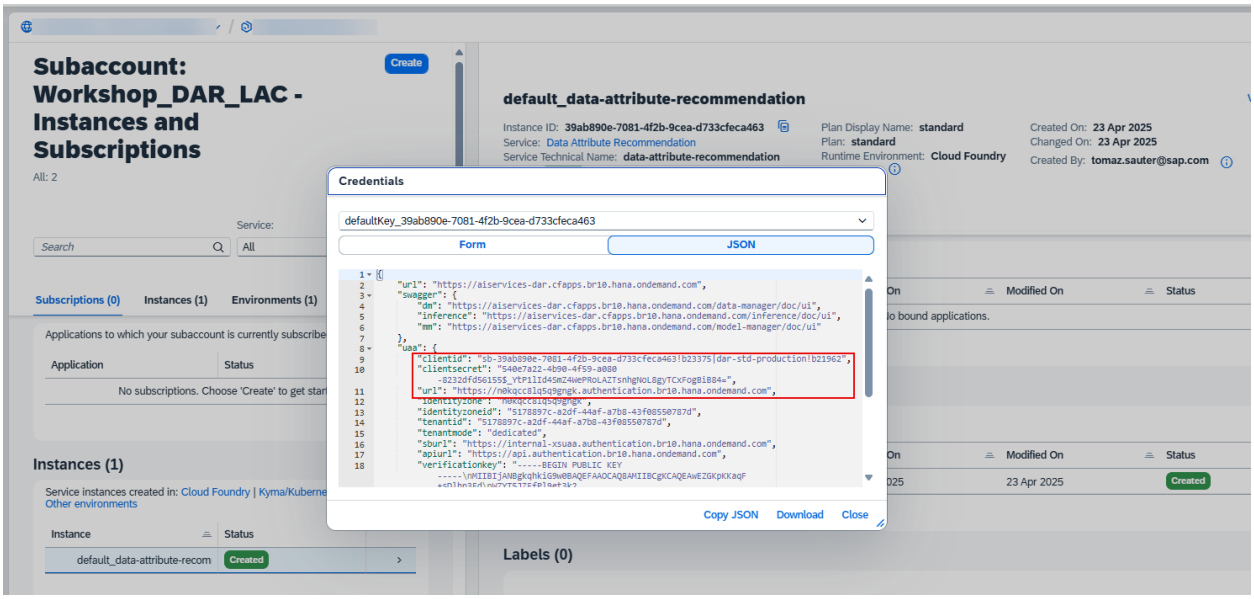
Exercise 2: Data Attribute Recommendation

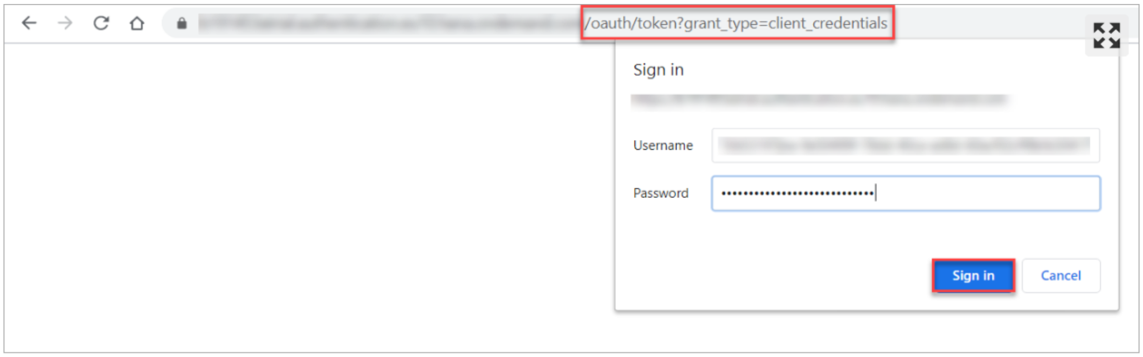
Step 1: Get OAuth Access Token for Data Attribute Recommendation via Web Browser

To get your OAuth access token you will need the values listed below from the service key you have download from the booster.

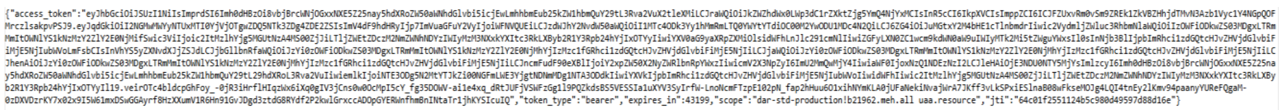


Step 2: Get OAuth access token

1. Add `/oauth/token?grant_type=client_credentials` to the `url` value (from inside the `uaa` section of the service key), paste it in any web browser and choose **Enter**.
2. Enter the `clientid` value from your service key as **Username**.
3. Enter the `clientsecret` value from your service key as **Password**.
4. Click **Sign in** or **OK** (depending on the web browser you work with).



You should receive a response like the following:


{"access_token":"eyJhbGciOiJSUzI1NiIsImprdSI6Imh0dHBzOi8vbjBrcWNjOGxxNXE5Z25nay5hdXRoZW50aWNhdGlvbi5icjEwLmhhbmEub25kZW1hbmQuY29tL3Rva2VuX2tleXMiLCJraWQiOiJkZWZhdWx0LWp3dC1rZXktZjg5YmQ4NjYxMCIsInR5cCI6IkpXVCIsImppZCI6ICJFZUxvRm0vSm9ZREk1ZkVBZHhjdTMvN3Azb1Vyc1Y4NGpQOF
","token_type":"bearer","expires_in":43199,"scope":"dar-std-production!b21962.meh.all uaa.resource","jti":"64c01f2551124b5c980d49597d88d16e"}

Once the token has been generated, make sure to copy and save it locally for easy access. The token is valid for 12 hours (43199 seconds). After that, you'll need to generate a new one.

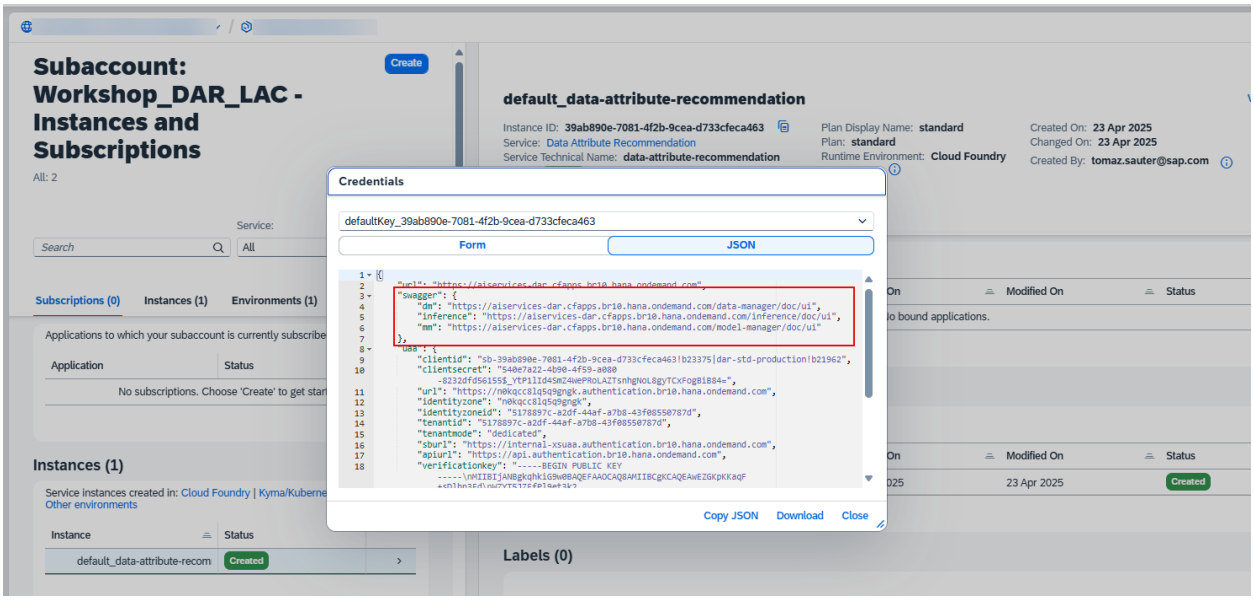
Step 3: Use an Invoice Object Recommendation (IOR) Dataset Schema to Upload Training Data to Data Attribute Recommendation

3.1

Authorize Swagger UI

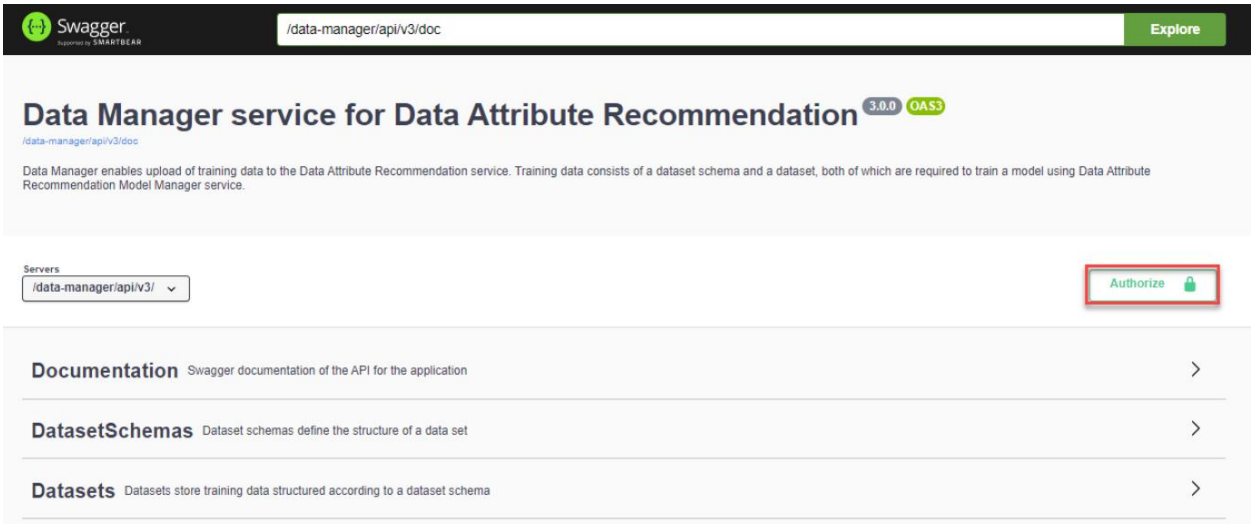
You'll use Swagger UI, via any web browser, to call the Data Attribute Recommendation APIs. Swagger UI allows developers to effortlessly interact and try out every single operation an API exposes for easy consumption. For more information, see **Swagger UI**.

In the service key you created for Data Attribute Recommendation in the previous step you will find a section called swagger (as highlighted in the image below) with three entries, called dm (data manager), mm (model manager) and inference. You'll use all three Swagger UIs throughout the exercise.



