\":\"Shyam\", \"email\":\"shyamjaiswal@gmail.com\"},
{\"name\":\"Bob\", \"email\":\"bob32@gmail.com\"},    {\"name\":\"Jai\",
\"email\":\"jai87@gmail.com\"}]}"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```

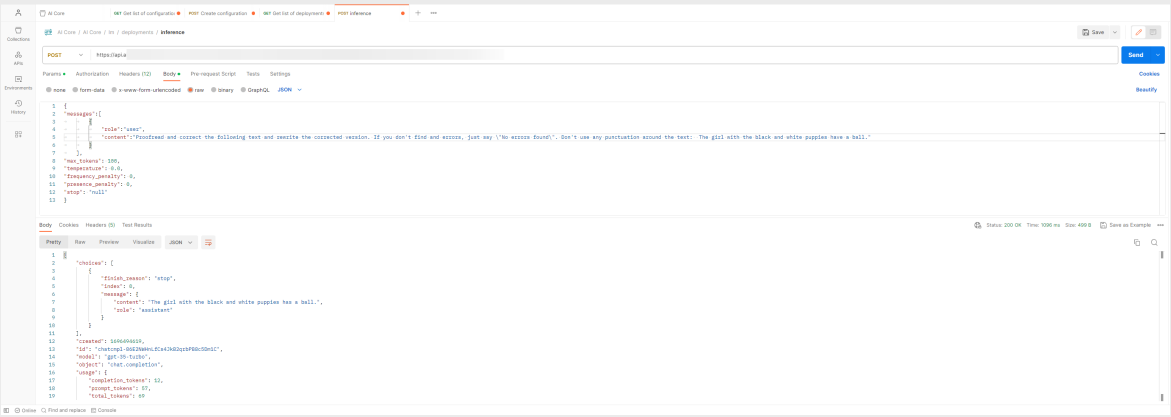


## Example

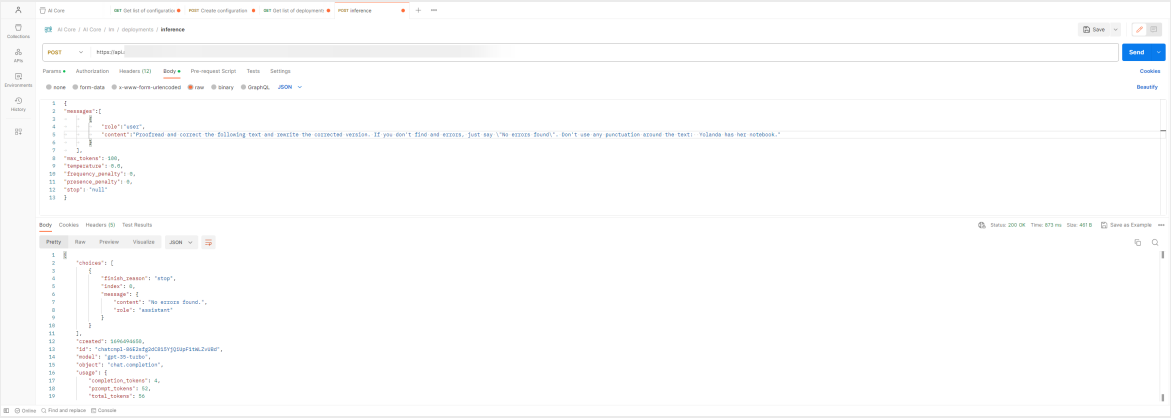
In these examples, a text is proofread. The text can be proofread and corrected, or simply proofread.

```
{
"messages":[
        {
```

```
            "role":"user",
            "content": "Proofread and correct the following text and rewrite
the corrected version. If you don't find and errors, just say \"No errors
found\". Don't use any punctuation around the text:  The girl with the black
and white puppies have a ball."
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```

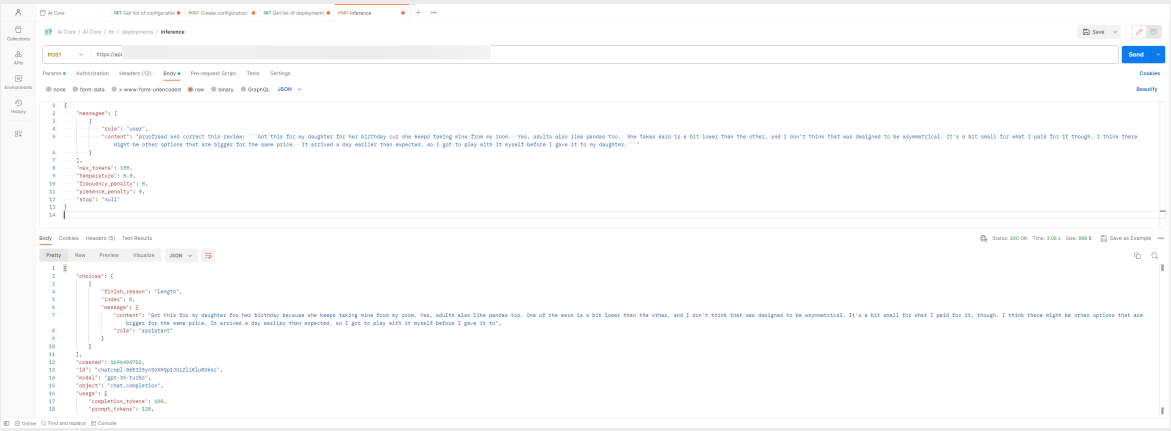


```
{
"messages":[
        {
            "role":"user",
            "content": "Proofread and correct the following text and rewrite
the corrected version. If you don't find and errors, just say \"No errors
found\". Don't use any punctuation around the text:  Yolanda has her
notebook."
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



```
{
```

```
"messages":[
        {
            "role":"user",
            "content": "proofread and correct this review: ```Got this for
my daughter for her birthday cuz she keeps taking mine from my room.  Yes,
adults also like pandas too.  She takes ears is a bit lower than the other,
and I don't think that was designed to be asymmetrical. It's a bit small for
what I paid for it though. I think there might be other options that are
bigger for the same price.  It arrived a day earlier than expected, so I got
to play with it myself before I gave it to my daughter.```"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



# Expansions

Expansions generate text based on a prompt.

## Procedure

Send a POST request to the endpoint `{{deploymentUrl}}/chat/completions?api-version={{api-version}}`.

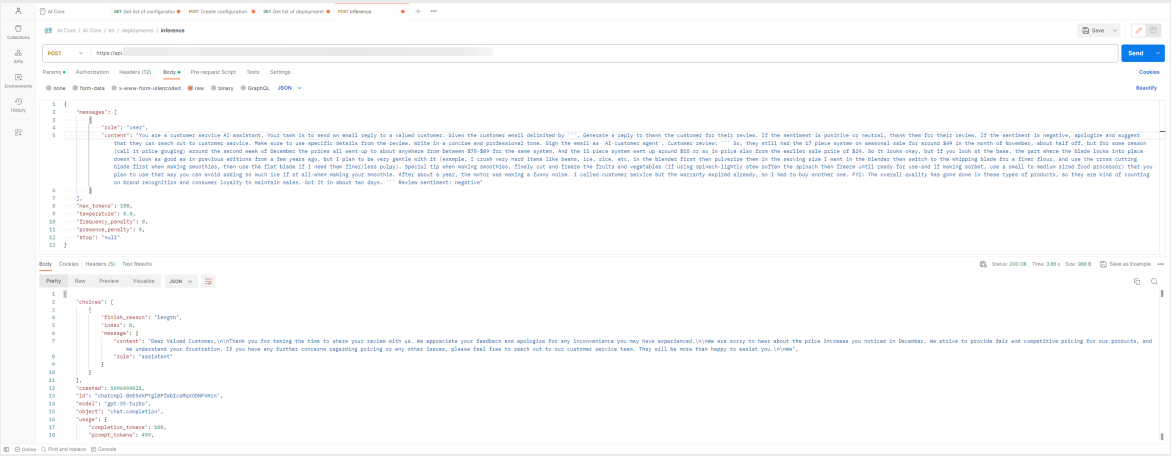
Include your query in the body.

### Example

This example generates an automated reply to a customer email.

```
{
"messages":[
        {
            "role":"user",
```

```
            "content": "You are a customer service AI assistant. Your task
is to send an email reply to a valued customer. Given the customer email
delimited by ```, Generate a reply to thank the customer for their review.
If the sentiment is positive or neutral, thank them for their review. If
the sentiment is negative, apologize and suggest that they can reach out to
customer service. Make sure to use specific details from the review. Write
in a concise and professional tone. Sign the email as `AI customer agent`.
Customer review: ```So, they still had the 17 piece system on seasonal sale
for around $49 in the month of November, about half off, but for some reason
(call it price gouging) around the second week of December the prices all
went up to about anywhere from between $70-$89 for the same system. And the
11 piece system went up around $10 or so in price also from the earlier sale
price of $29. So it looks okay, but if you look at the base, the part where
the blade locks into place doesn't look as good as in previous editions from
a few years ago, but I plan to be very gentle with it (example, I crush very
hard items like beans, ice, rice, etc. in the blender first then pulverize
them in the serving size I want in the blender then switch to the whipping
blade for a finer flour, and use the cross cutting blade first when making
smoothies, then use the flat blade if I need them finer/less pulpy). Special
tip when making smoothies, finely cut and freeze the fruits and vegetables
(if using spinach-lightly stew soften the spinach then freeze until ready
for use-and if making sorbet, use a small to medium sized food processor)
that you plan to use that way you can avoid adding so much ice if at all-
when making your smoothie. After about a year, the motor was making a funny
noise. I called customer service but the warranty expired already, so I had
to buy another one. FYI: The overall quality has gone done in these types
of products, so they are kind of counting on brand recognition and consumer
loyalty to maintain sales. Got it in about two days.``` Review sentiment:
negative"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



# Chatbot

## Context

Chatbots use input and give output in the form of conversations.