For this tutorial, copy the URL of the Swagger UI for dm and open it in a browser tab.

To be able to use the Swagger UI endpoints, you need to authorize yourself. In the top right corner, click **Authorize**.



Get the `access_token` value created in the previous step then add **Bearer** (with capitalized “B”) in front of it, and enter in the **Value** field.

```
Bearer <access_token>
```

Click **Authorize** and then click **Close**.

3.2

Create dataset schema

Now, you need to create a new dataset schema. A dataset schema describes the structure of datasets.

Use the following dataset schema:

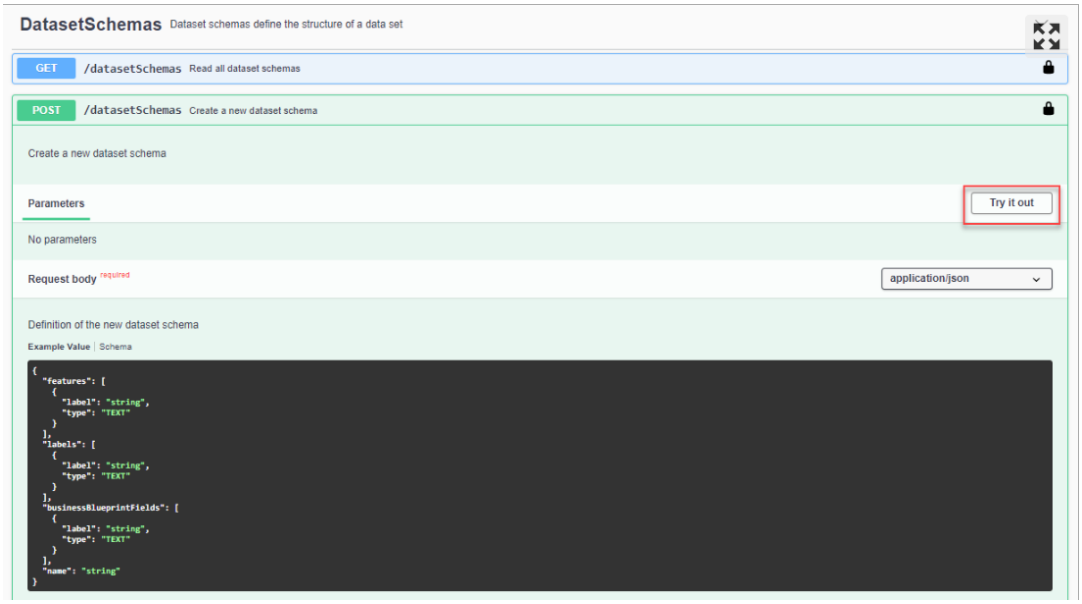
```
{
   "features":[
      {
         "label":"BUKRS",
         "type":"CATEGORY"
      },
      {
         "label":"BELNR",
         "type":"CATEGORY"
      },
      {
         "label":"GJAHR",
         "type":"NUMBER"
      },
      {
         "label":"BUZEI",
         "type":"NUMBER"
      },
      {
         "label":"KOART",
         "type":"CATEGORY"
      },
      {
         "label":"WRBTR",
         "type":"NUMBER"
      },
      {
         "label":"LIFNR",
         "type":"CATEGORY"
      },
      {
         "label":"BLART",
         "type":"CATEGORY"
      },
      {
         "label":"BUDAT",
         "type":"CATEGORY"
```

```
        },
        {
            "label":"MWSKZ",
            "type":"CATEGORY"
        }
    ],
    "labels":[
        {
            "label":"HKONT",
            "type":"CATEGORY"
        }
    ],
    "businessBlueprintFields":[
        {
            "label":"SHKZG",
            "type":"CATEGORY"
        },
        {
            "label":"BUZID",
            "type":"CATEGORY"
        },
        {
            "label":"BSCHL",
            "type":"CATEGORY"
        },
        {
            "label":"STBLG",
            "type":"CATEGORY"
        }
    ],
    "name":"ior-dataset-schema"
}
```

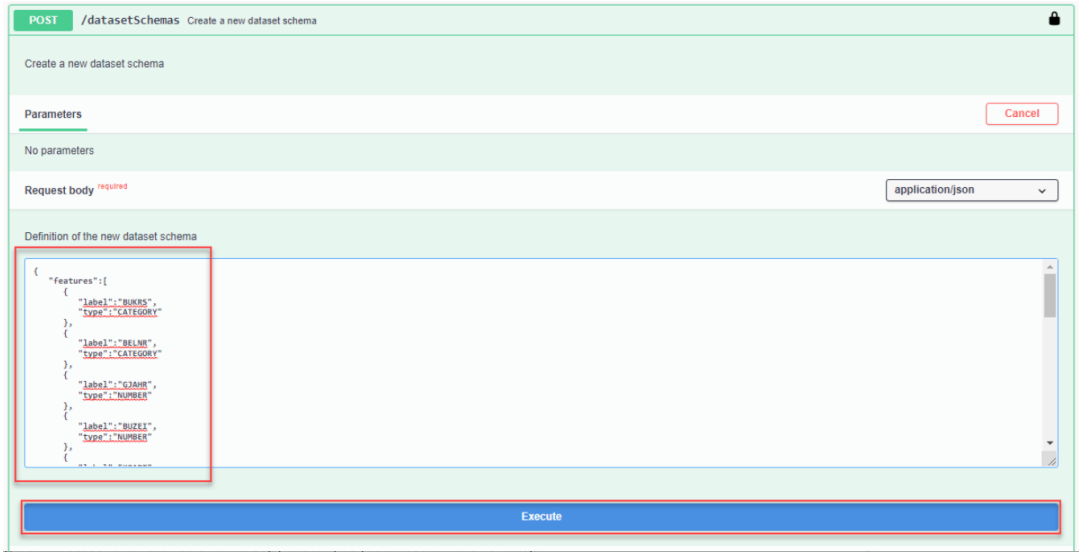

The schema is divided into `features` and `labels`. The features are the inputs for the machine learning model whereas the labels are the fields that will be predicted. Thus, this schema provides as input training fields such as BUKRS for company code and wants to predict G/L accounts (HKONT).

To create the dataset schema, proceed as follows:

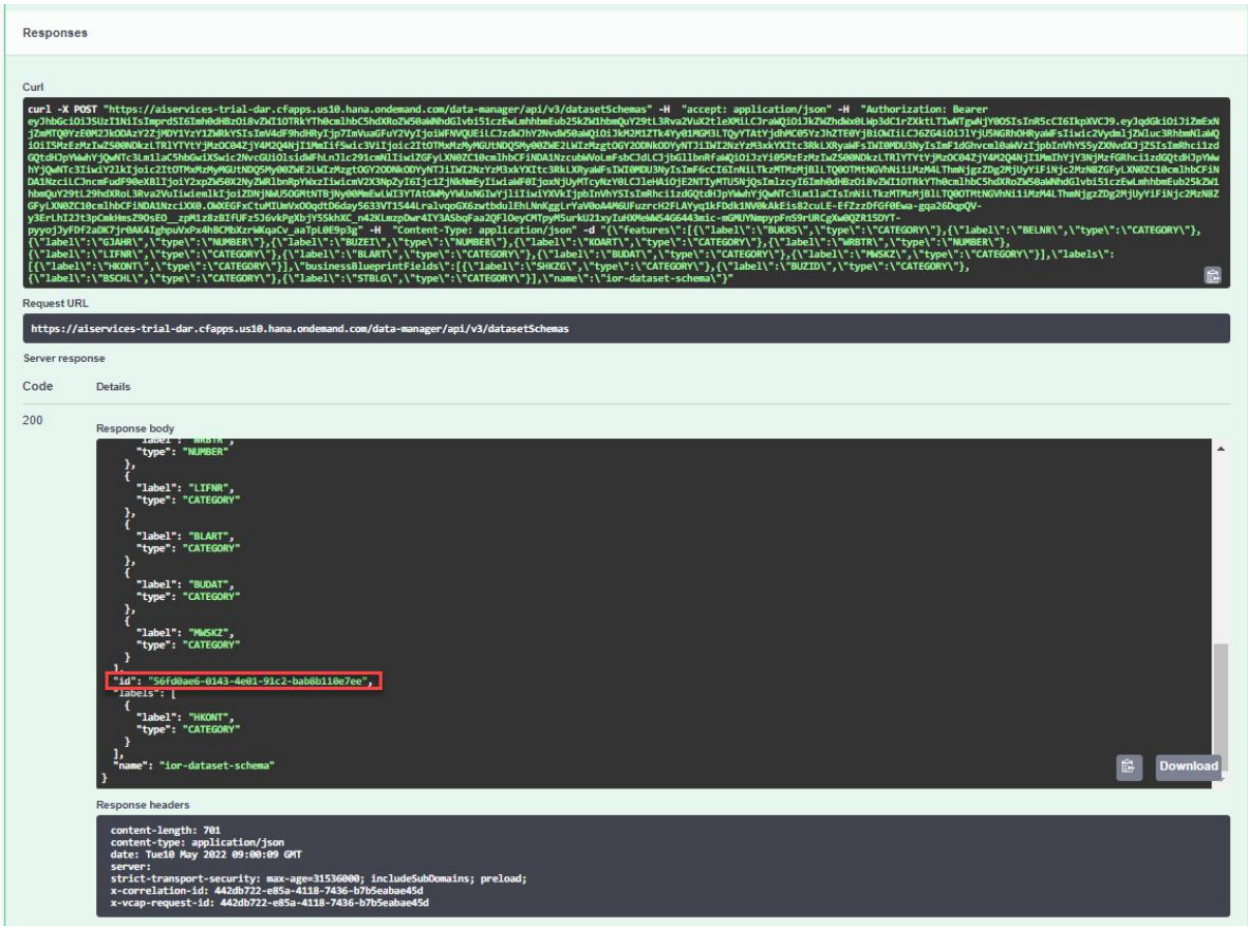
1. In Swagger UI, expand the endpoint `POST /datasetSchemas` by clicking on it. Then click **Try it out**.



2. Copy the above dataset schema into the text area. Then click **Execute** to create it.



3. Further below, you find the response of the service. The response includes a representation of dataset schema that was just created. Additionally, the dataset schema received an `id`. Copy it locally as you'll need it in the next step.