## Procedure

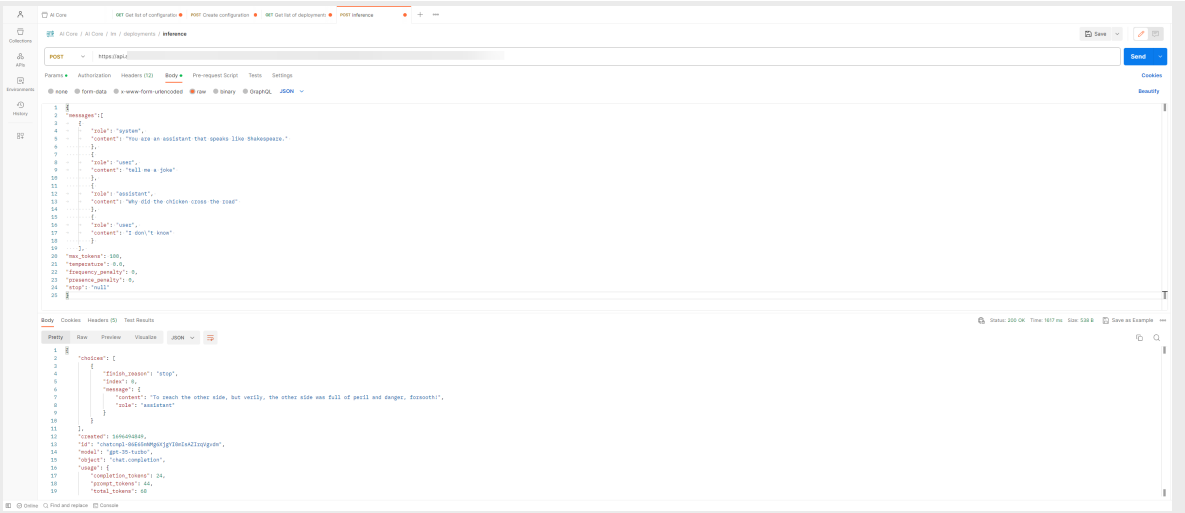
Send a POST request to the endpoint `{{deploymentUrl}}/chat/completions?api-version={{api-version}}`.

Include your query in the body. To provide more context or set a precedent, include examples of the desired outputs in your prompt.

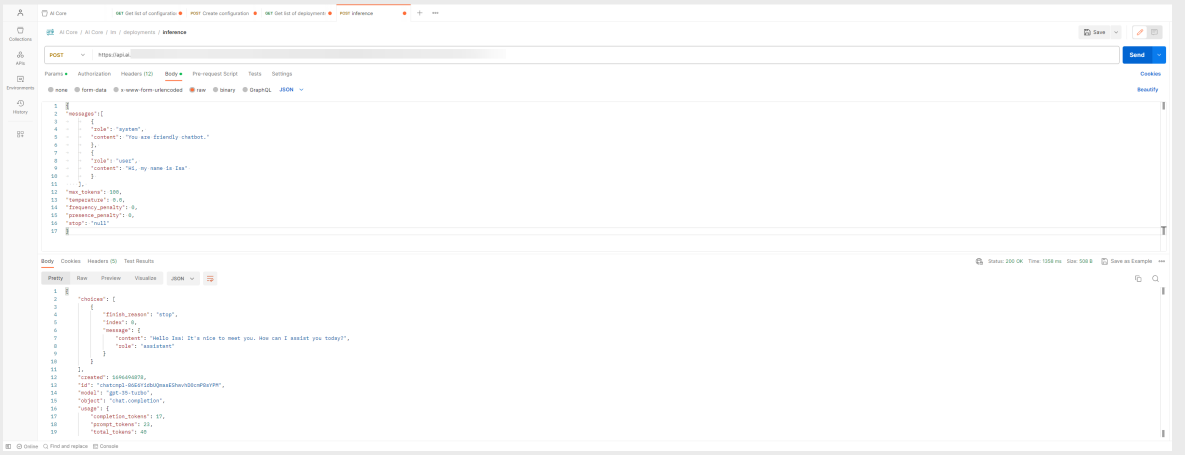
### Example

The following examples include few shot prompts for a chatbot.

```
{
"messages":[
    {
        "role": "system",
        "content": "You are an assistant that speaks like Shakespeare."
        },
        {
        "role": "user",
        "content": "tell me a joke"
        },
        {
        "role": "assistant",
        "content": "Why did the chicken cross the road"
        },
        {
        "role": "user",
        "content": "I don\"t know"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```

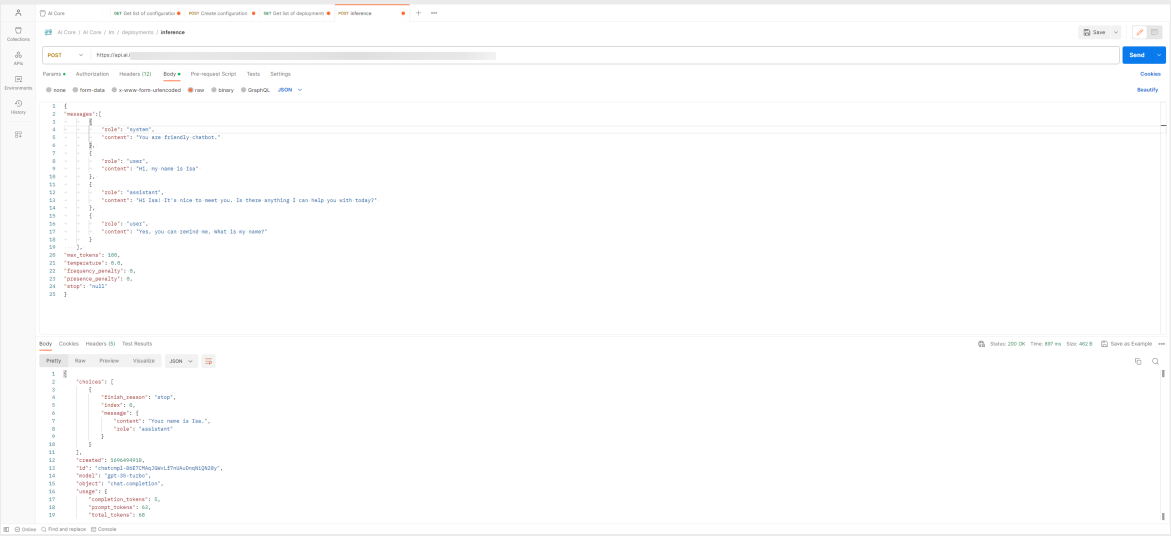


```
{
"messages":[
        {
        "role": "system",
        "content": "You are friendly chatbot."
        },
        {
        "role": "user",
        "content": "Hi, my name is Isa"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



```
{
"messages":[
        {
            "role": "system",
            "content": "You are friendly chatbot."
        },
        {
            "role": "user",
            "content": "Hi, my name is Isa"
```

```
        },
        {
            "role": "assistant",
            "content": "Hi Isa! It's nice to meet you. Is there anything I
can help you with today?"
        },
        {
            "role": "user",
            "content": "Yes, you can remind me, What is my name?"
        }
    ],
"max_tokens": 100,
"temperature": 0.0,
"frequency_penalty": 0,
"presence_penalty": 0,
"stop": "null"
}
```



## 6.2 Consume Large Language Models Using SAP AI Launchpad

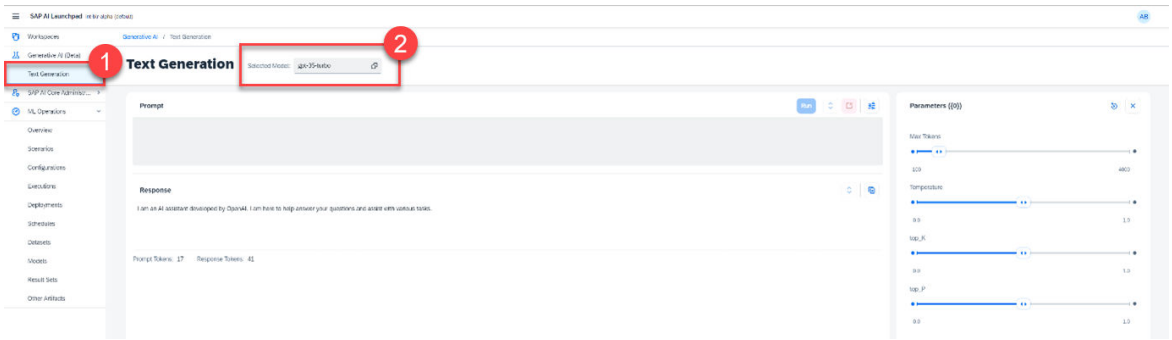
You consume a large language model (LLM) from the *Generative AI* app. The LLM can carry out natural language-related tasks such as answering questions, summarizing text, and extracting information from a body of text.

### Prerequisites

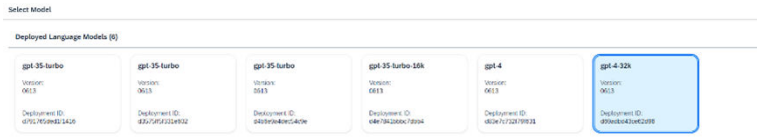
- You've completed the steps described at Activate the Generative AI Hub for SAP AI Launchpad [page 35].
- The virtual deployment for your model is running.
- You've selected the resource group that you used in the activation steps.

## Procedure

1. In SAP AI Launchpad, choose *Generative AI* and *Text Generation*.




2. Choose which model you want to use.

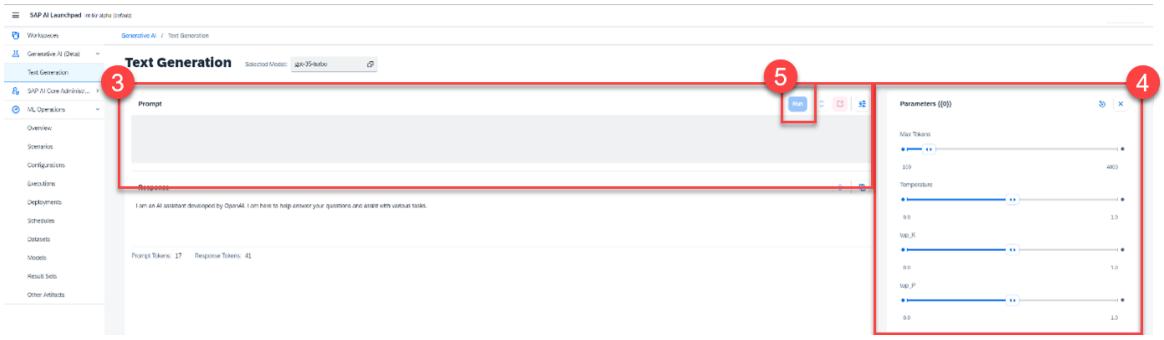



3. Enter your prompt in the *Prompt* field.

   In the *Parameters* section, you can include optional parameters such as:

   - `Max Tokens`: An integer that defines the maximum number of tokens allowed for the generated answer.
   - `Temperature`: Higher values make the output more random; lower values make it more focused and deterministic.
   - `top_K`: Introduces random sampling for generated tokens by randomly selecting the next token from the k most likely options. Set to 0.0 if repeatable output is desired.
   - `top_P`: Introduces random sampling for generated tokens by randomly selecting the next token from the smallest possible set of tokens whose cumulative probability exceeds the probability top_p. Set to 0.0 if repeatable output is desired.

   When your prompt is complete, choose run.



## Question Answering

You can ask the LLM a question and receive a response written in natural language.

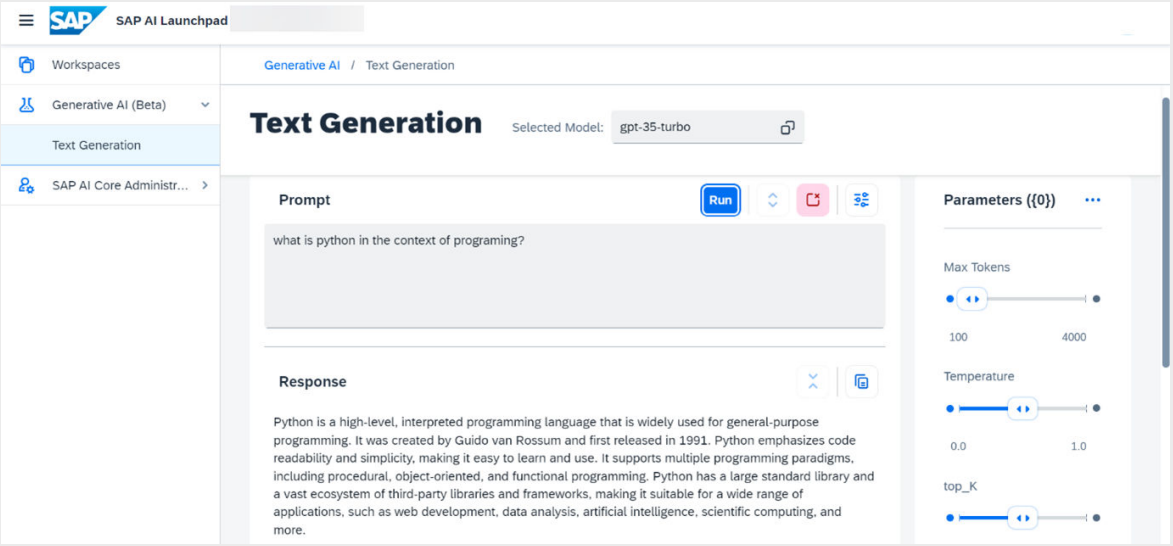
Example

Prompt:

```
What is python in the context of programing?
```

Response:

```
Python is a high-level, interpreted programming language that is widely used
for general-purpose programming. It was created by Guido van Rossum and first
released in 1991. Python emphasizes code readability and simplicity, making it
easy to learn and use. It supports multiple programming paradigms, including
procedural, object-oriented, and functional programming. Python has a large
standard library and a vast ecosystem of third-party libraries and frameworks,
making it suitable for a wide range of applications, such as web development,
data analysis, artificial intelligence, scientific computing, and more.
```




## Summarizing

You can provide the LLM with a text and ask for a summary of it.

Example

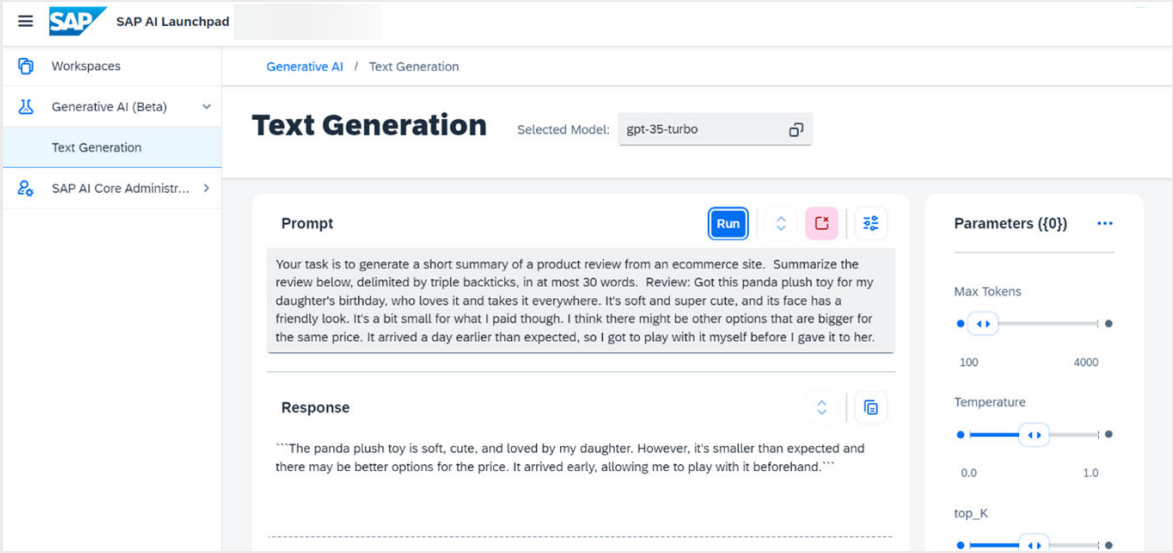
Prompt:

```
Your task is to generate a short summary of a product review from an ecommerce
site. Summarize the review below, delimited by triple backticks, in at most 30
words. Review: Got this panda plush toy for my daughter's birthday, who loves it
and takes it everywhere. It's soft and super cute, and its face has a friendly
```

**look. It's a bit small for what I paid though. I think there might be other options that are bigger for the same price. It arrived a day earlier than expected, so I got to play with it myself before I gave it to her.**