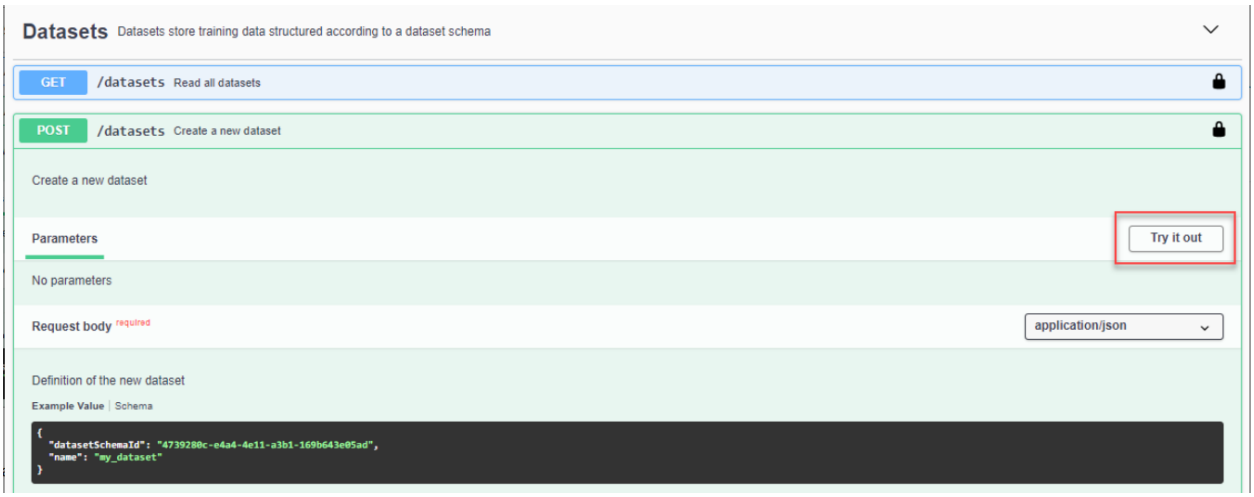
You have successfully created a dataset schema.

3.3

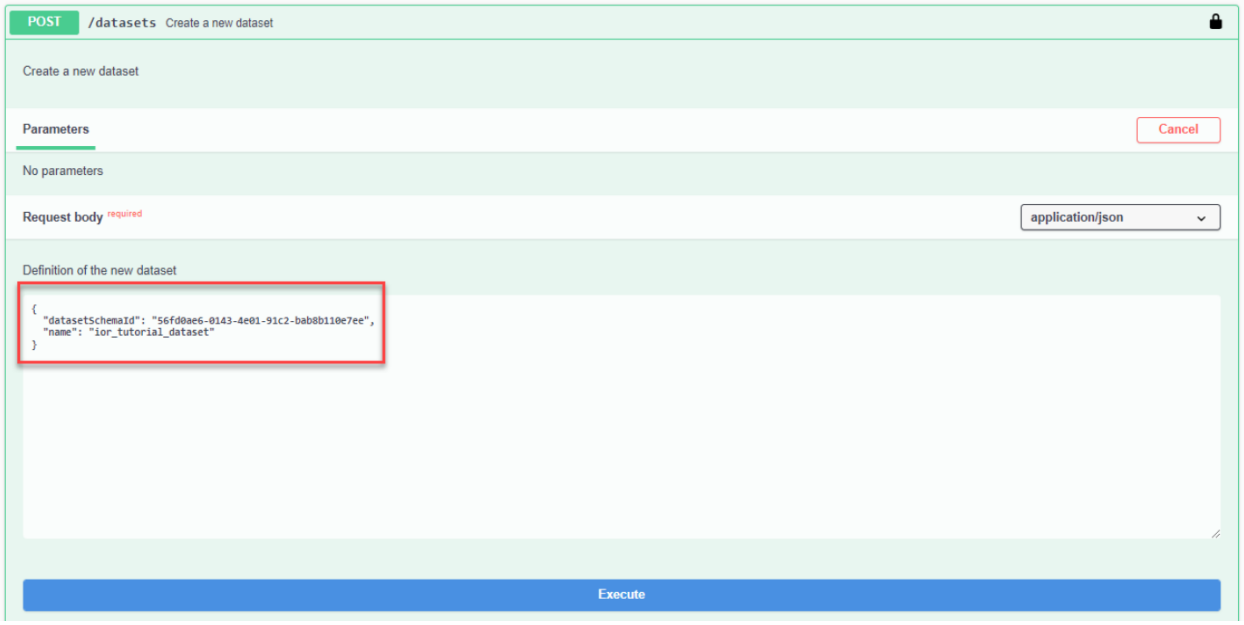
Create dataset

Next, you need to create a dataset using the dataset schema that you have created. The dataset is a table that holds the data that you'll upload later.
To create the dataset, proceed as follows:

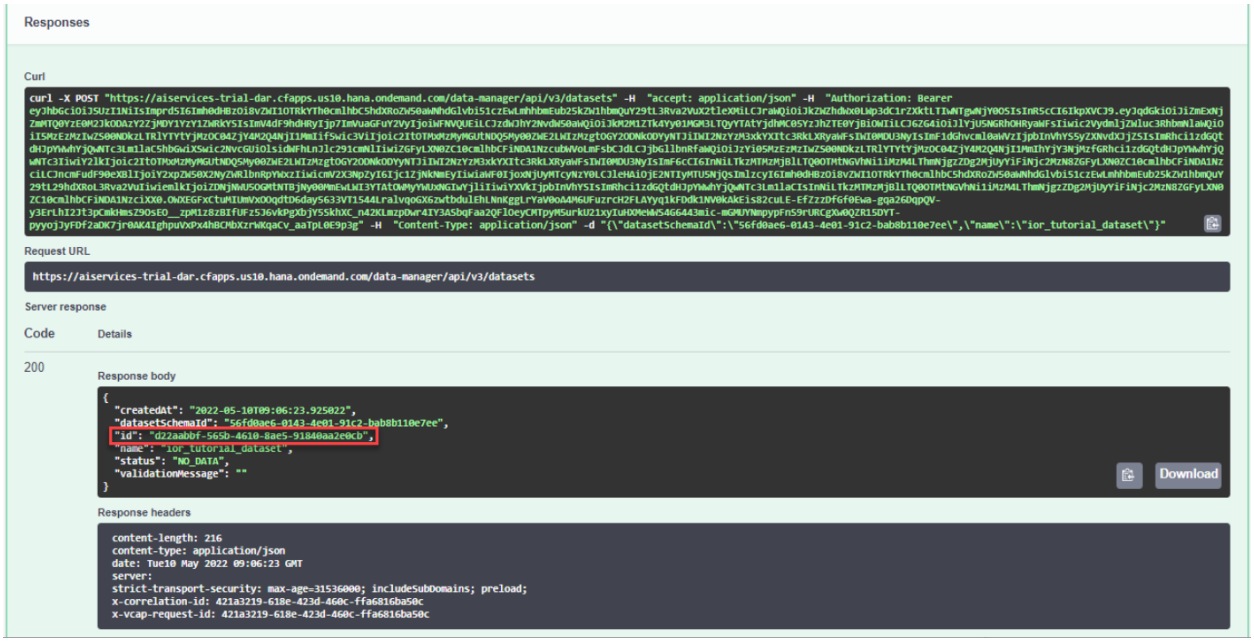
1. Expand the endpoint `POST /datasets` by clicking on it. Then click **Try it out**.



2. In the text area, replace the parameter `datasetSchemaId` with the `id` that you copied from the previous step and replace the parameter `name` with an appropriate name for you dataset, `ior_tutorial_dataset`, for example. Then click **Execute** to create the dataset.



3. In the response of the service, you find the `id` of your dataset. Copy it locally as you'll need it in the next steps. Additionally, you find the `status` of the dataset. The status is `NO_DATA` as no data file has been uploaded yet.

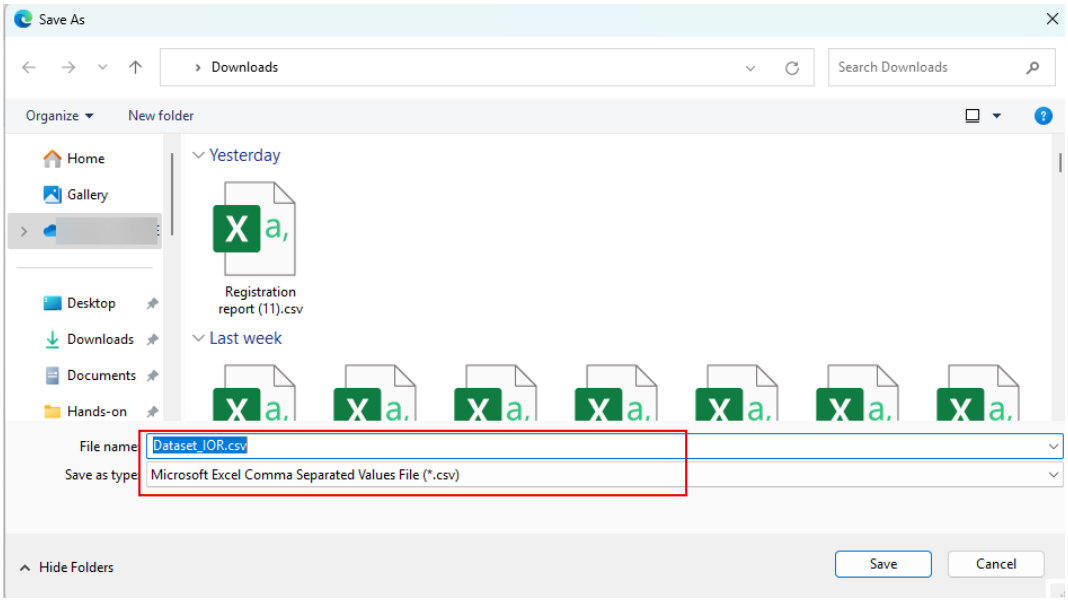


You have successfully created a dataset.

3.4

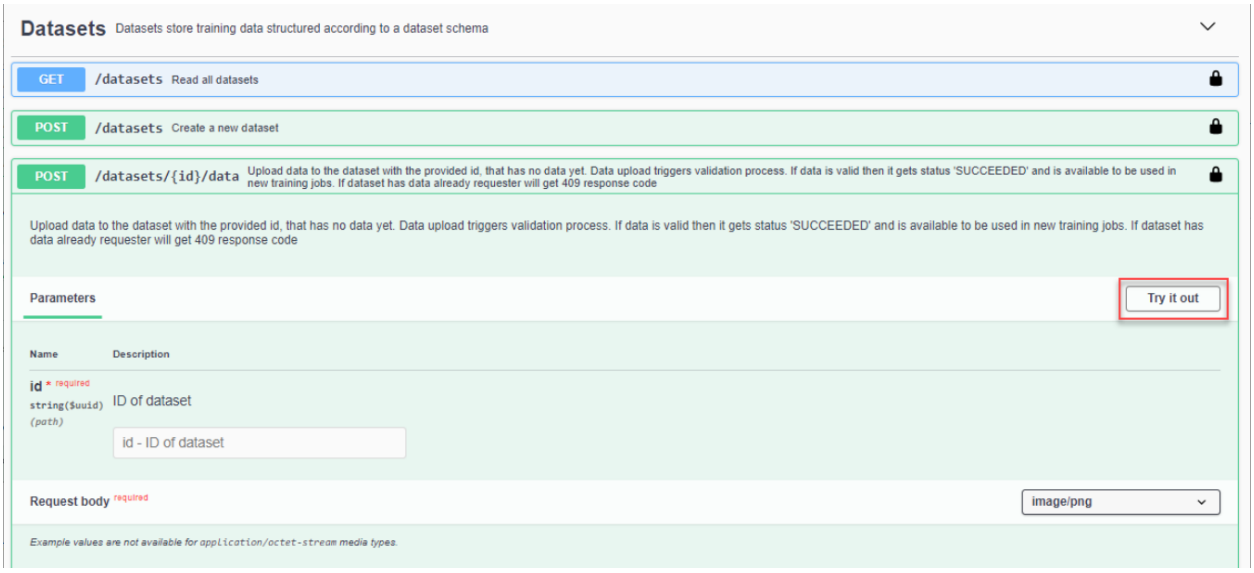
Upload data

The final step of this tutorial is to upload data to your dataset.
In this tutorial, you'll use this **dataset**. Right click on the link, then click ***Save link as*** to open the file dialog. Make sure the file is saved as `csv`.

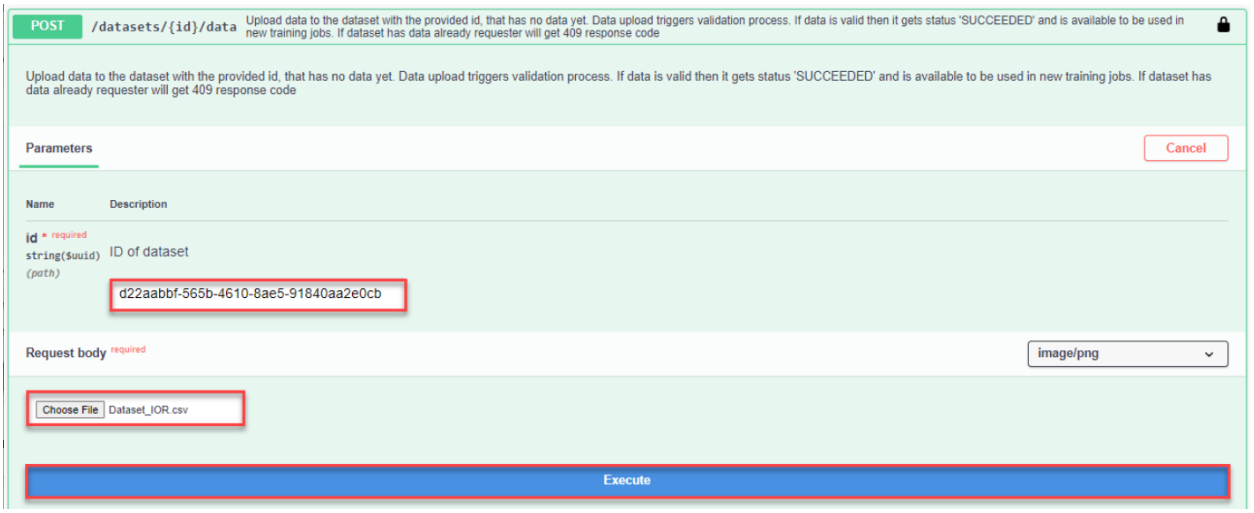


In Swagger UI, proceed as follows to upload to the data:

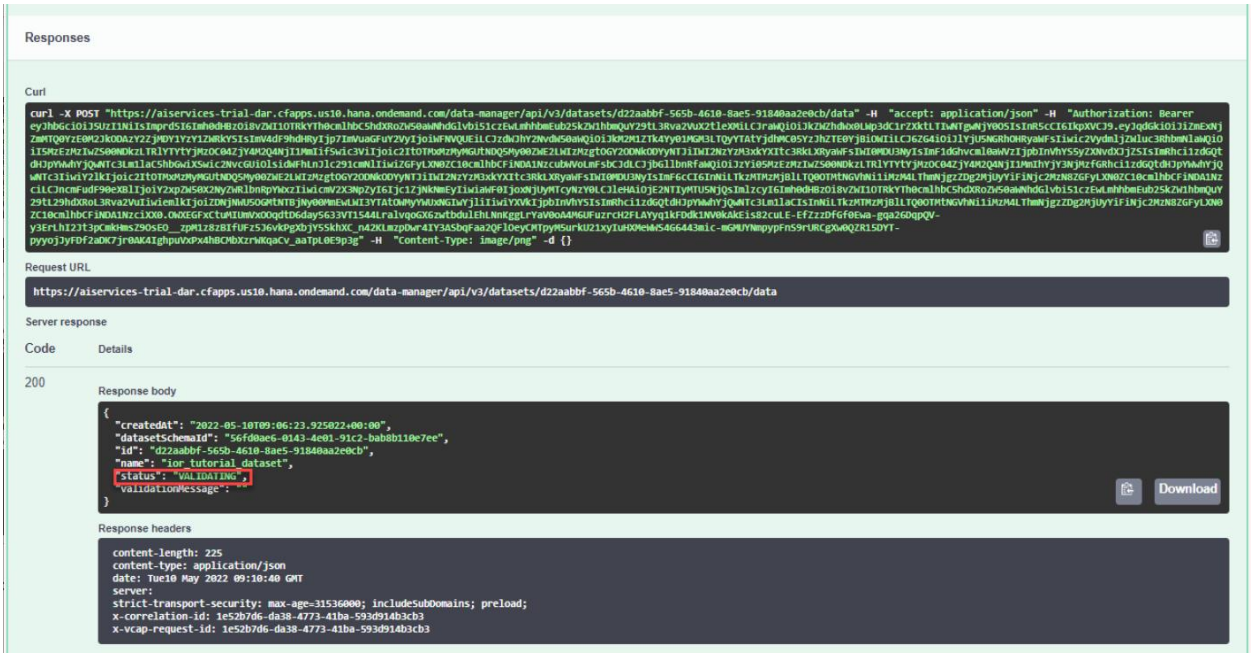
1. Expand the endpoint `POST /datasets/{id}/data` by clicking on it. Then click **Try it out**.



1. Fill the parameter `id` with the `id` of your dataset that you previously copied.
2. Click **Choose File** below the parameter `Request body`. In the dialog that opens, select the IOR dataset that you just downloaded. Then click **Execute** to upload the data.



In the response, you'll see that the status of your dataset has changed to `VALIDATING`. The service is now validating the data that you have uploaded.



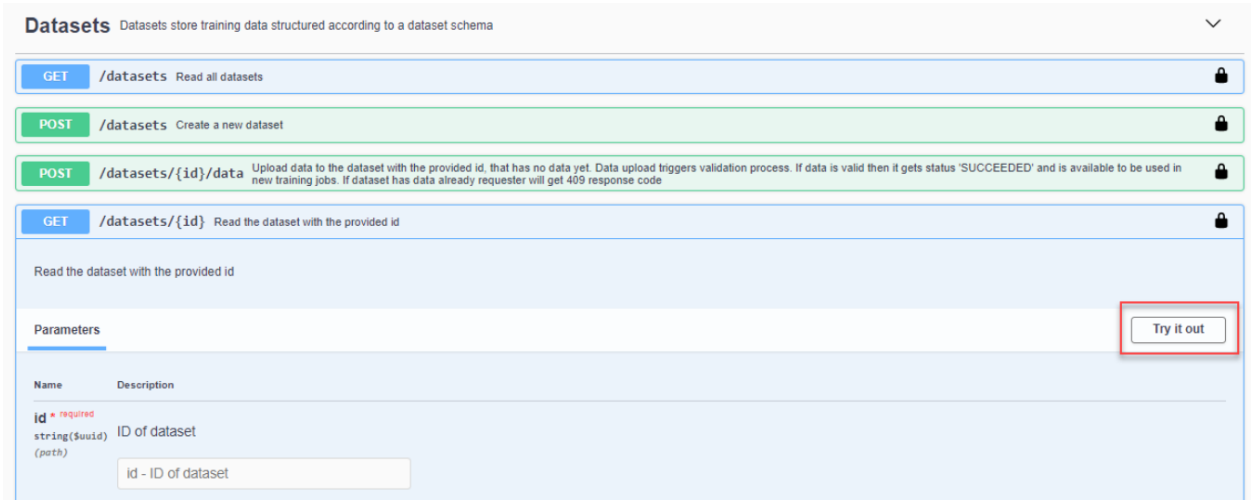
You have successfully uploaded data to your dataset.

3.5

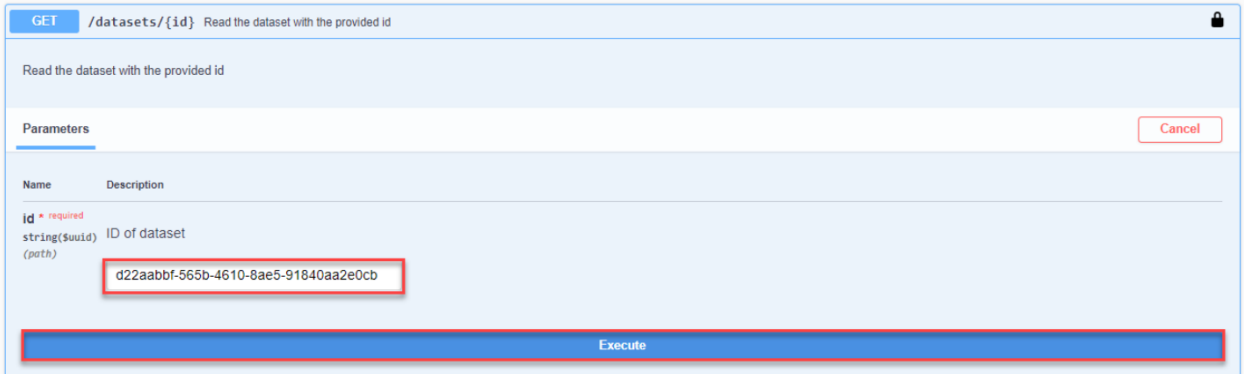
Check dataset status

To check the validation status of your data, proceed as follows:

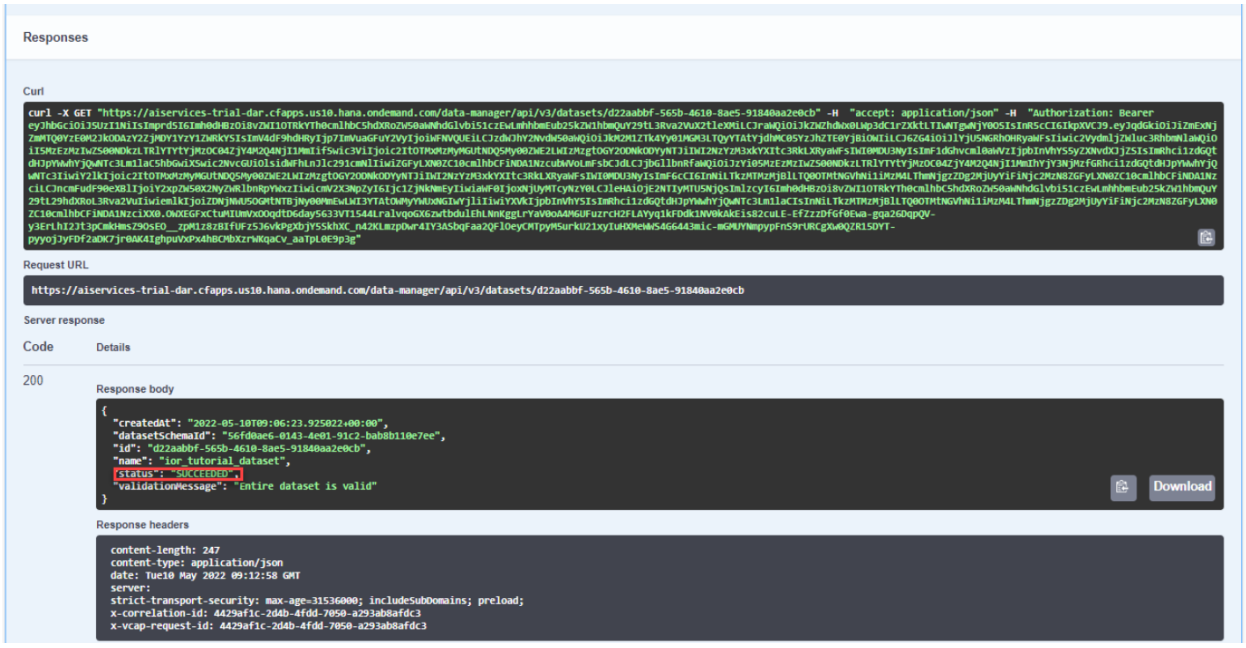
1. Expand the endpoint `GET /datasets/{id}` by clicking on it. Then click **Try it out**.



Fill the parameter `id` with the `id` of your dataset. Click **Execute**.



In the response of the service, you find the status of your dataset. If the status is still `VALIDATING`, check back in a few minutes. If the status is `SUCCEEDED`, your data is valid. In case the status is either `INVALID_DATA` or `VALIDATION_FAILED`, create a new dataset and upload the data once again.



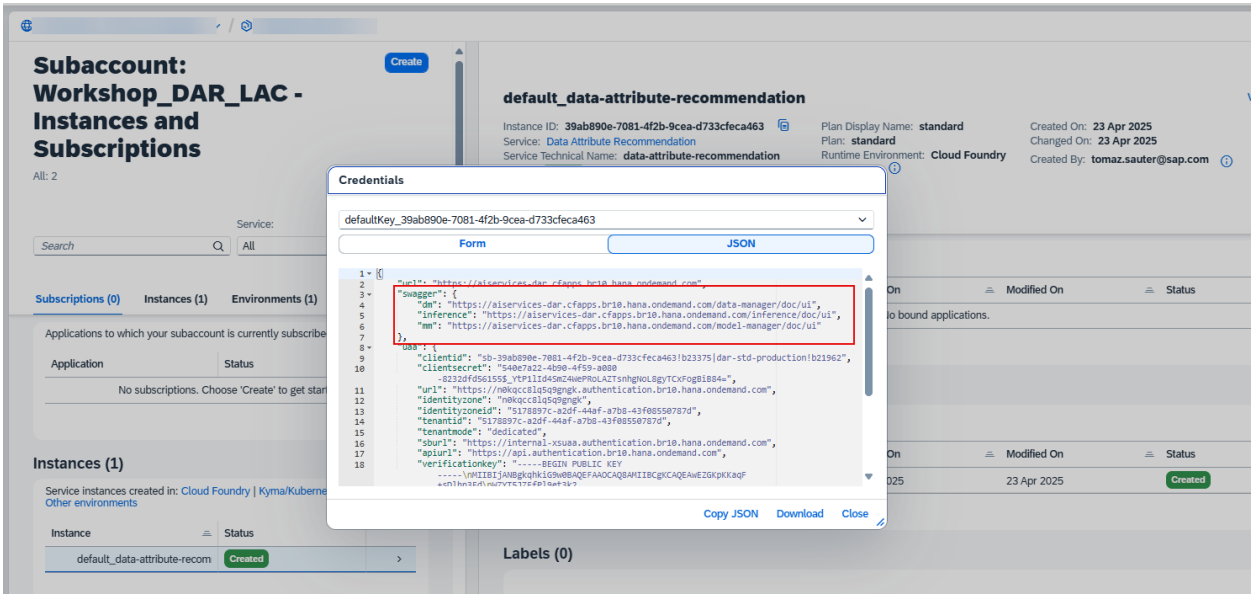
You have successfully created a dataset and uploaded data. You can now use the dataset to train a machine learning model.

Step 4: Use the Invoice Object Recommendation (IOR) Business Blueprint to Train a Machine Learning Model

4.1

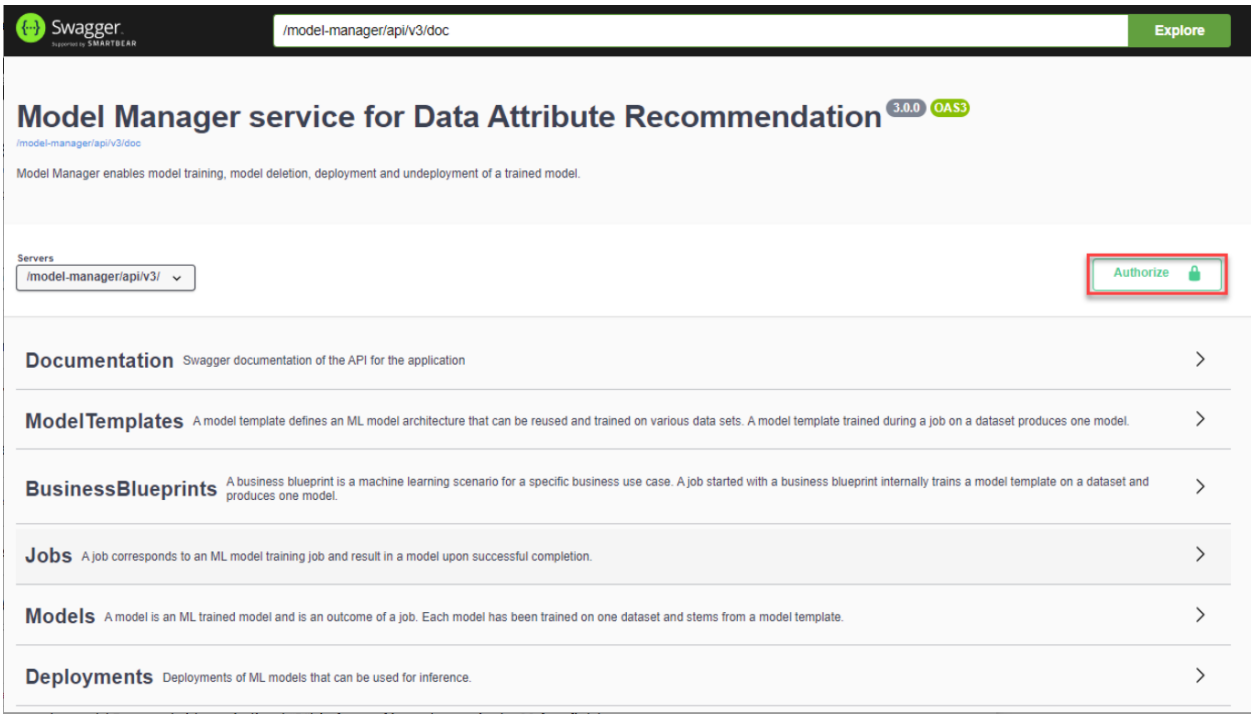
Authorize Swagger UI

In the service key you created for Data Attribute Recommendation in the previous step you will find a section called swagger (as highlighted in the image below) with three entries, called dm (data manager), mm (model manager) and inference. You'll use all three Swagger UIs throughout the exercise.



For this tutorial, copy the URL of the Swagger UI for `mm` and open it in a browser tab. The Swagger UI for the model manager allows you to train a machine learning model, to delete it, to deploy the model as well as to `undeploy` the model.

1. To be able to use the Swagger UI endpoints, you need to authorize yourself. In the top right corner, click **Authorize**.



2. Get the `access_token` value created in the previous step and add **Bearer** (with capitalized "B") in front of it, and enter in the **Value** field.

```
Bearer <access_token>
```

3. Click **Authorize** and then click **Close**.

4.2

Create a training job

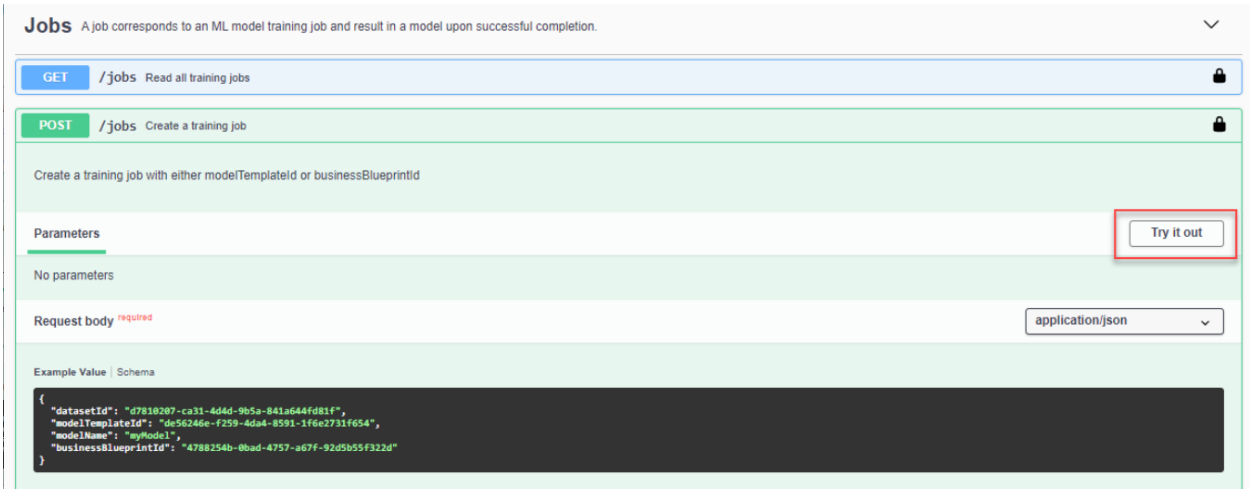
To train a machine learning model using the data that you uploaded in the previous step, you create a training job.

With each training job you provide a model template or a business blueprint which combines data processing rules and machine learning model architecture. You can find the list of available model templates **here**, and the list of available business blueprints **here**.