Response:

```The panda plush toy is soft, cute, and loved by my daughter. However, it's smaller than expected and there may be better options for the price. It arrived early, allowing me to play with it beforehand```



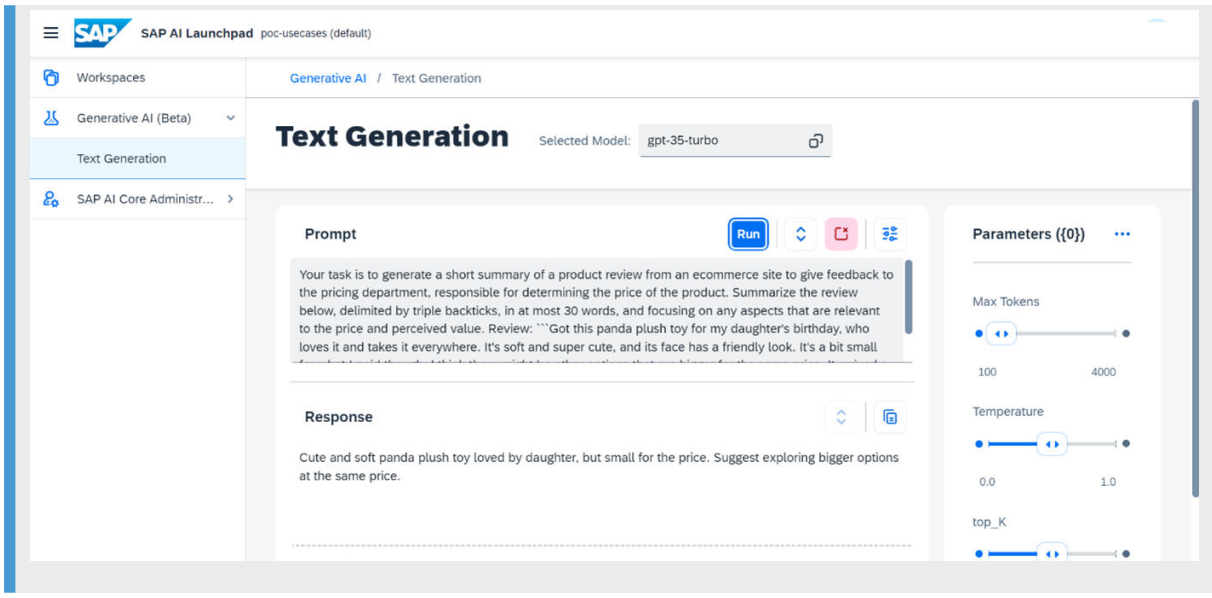
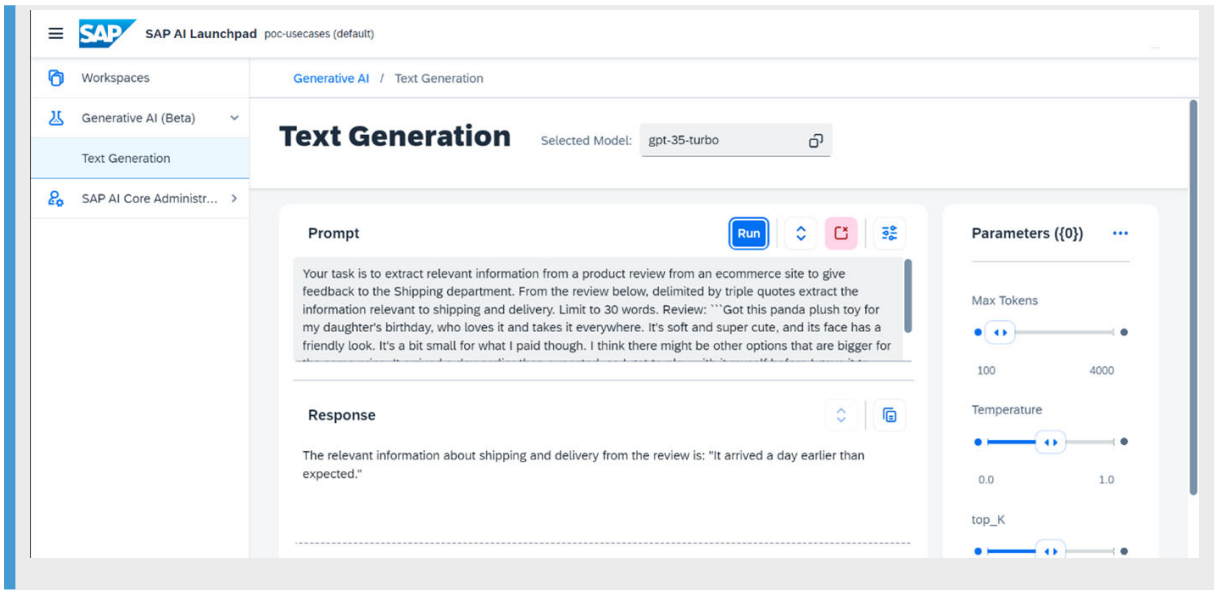
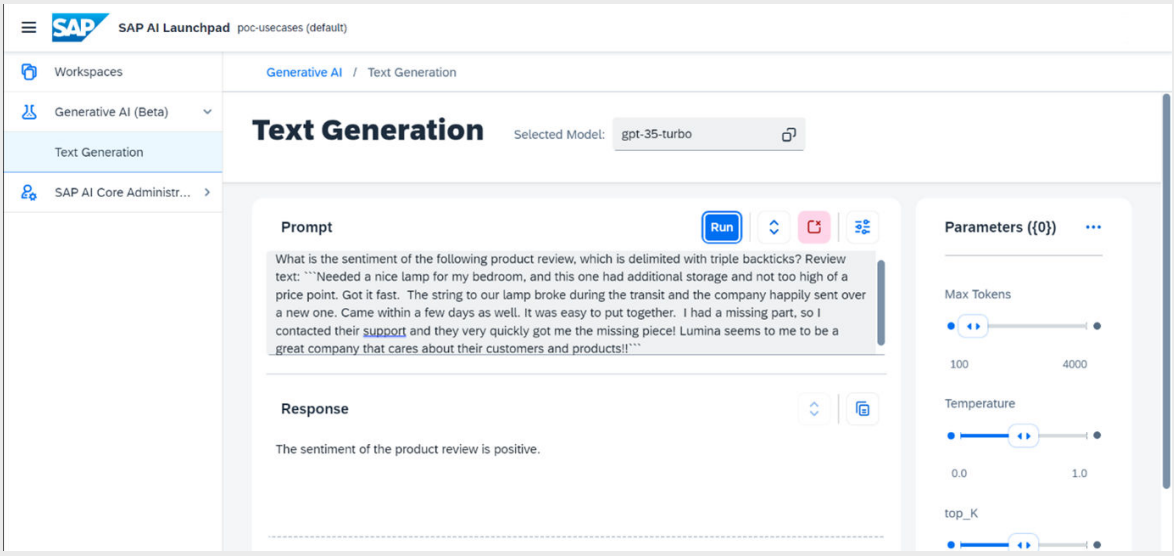
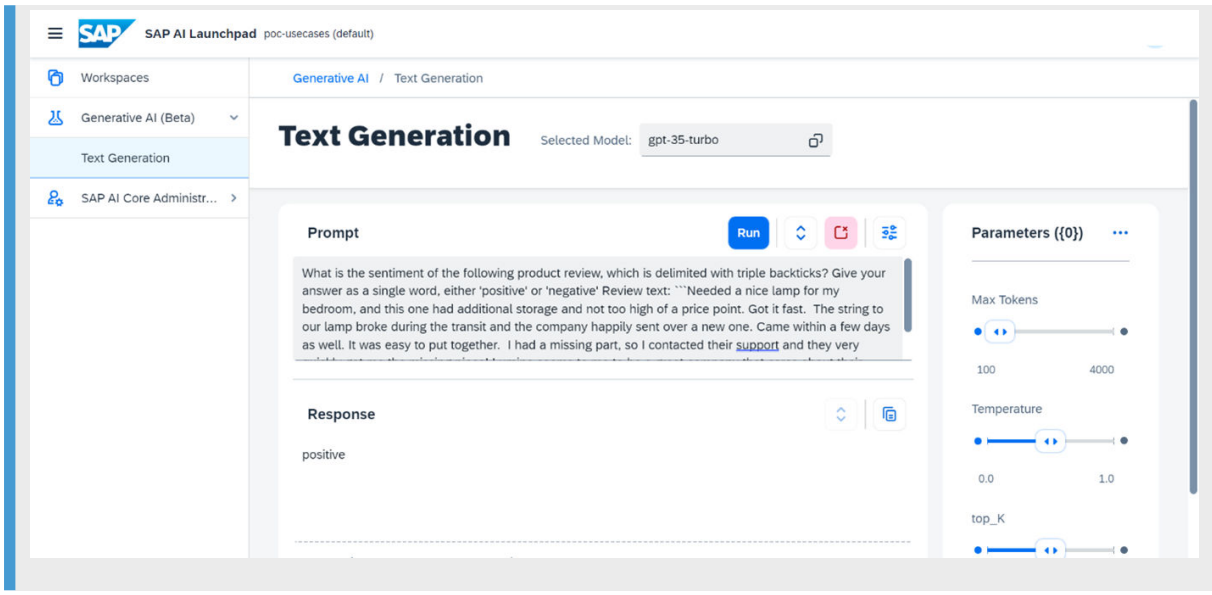
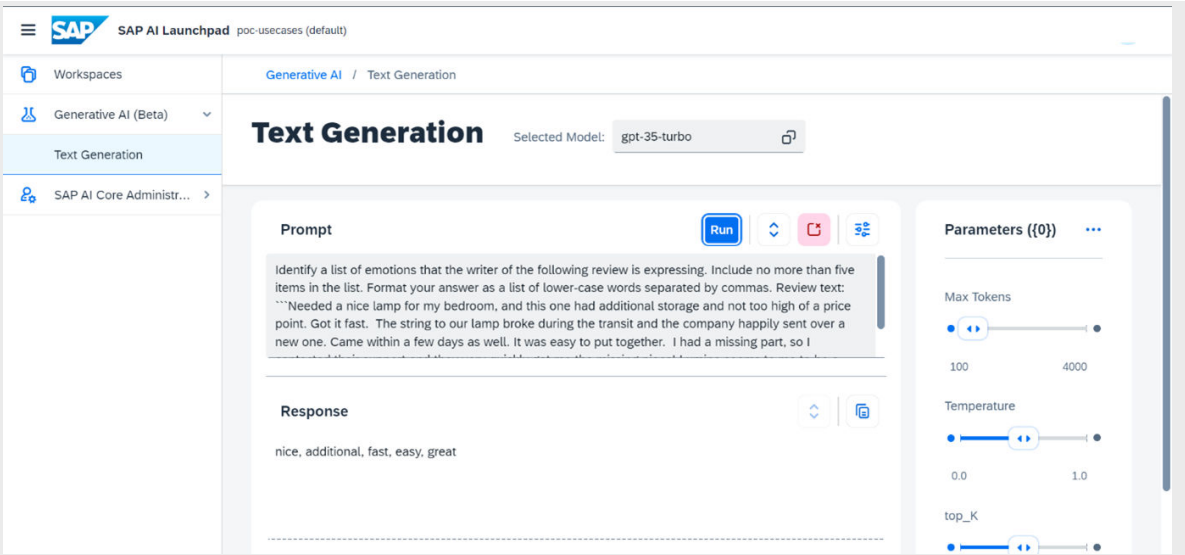

## Example

This example specifies topic focus points.

Prompt:

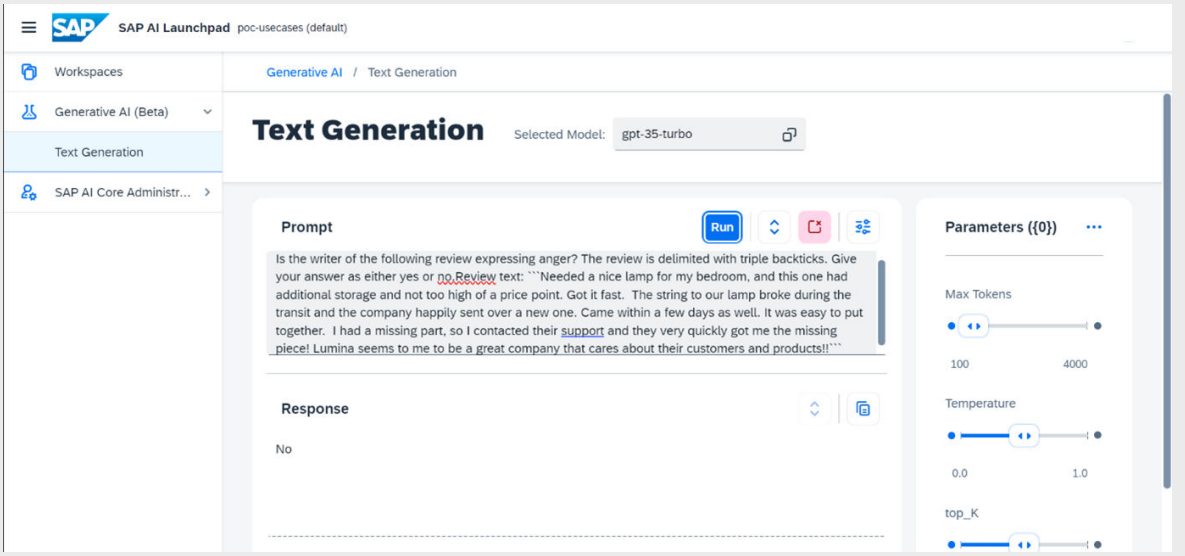
**Your task is to generate a short summary of a product review from an ecommerce site to give feedback to the pricing department, responsible for determining the price of the product. Summarize the review below, delimited by triple backticks, in at most 30 words, and focusing on any aspects that are relevant to the price and perceived value. Review: ```Got this panda plush toy for my daughter's birthday, who loves it and takes it everywhere. It's soft and super cute, and its face has a friendly look. It's a bit small for what I paid though. I think there might be other options that are bigger for the same price. It arrived a day earlier than expected, so I got to play with it myself before I gave it to her.**

Response:

Cute and soft panda plush toy loved by daughter, but small for the price. Suggest exploring bigger options at the same price.
```




## Example

This example uses “extract” instead of “summarize”.

Prompt:

```
Your task is to extract relevant information from a product review from an
ecommerce site to give feedback to the Shipping department. From the review
below, delimited by triple quotes extract the information relevant to shipping
and delivery. Limit to 30 words. Review: ```Got this panda plush toy for my
daughter's birthday,who loves it and takes it everywhere. It's soft and super
cute, and its face has a friendly look. It's a bit small for what I paid though.
I think there might be other options that are bigger for the same price. It
arrived a day earlier than expected, so I got to play with it myself before I
gave it to her.```
```

Response:

```
The relevant information about shipping and delivery from the review is: “It
arrived a day early than expected.”
```




# Inferencing

Inferencing uses the information in a given text to draw a conclusion.

### Example

This example performs a sentiment analysis on a product review.

Prompt:

```
What is the sentiment of the following product review, which is delimited with
triple backticks? Review text: ```Needed a nice lamp for my bedroom, and this
one had additional storage and not too high of a price point. Got it fast. The
string to our lamp broke during the transit and the company happily sent over
a new one. Came within a few days as well. It was easy to put together. I had
a missing part, so I contacted their support, and they very quickly got me the
missing piece! Lumina seems to me to be a great company that cares about their
customers and products!!```
```

Response:

```
The sentiment of the product review is positive.
```




## Example

This example generates the sentiment as a one word response.