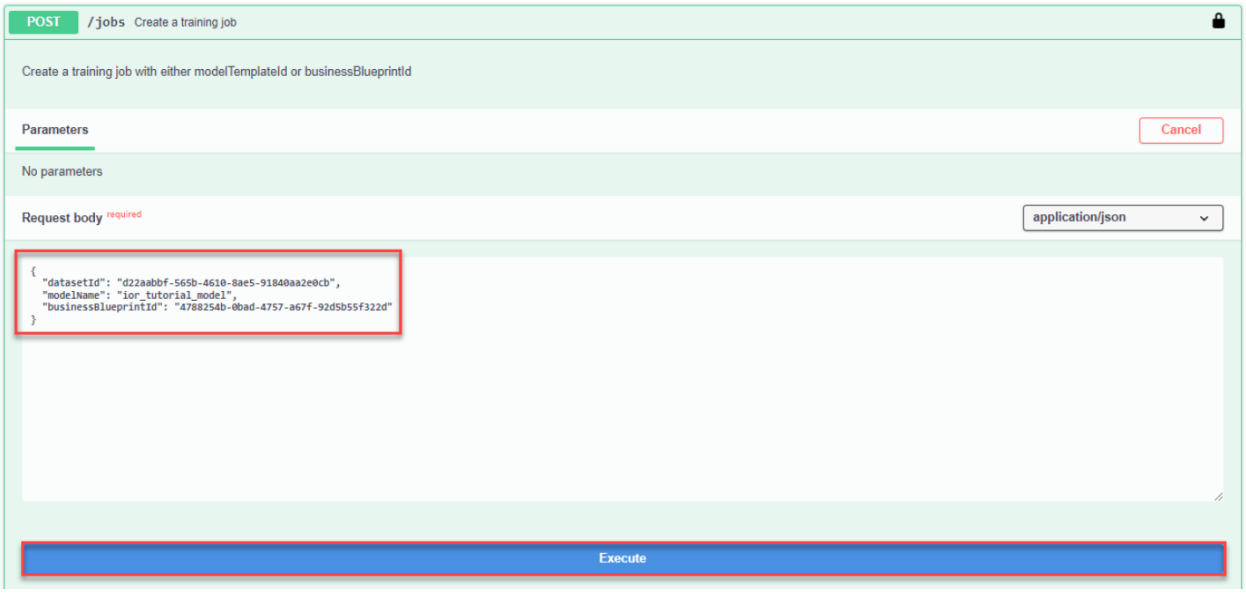
The Invoice Object Recommendation (IOR) business blueprint that you use in this tutorial is suited to assign G/L (general ledger) accounts and other financial objects to incoming invoices without a purchase order reference.

To create the training job, proceed as follows:

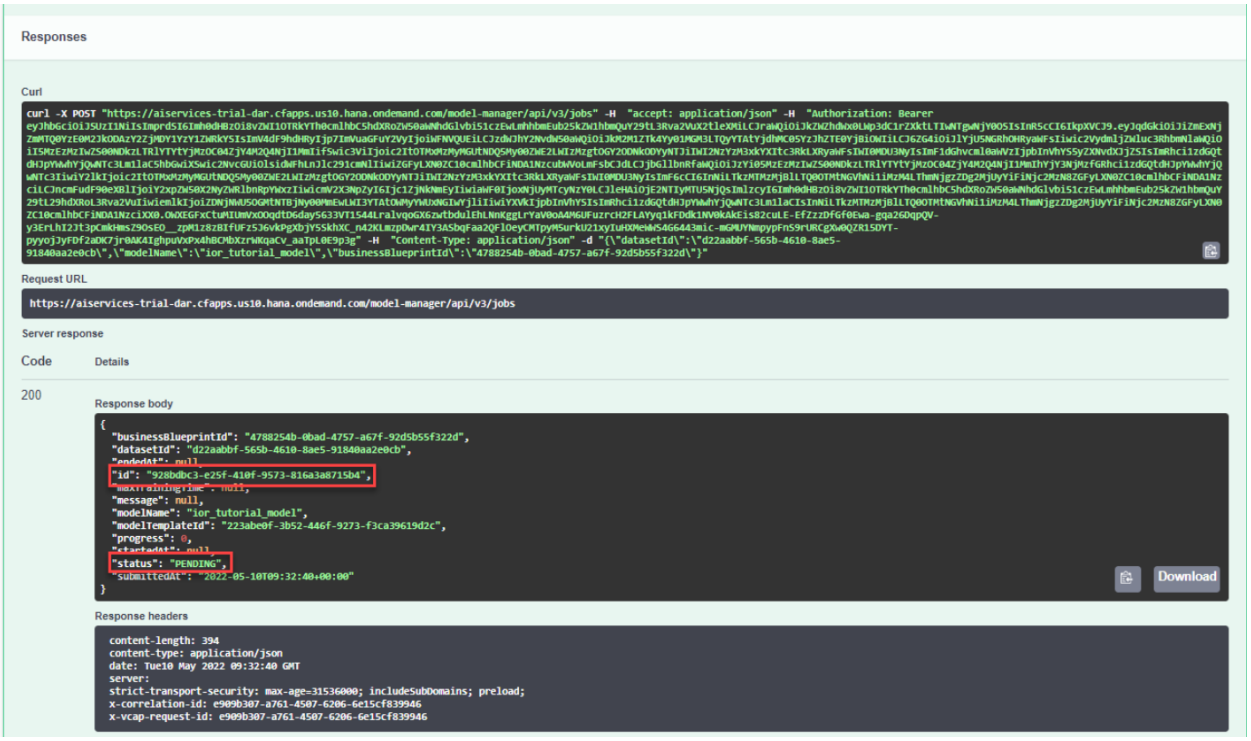
1. Expand the endpoint **POST /jobs** by clicking on it. Then click **Try it out**.



2. In the text area, replace the parameter value for `datasetId` with the `id` of your dataset that you have created previously. Delete the `modelTemplateId` line from the **Request body**. Replace the parameter value `modelName` with your model name, `ior_tutorial_model`, for example. Make sure the parameter value for `businessBlueprintId` is `4788254b-0bad-4757-a67f-92d5b55f322d`. Click **Execute** to create the training job.



3. In the response of the service, you find the `id` of your training job. Copy the `id` as you'll need it in the next step. Along side the `id`, you find the training job's current status. Initially, the status is `PENDING` which says that the training job is in queue but has not started yet.



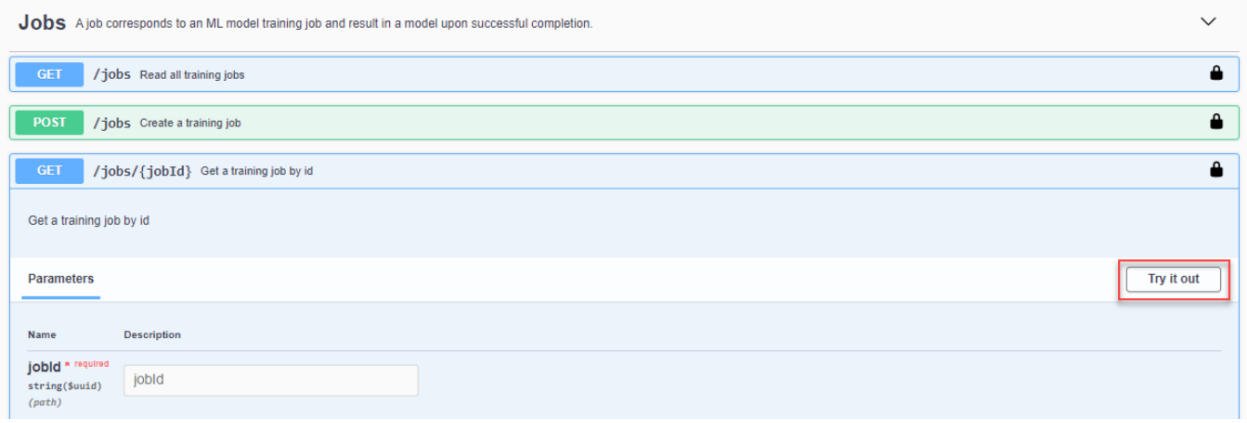
You have successfully created a training job.

4.3

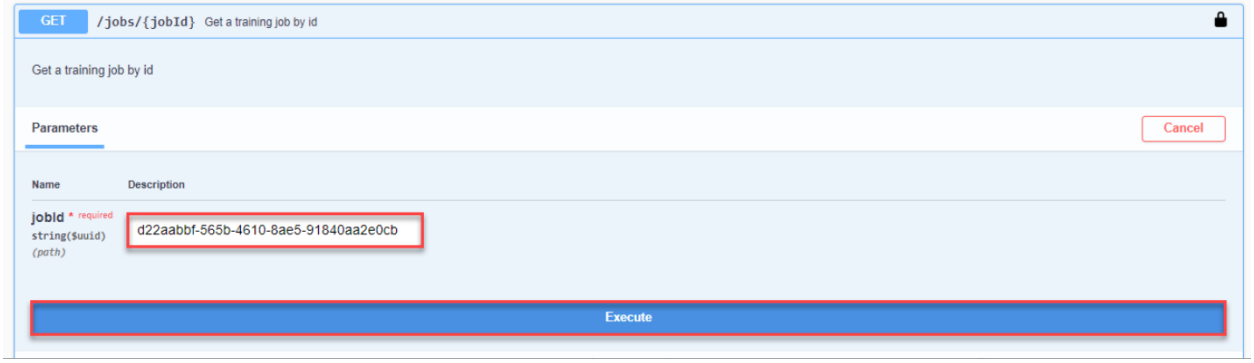
Check training job status

To know when your training job has ended, you have to frequently check its status. For that, proceed as follows:

1. Expand the endpoint `GET /jobs/{jobId}` by clicking on it. Then click **Try it out**.

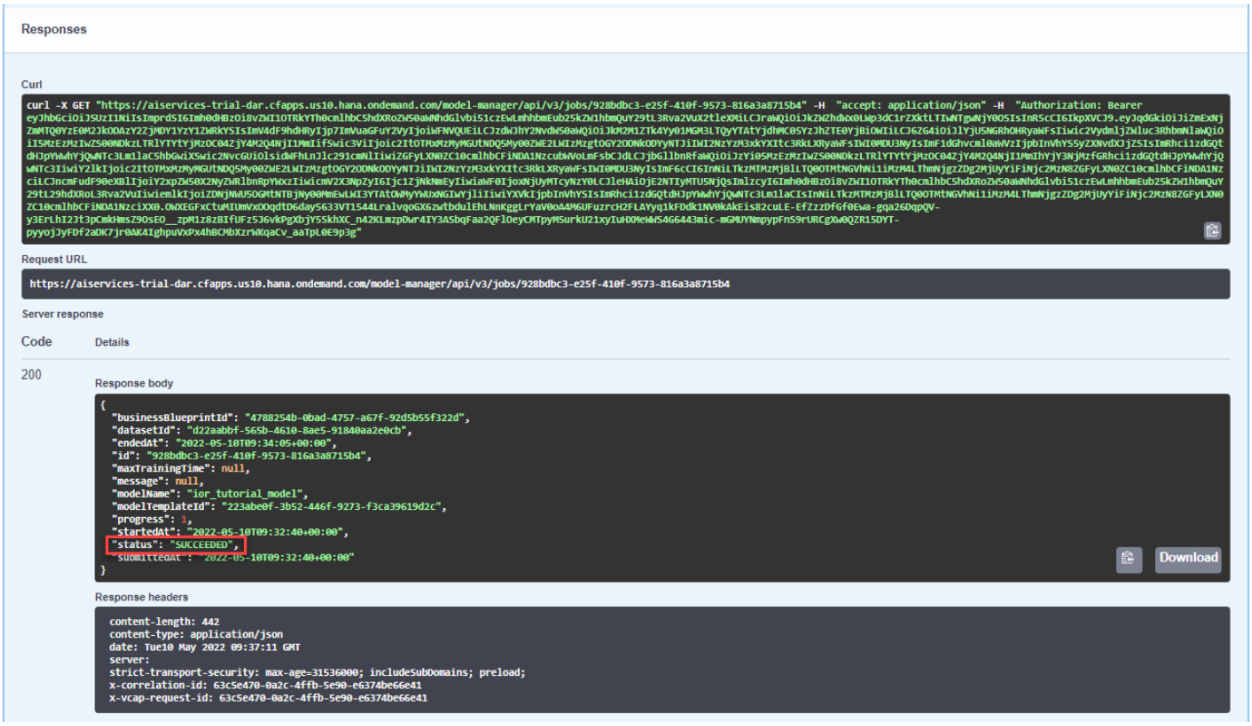


2. Fill the parameter `jobId` with `id` of your training job that you copied in the previous step. Click **Execute**.



3. In the response, you find again the current status of your training job along with other details. Immediately after creation of the training job, the status is `PENDING`. Shortly after, it changes to `RUNNING` which means that the model is being trained. The training of the sample data usually takes about 5 minutes to complete but may run longer, up to a few hours due to limited availability of resources in the free tier environment. You can check the status every now and

then. Once training is finished, the status changes to `SUCCEEDED` which means the service has created a machine learning model and you can proceed.



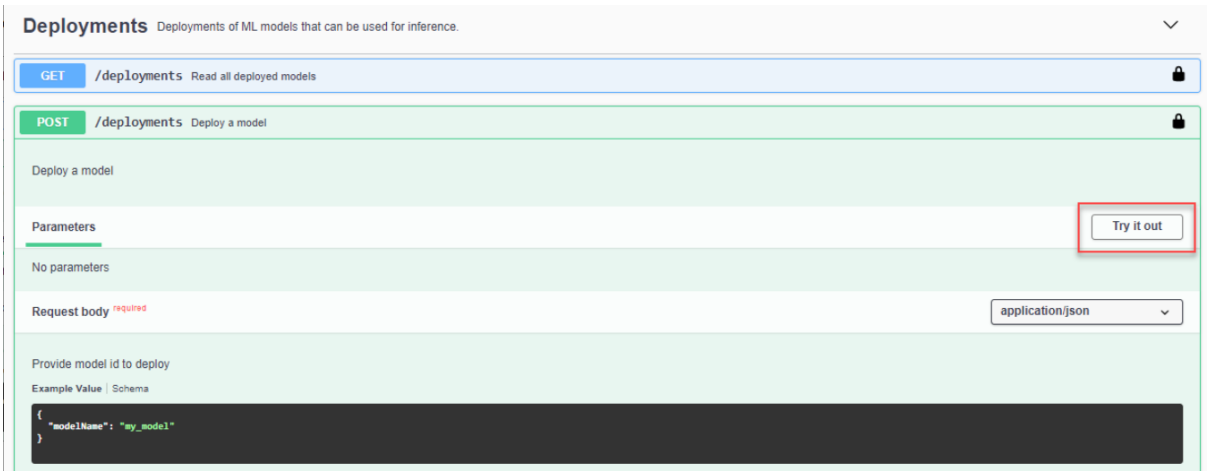
You have successfully trained a machine learning model.

4.4

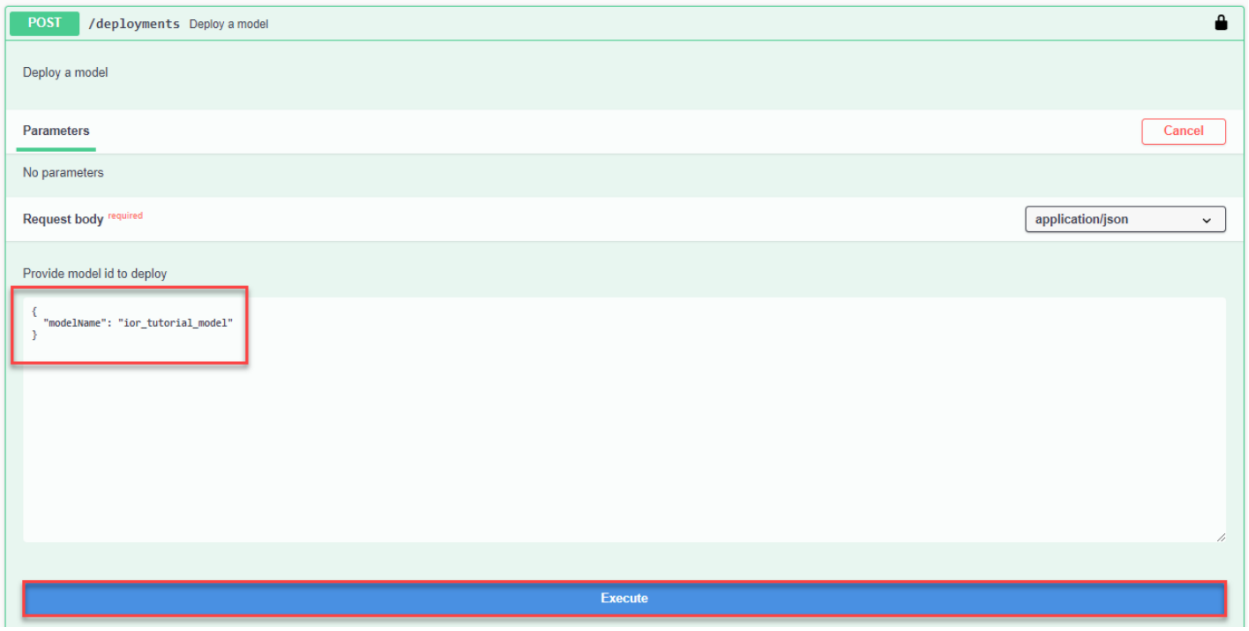
Deploy machine learning model

To use the trained model, you need to deploy it. Once deployed, the model is ready to make predictions. To deploy your model, proceed as follows:

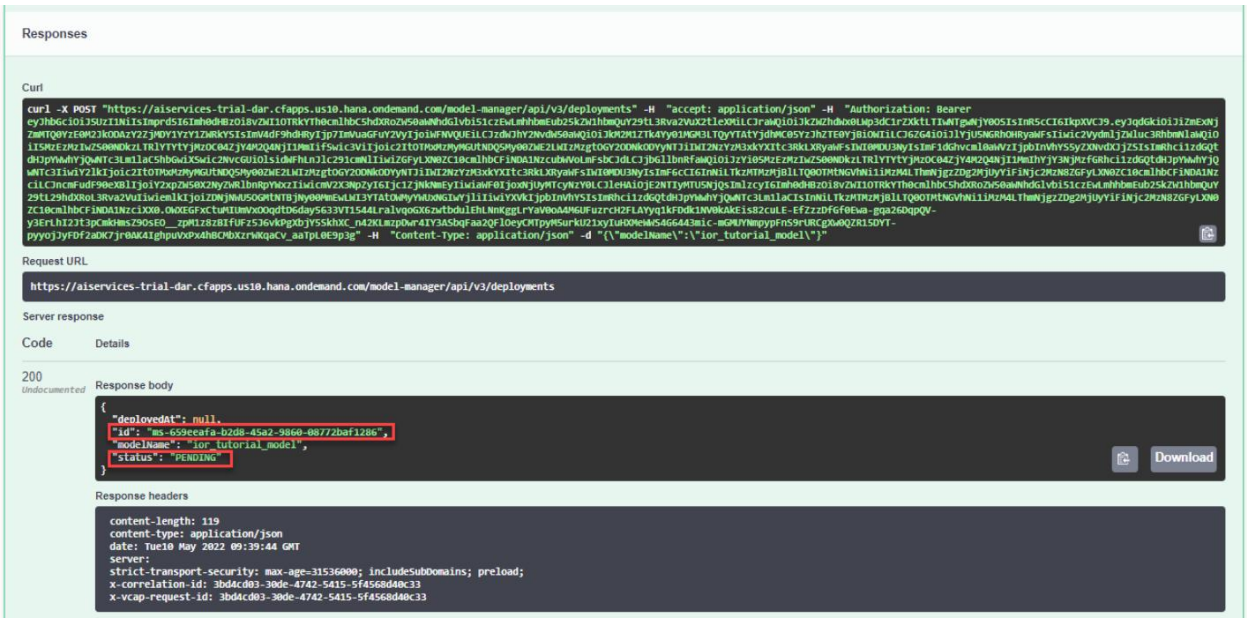
1. Expand the endpoint `POST /deployments` by clicking on it. Then click **Try it out**.



2. In the text area, replace the parameter `modelName` with the name of your model (`ior_tutorial_model`). Click **Execute** to deploy the model.



3. In the response of the service, you find the `id` of the deployment and its status. Initially, the status is `PENDING`, indicating the deployment is in progress. Make sure to copy the `id` as you need it in the next step.