Prompt:

```
What is the sentiment of the following product review, which is delimited with
triple backticks? Give your answer as a single word, either 'positive' or
'negative' Review text: ```Needed a nice lamp for my bedroom, and this one had
additional storage and not too high of a price point. Got it fast. The string to
our lamp broke during the transit and the company happily sent over a new one.
Came within a few days as well. It was easy to put together. I had a missing
part, so I contacted their support and they very quickly got me the missing
piece! Lumina seems to me to be a great company that cares about their customers
and products!!```
```

Response:

```
positive
```




### Example

This example analyzes the emotions expressed in the review.

Prompt:

```
Identify a list of emotions that the writer of the following review is
expressing. Include no more than five items in the list. Format your answer as
a list of lower-case words separated by commas. Review text: ```Needed a nice
lamp for my bedroom, and this one had additional storage and not too high of a
price point. Got it fast. The string to our lamp broke during the transit and the
company happily sent over a new one. Came within a few days as well. It was easy
to put together. I had a missing part, so I contacted their support and they very
quickly got me the missing piece! Lumina seems to me to be a great company that
cares about their customers and products!!
```

Response:

```
nice, additional, fast, easy, great
```




## Example

This example detects whether anger is present.

Prompt:

```
Is the writer of the following review expressing anger? The review is delimited
with triple backticks. Give your answer as either yes or no. Review text:
```Needed a nice lamp for my bedroom, and this one had additional storage and
not too high of a price point. Got it fast. The string to our lamp broke during
the transit and the company happily sent over a new one. Came within a few days
as well. It was easy to put together. I had a missing part, so I contacted their
support and they very quickly got me the missing piece! Lumina seems to me to be
a great company that cares about their customers and products!!```
```

Response:

No


```

## Example

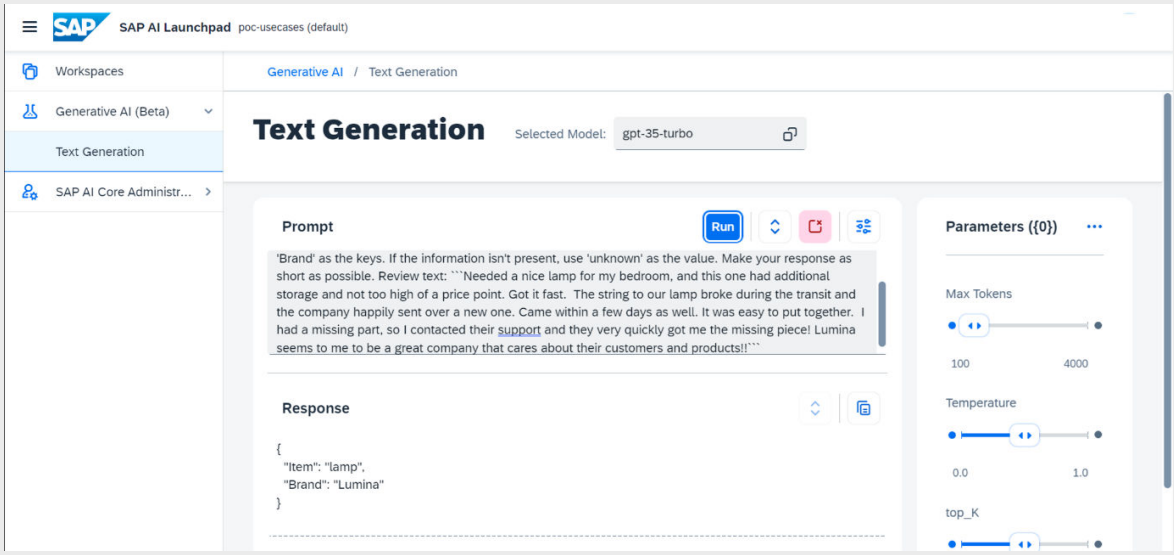
This example detects product and company names from the customer review.

Prompt:

```
Identify the following items from the review text: - Item purchased by reviewer
- Company that made the item The review is delimited with triple backticks.
Format your response as a JSON object with 'Item' and 'Brand' as the keys. If
the information isn't present, use 'unknown' as the value. Make your response as
short as possible. Review text: ```Needed a nice lamp for my bedroom, and this
one had additional storage and not too high of a price point. Got it fast. The
string to our lamp broke during the transit and the company happily sent over
a new one. Came within a few days as well. It was easy to put together. I had
a missing part, so I contacted their support and they very quickly got me the
missing piece! Lumina seems to me to be a great company that cares about their
customers and products!!``
```

Response:

```
{  "Item": "lamp",
"Brand": "Lumina"}
```




## Example

This example performs multiple tasks in a single query.

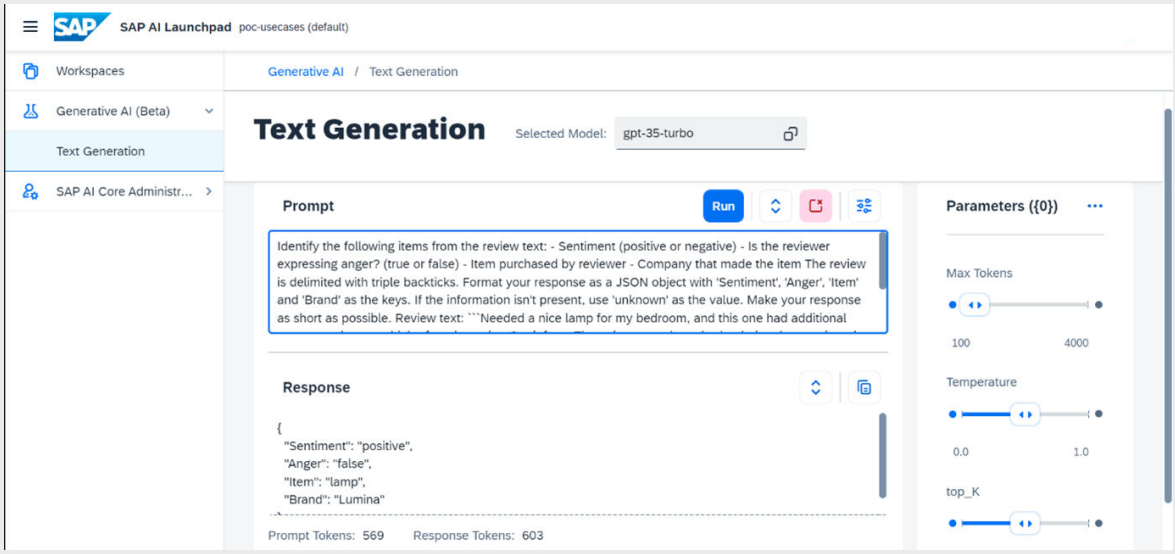
Prompt:

```
Identify the following items from the review text: - Sentiment (positive or
negative) - Is the reviewer expressing anger? (true or false) - Item purchased
by reviewer - Company that made the item The review is delimited with triple
backticks. Format your response as a JSON object with 'Sentiment', 'Anger',
'Item' and 'Brand' as the keys. If the information isn't present, use 'unknown'
as the value. Make your response as short as possible. Review text: ```Needed a
nice lamp for my bedroom, and this one had additional storage and not too high of
```

```
a price point. Got it fast. The string to our lamp broke during the transit and
the company happily sent over a new one. Came within a few days as well. It was
easy to put together. I had a missing part, so I contacted their support and they
very quickly got me the missing piece! Lumina seems to me to be a great company
that cares about their customers and products!!```
```

Response:

```
{
"Sentiment": "positive",
"Anger": "false",
"Item": "lamp",
"Brand": "Lumina"
}
```




## Example

This example identifies the five topics discussed in a story.

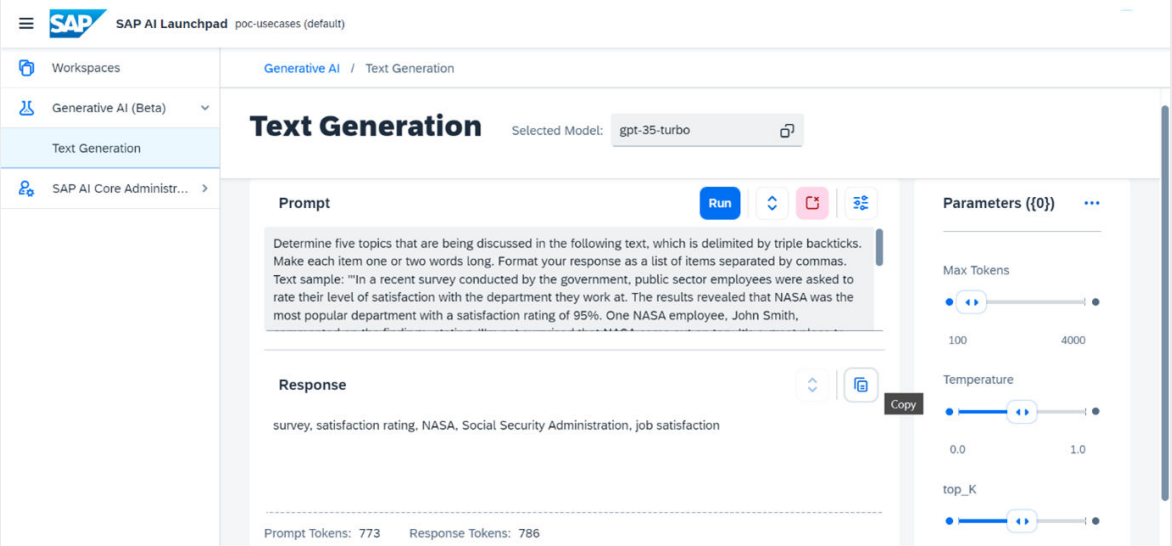
Prompt:

```
Determine five topics that are being discussed in the following text, which is
delimited by triple backticks. Make each item one or two words long. Format
your response as a list of items separated by commas. Text sample: '''In a
recent survey conducted by the government, public sector employees were asked to
rate their level of satisfaction with the department they work at. The results
revealed that NASA was the most popular department with a satisfaction rating of
95%. One NASA employee, John Smith, commented on the findings, stating, 'I'm not
surprised that NASA came out on top. It's a great place to work with amazing
people and incredible opportunities. I'm proud to be a part of such an innovative
organization.' The results were also welcomed by NASA's management team, with
Director Tom Johnson stating, 'We are thrilled to hear that our employees are
satisfied with their work at NASA. We have a talented and dedicated team who work
tirelessly to achieve our goals, and it's fantastic to see that their hard work
is paying off.' The survey also revealed that the Social Security Administration
had the lowest satisfaction rating, with only 45% of employees indicating they
```

**were satisfied with their job. The government has pledged to address the concerns raised by employees in the survey and work towards improving job satisfaction across all departments.**

Response:

```
survey, satisfaction rating, NASA, Social Security Administration, job
satisfaction
```




# Transformations

Transformations transform a given text into another language or register.

### Example

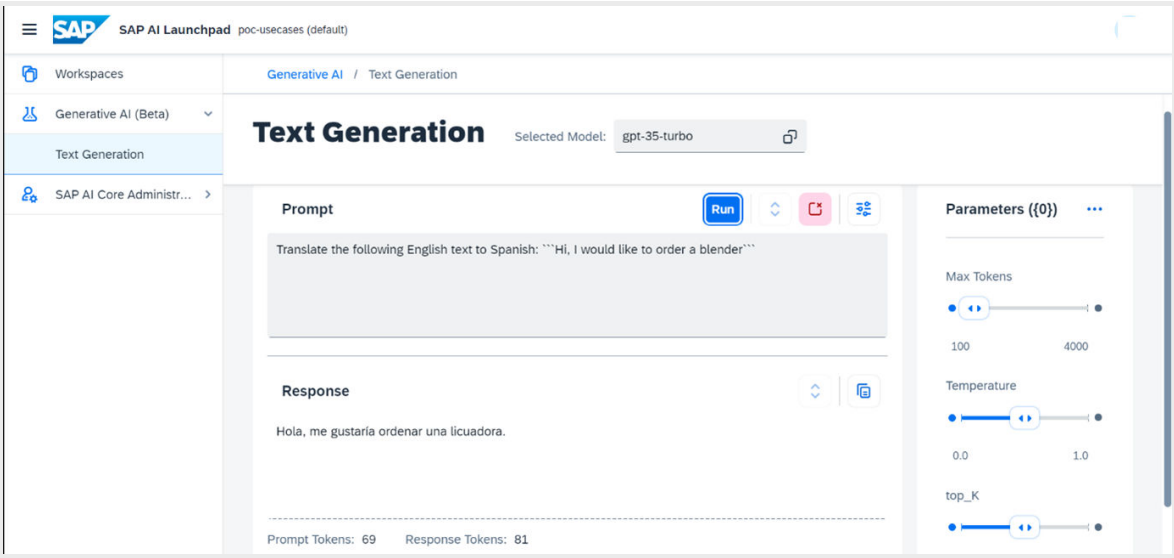
This example translates text from English to Spanish.

Prompt:

**Translate the following English text to Spanish: ```Hi, I would like to order a blender```**

Response:

```
Hola, me gustaría ordenar una licuadora.
```




## Example

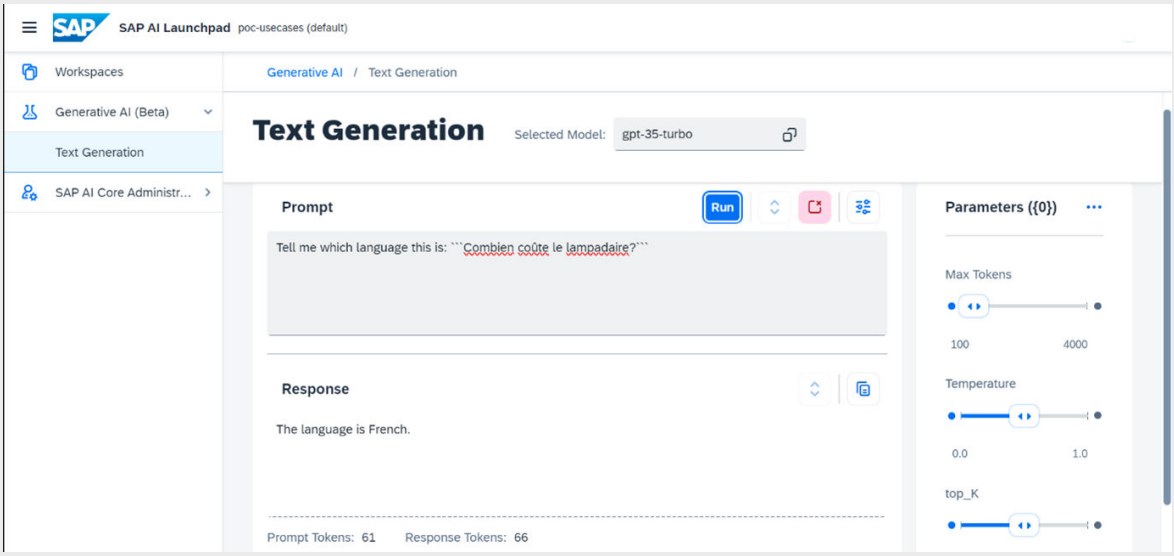
This example detects the language that the text is written in.

Prompt:

```
Tell me which language this is: ```Combien coûte le lampadaire?```
```

Response:

```
The language is French.
```




## Example

This example translates the given text into multiple languages.

Prompt:

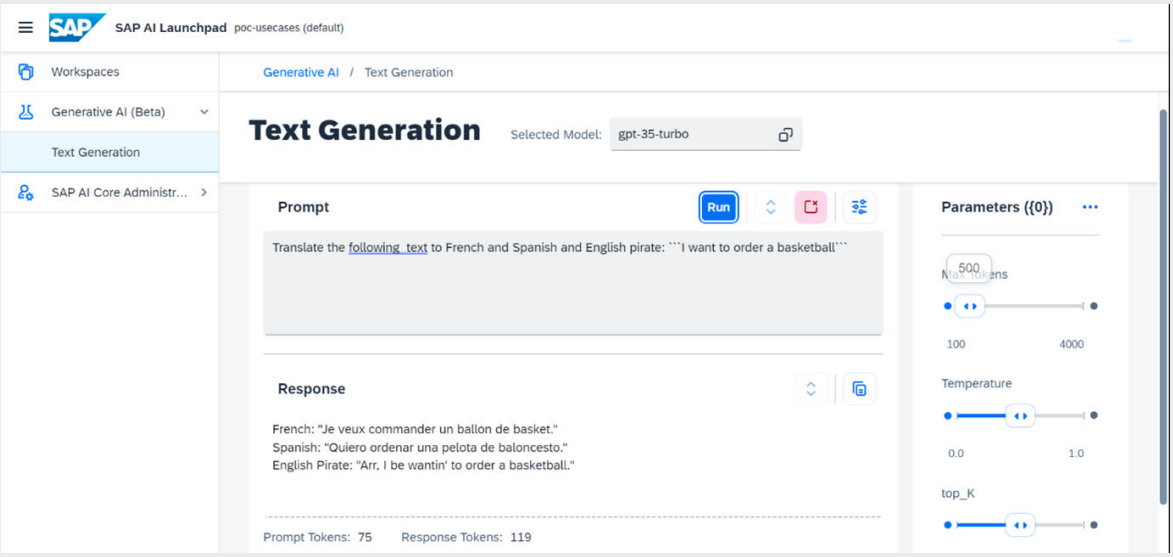
```
Translate the following text to French and Spanish and English pirate: ```I want to order a basketball```
```

Response:

```
French: "Je veux commander un ballon de basket."

Spanish: "Quiero ordenar una pelota de baloncesto."

English Pirate: "Arrr, I be wantin' to order a basketball, matey!"
```




## Example

These examples translate both the language and register of the text.

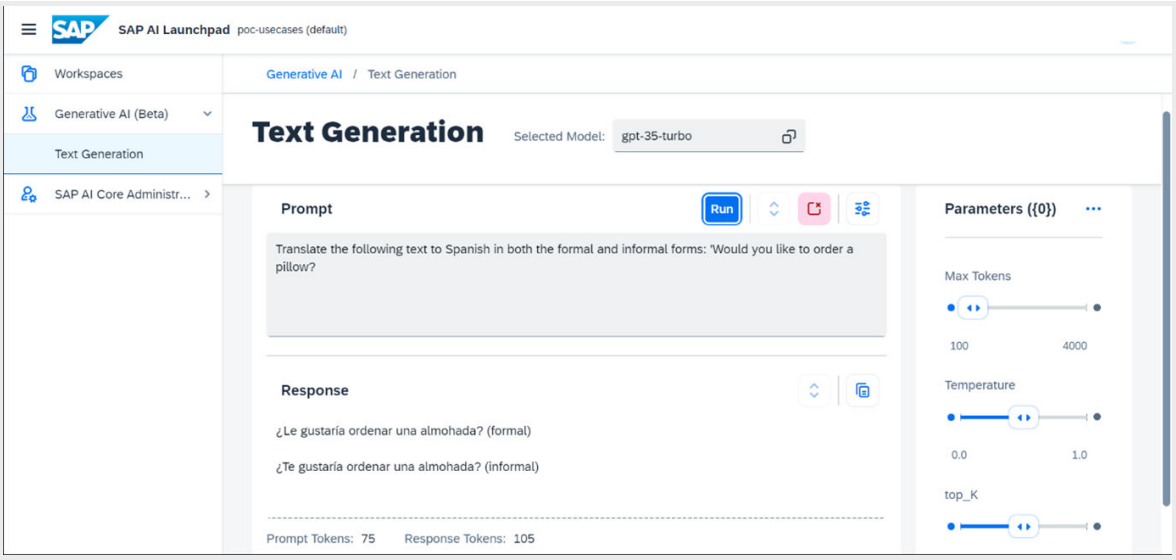
Prompt:

```
Translate the following text to Spanish in both the formal and informal forms:
'Would you like to order a pillow?
```

Response:

```
¿Le gustaría ordenar una almohada? (formal)

¿Te gustaría ordenar una almohada? (informal)
```




Prompt:

**Translate the following from slang to a business letter: 'Dude, This is Joe, check out this spec on this standing lamp.**

Response:

Dear Sir/Madam,

I hope this letter finds you well. My name is Joe, and I am writing to bring your attention to a specific standing lamp that I believe may be of interest to you.

I have recently come across a specification for this particular standing lamp, and I must say, it is quite impressive. The design, functionality, and overall quality of this lamp are truly remarkable.

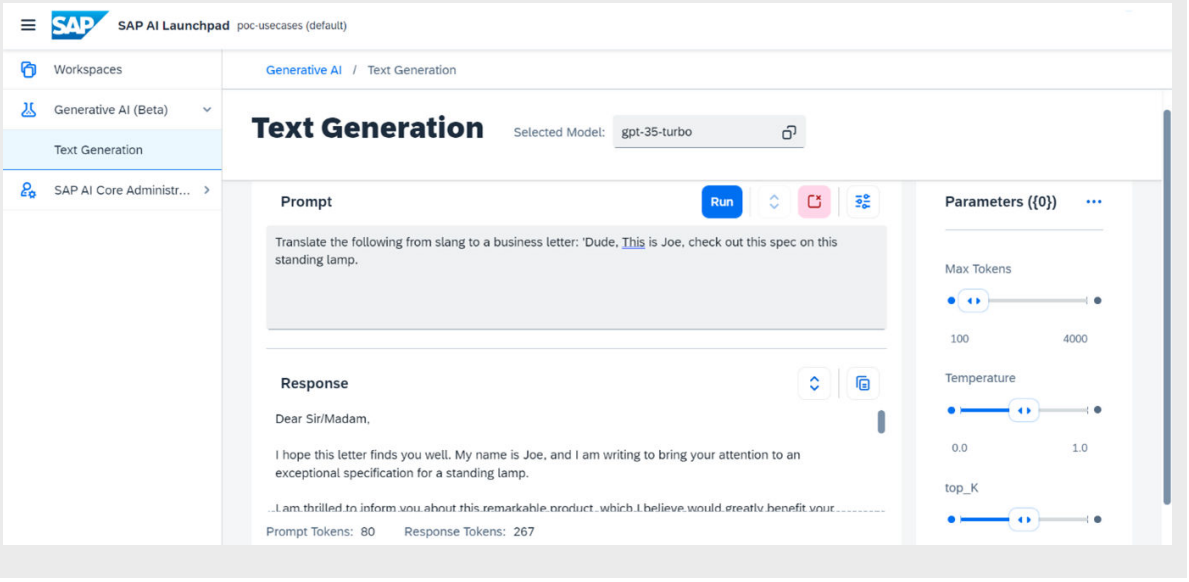
I would highly recommend taking a closer look at this spec, as I believe it presents a valuable opportunity for your business. The standing lamp not only possesses aesthetic appeal but also offers practicality and durability, making it an ideal addition to your product range.

Should you require any further information or assistance regarding this standing lamp, please do not hesitate to contact me. I would be more than happy to provide you with additional details or arrange a meeting to discuss this opportunity further.

Thank you for your time and consideration. I look forward to hearing from you soon.

Sincerely,

Joe




## Example

This example translates between output formats. The prompt describes both the input and output format.

Prompt:

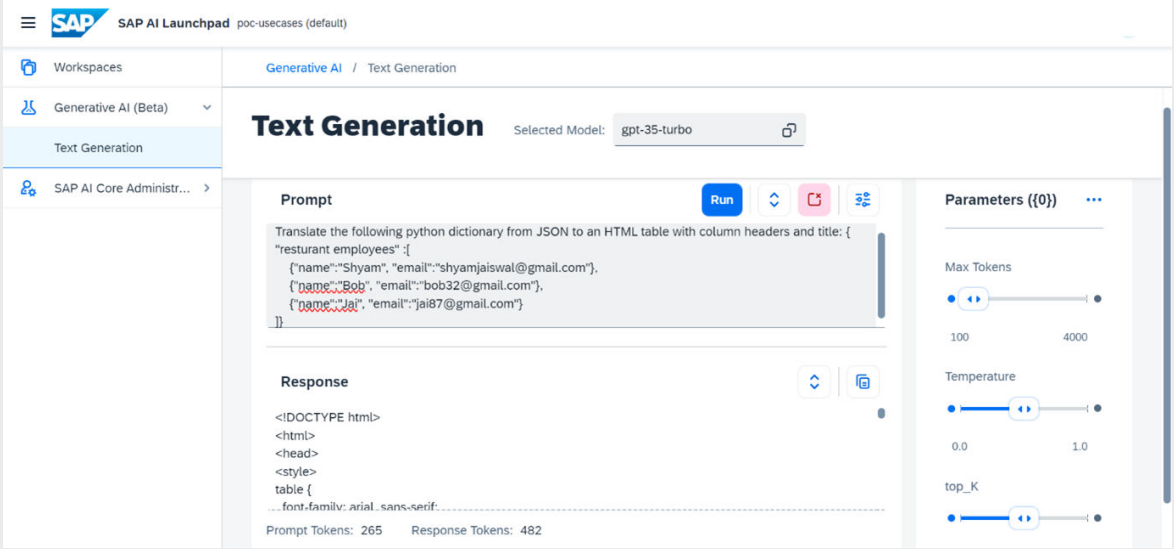
**`Translate the following python dictionary from JSON to an HTML table with column headers and title:`**

```
{ "resturant employees" :[
{"name":"Shyam", "email":"shyamjaiswal@gmail.com"},
{"name":"Bob", "email":"bob32@gmail.com"},
{"name":"Jai", "email":"jai87@gmail.com"}
]}
```

Response:

```
 <!DOCTYPE html>
<html>
<head>
<style>
table {
  font-family: arial, sans-serif;
  border-collapse: collapse;
  width: 100%;
}
td, th {
  border: 1px solid #dddddd;
  text-align: left;
  padding: 8px;
}
tr: nth-child(even) {
  background-color: #dddddd;
}
</style>
</head>
<body>
<h2>Restaurant Employees</h2>
<table>
  <tr>
    <th>Name</th>
    <th>Email</th>
  </tr>
```

```
  <tr>
    <td>Shyam</td>
    <td>shyamjaiswal@gmail.com</td>
  </tr>
  <tr>
    <td>Bob</td>
    <td>bob32@gmail.com</td>
  </tr>
  <tr>
    <td>Jai</td>
    <td>jai87@gmail.com</td>
  </tr>
</table>
</body>
</html>
```




## Example

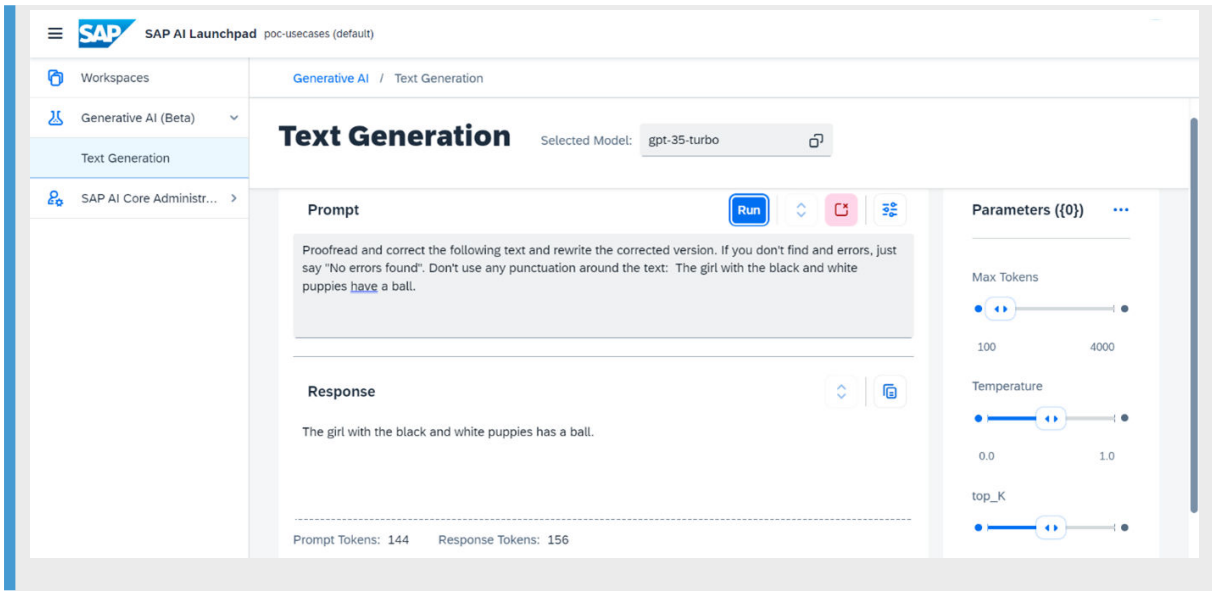
In these examples, a text is proofread. The text can be proofread and corrected, or simply proofread.

Prompt:

**`Proofread and correct the following text and rewrite the corrected version. If you don't find and errors, just say \"No errors found\". Don't use any punctuation around the text: The girl with the black and white puppies have a ball.`**

Response:

`The girl with the black and white puppies has a ball.`

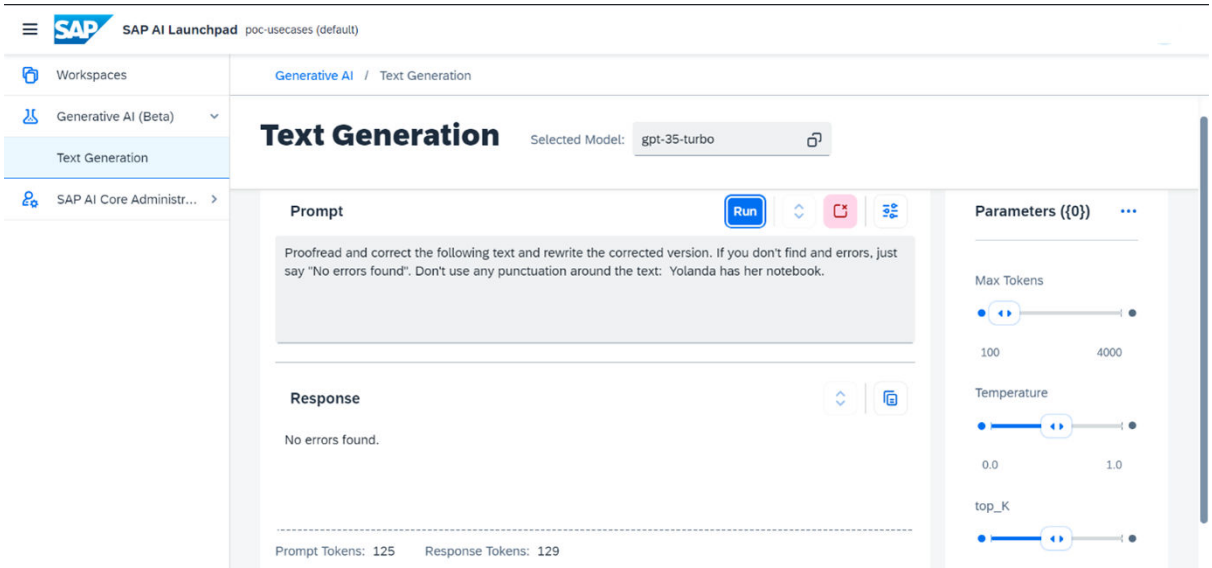



Prompt:

**Proofread and correct the following text and rewrite the corrected version. If you don't find and errors, just say \"No errors found\". Don't use any punctuation around the text: Yolanda has her notebook.**

Response:

No errors found

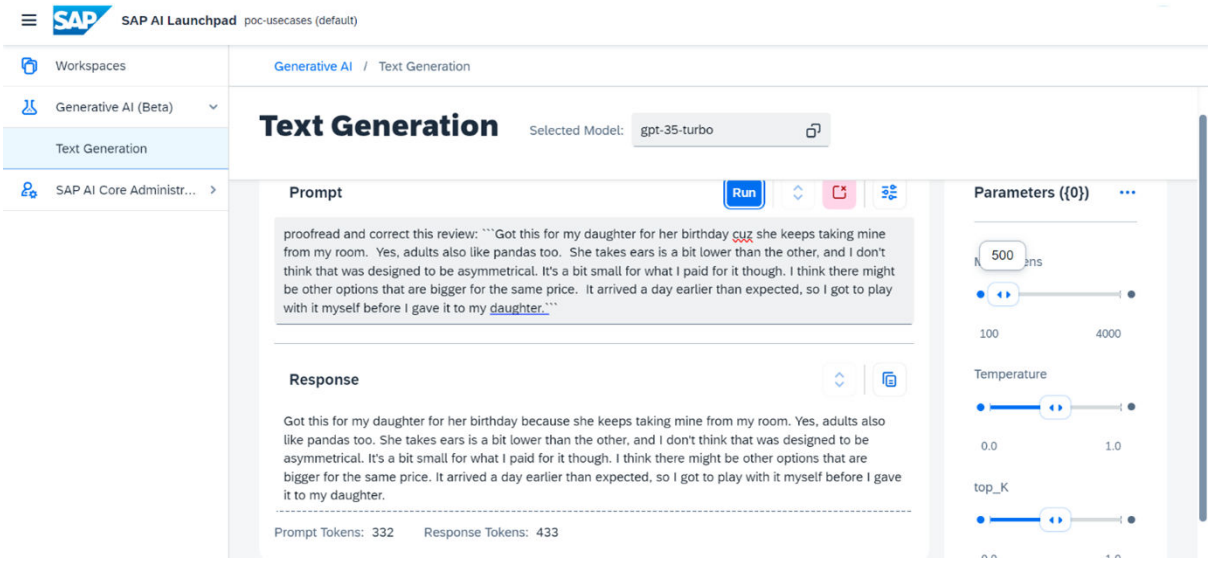



Prompt:

**Proofread and correct this review: ```Got this for my daughter for her birthday cuz she keeps taking mine from my room. Yes, adults also like pandas too. She takes ears is a bit lower than the other, and I don't think that was designed to be asymmetrical. It's a bit small for what I paid for it though. I think there might be other options that are bigger for the same price. It arrived a day earlier than expected, so I got to play with it myself before I gave it to my daughter.```**

Response:

```
Got this for my daughter for her birthday because she keeps taking mine from my
room. Yes, adults also like pandas too. She takes ears is a bit lower than the
other, and I don't think that was designed to be asymmetrical. It's a bit small for
what I paid for it though. I think there might be other options that are bigger for
the same price. It arrived a day earlier than expected, so I got to play with it
myself before I gave it to my daughter.
```




# Expansions

Expansions generate text based on a prompt.

> ✤ Example
>
> This example generates an automated reply to a customer email.
>
> Prompt:
>
> ```
> You are a customer service AI assistant. Your task is to send an email reply to
> a valued customer. Given the customer email delimited by ```, Generate a reply
> to thank the customer for their review. If the sentiment is positive or neutral,
> thank them for their review. If the sentiment is negative, apologize and suggest
> that they can reach out to customer service. Make sure to use specific details
> from the review. Write in a concise and professional tone. Sign the email as
> `AI customer agent`. Customer review: ```So, they still had the 17 piece system
> on seasonal sale for around $49 in the month of November, about half off, but
> for some reason (call it price gouging) around the second week of December the
> prices all went up to about anywhere from between $70-$89 for the same system.
> And the 11 piece system went up around $10 or so in price also from the earlier
> sale price of $29. So it looks okay, but if you look at the base, the part where
> the blade locks into place doesn’t look as good as in previous editions from a
> few years ago, but I plan to be very gentle with it (example, I crush very hard
> ```

**items like beans, ice, rice, etc. in the blender first then pulverize them in the serving size I want in the blender then switch to the whipping blade for a finer flour, and use the cross cutting blade first when making smoothies, then use the flat blade if I need them finer/less pulpy). Special tip when making smoothies, finely cut and freeze the fruits and vegetables (if using spinach-lightly stew soften the spinach then freeze until ready for use-and if making sorbet, use a small to medium sized food processor) that you plan to use that way you can avoid adding so much ice if at all-when making your smoothie. After about a year, the motor was making a funny noise. I called customer service but the warranty expired already, so I had to buy another one. FYI: The overall quality has gone done in these types of products, so they are kind of counting on brand recognition and consumer loyalty to maintain sales. Got it in about two days.```**
**Review sentiment: negative.**

Response:

Dear Valued Customer,

Thank you for taking the time to share your feedback with us. We truly appreciate your review and your loyalty to our brand.

We sincerely apologize for any inconvenience you experienced regarding the pricing changes of our 17 piece system in December. We understand how frustrating it can be when prices fluctuate unexpectedly. We strive to offer competitive prices and ensure transparency in our pricing policies. If you have any further concerns or questions regarding the pricing, please do not hesitate to reach out to our customer service team, who will be more than happy to assist you.

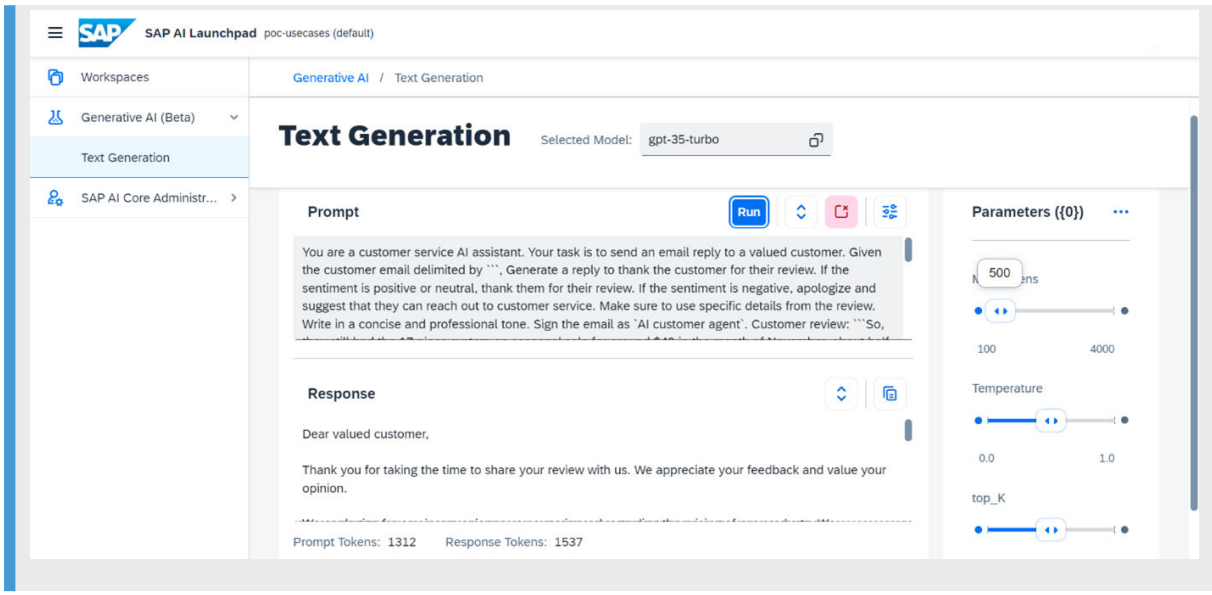
We also appreciate your valuable feedback regarding the base of the system. We continuously strive to improve our products and your feedback helps us identify areas where we can make enhancements. Rest assured that we take your comments seriously and will forward them to our product development team for further consideration.

In regards to the motor issue you encountered, we sincerely apologize for any inconvenience caused. We understand how important it is to have a reliable product. If you encounter any issues with your new purchase or have any further questions, please don't hesitate to contact our customer service team. They will be glad to assist you and provide a suitable solution.

Thank you again for your review and for choosing our products. We greatly appreciate your support and loyalty. If there is anything else we can assist you with, please feel free to let us know.

Best regards,

AI customer agent


SAP AI Launchpad poc-usecases (default)
Workspaces
Generative AI (Beta)
Text Generation
SAP AI Core Administr...
Generative AI / Text Generation
Text Generation
Selected Model: gpt-35-turbo
Prompt
Run
You are a customer service AI assistant. Your task is to send an email reply to a valued customer. Given the customer email delimited by ```, Generate a reply to thank the customer for their review. If the sentiment is positive or neutral, thank them for their review. If the sentiment is negative, apologize and suggest that they can reach out to customer service. Make sure to use specific details from the review. Write in a concise and professional tone. Sign the email as `AI customer agent`. Customer review: ```So,
Response
Dear valued customer,
Thank you for taking the time to share your review with us. We appreciate your feedback and value your opinion.
Prompt Tokens: 1312
Response Tokens: 1537
Parameters ({0})
500
100
4000
Temperature
0.0
1.0
top_K

# 7 Stopping or Deleting a Virtual Deployment

## 7.1 Stop or Delete a Virtual Deployment in SAP AI Core

### Stop a Virtual Deployment

#### Procedure

Send a PATCH request to the endpoint `{{apiurl}}/v2/lm/deployments/{{deploymentid}}`. Include the follwing data in your request:

```
{
"targetStatus": "STOPPED"
}
```

### Delete a Virtual Deployment

#### Procedure

Send a DELETE request to the endpoint `{{apiurl}}/v2/lm/deployments/{{deploymentid}}`.

# 7.2 Stop or Delete a Virtual Deployment in SAP AI Launchpad

## Stop a Virtual Deployment

### Procedure

1. Navigate to the virtual deployment's details.
2. Choose *Stop* in the header. A *Warning* dialog box appears.
3. Choose *Stop* to stop running the virtual deployment.

## Delete a Virtual Deployment

### Procedure

1. Navigate to the virtual deployment's details.
2. Choose *Delete* in the header. A *Warning* dialog box appears.
3. Choose *Delete* to confirm the deletion.

# Important Disclaimers and Legal Information

## Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

## Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

## Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

## Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

## Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

© 2023 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see https://www.sap.com/about/legal/trademark.html for additional trademark information and notices.

THE BEST RUN SAP