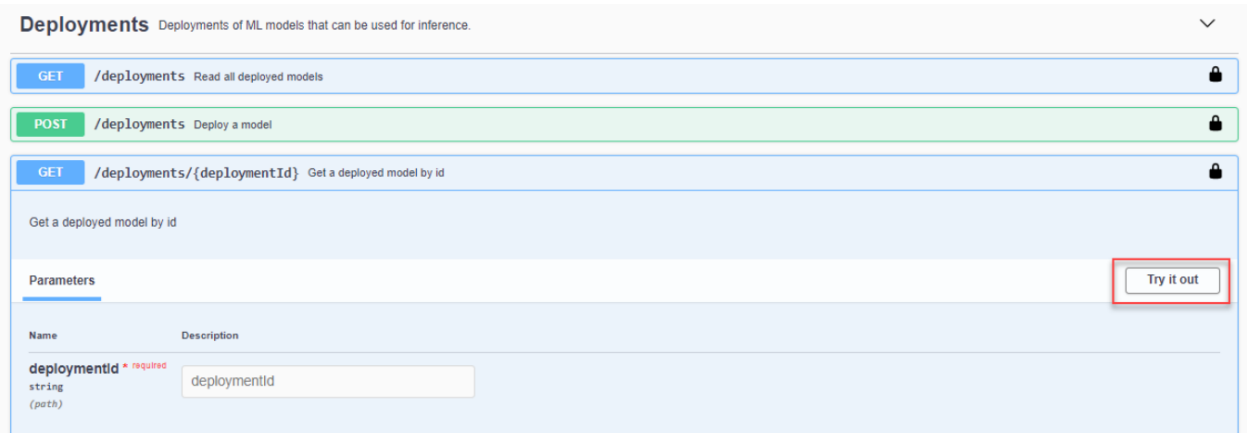


4.5

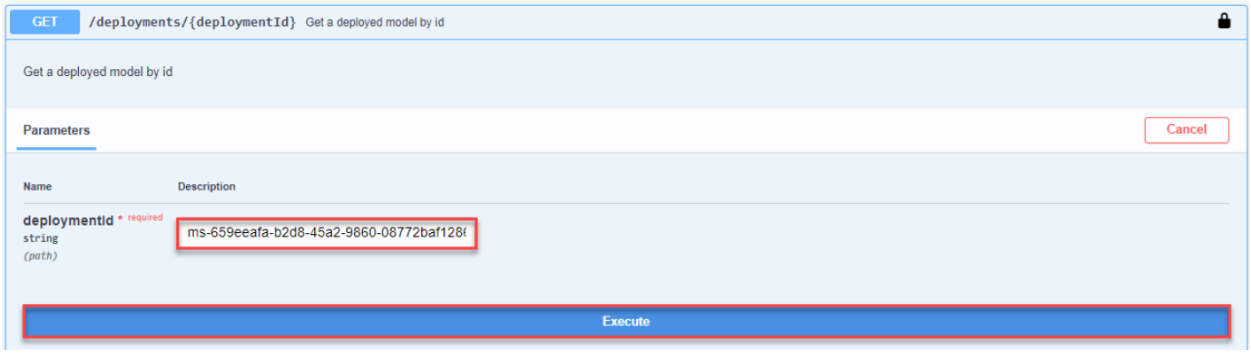
Check deployment status

Finally, you have to ensure that your model is deployed successfully in order to use it for predictions. To check the status of your deployment, proceed as follows:

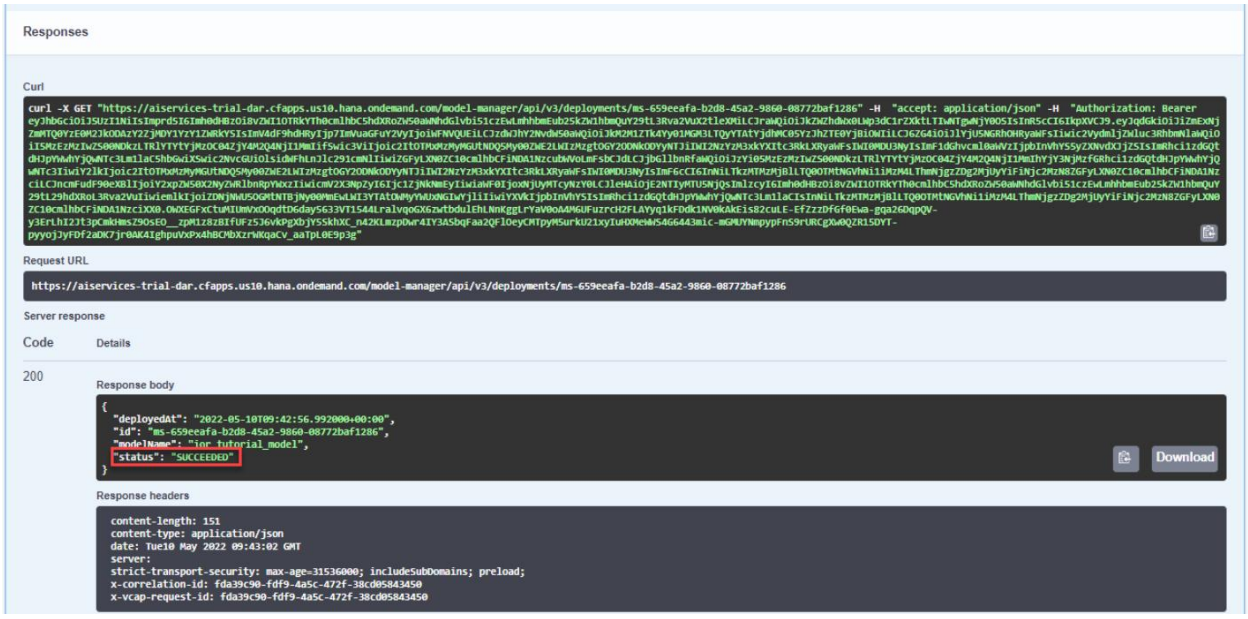
1. Expand the endpoint `GET /deployments/{id}` by clicking on it. Then click **Try it out**.



2. Fill the parameter `deploymentId` with the `id` of your deployment that you copied in the previous step. Click **Execute**.



3. In the response of the service, you find the current status of the deployment. If the status is `SUCCEEDED`, your deployment is done. If the status is still `PENDING`, check back in a few minutes.



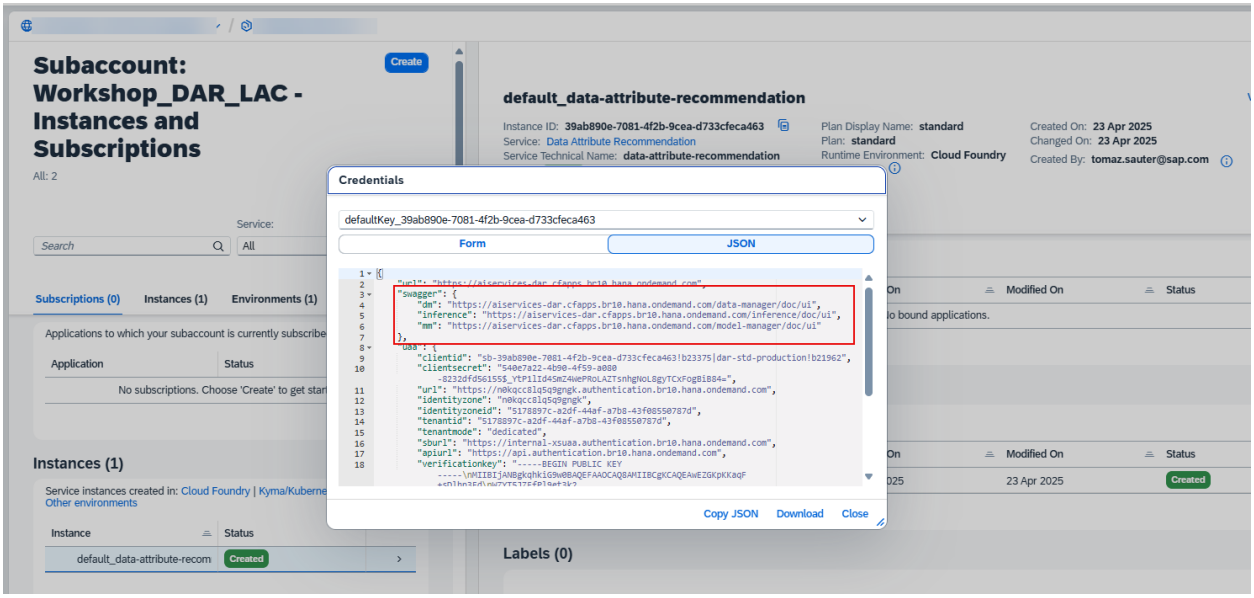
You have successfully trained a machine learning model and deployed it. Next, you'll use your model to make predictions.

Step 5: Use the Invoice Object Recommendation (IOR) Business Blueprint to Predict Financial Objects

5.1

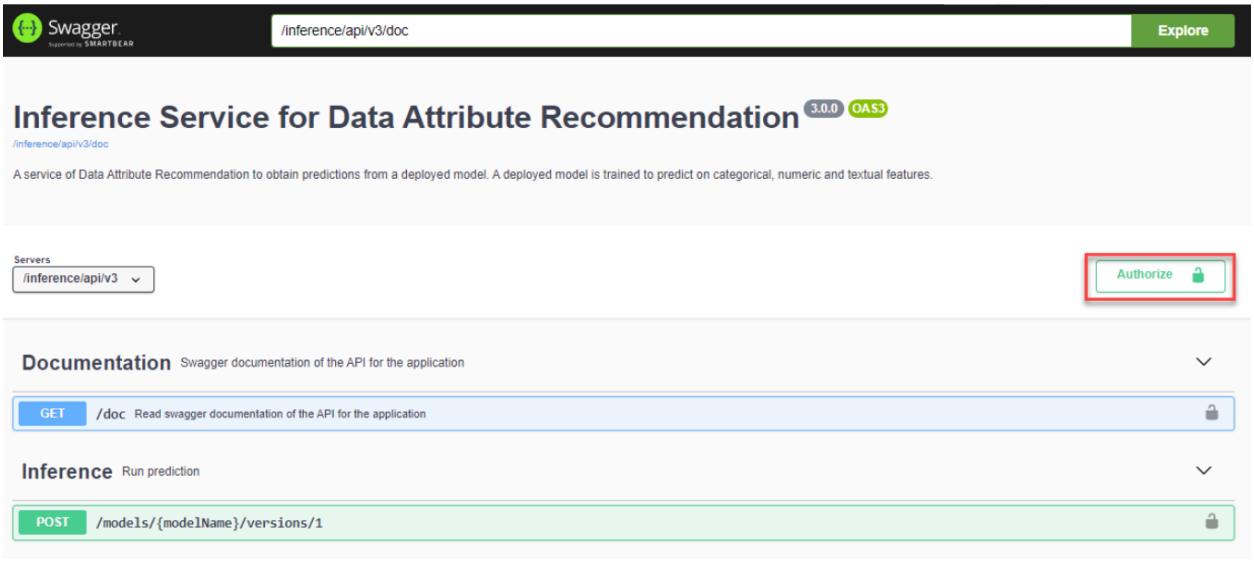
Authorize Swagger UI

In the service key you created for Data Attribute Recommendation in the previous step you will find a section called swagger (as highlighted in the image below) with three entries, called dm (data manager), mm (model manager) and inference. You'll use all three Swagger UIs throughout the exercise.



For the following step, copy the URL of the Swagger UI for `inference` and open it in a browser tab. The Swagger UI for inference allows you to classify new data using your machine learning model that you have created previously.

1. To be able to use the Swagger UI endpoints, you need to authorize yourself. In the top right corner, click **Authorize**.



2. Get the `access_token` value created in the previous step then add **Bearer** (with capitalized "B") in front of it, and enter in the **Value** field.

```
Bearer <access_token>
```

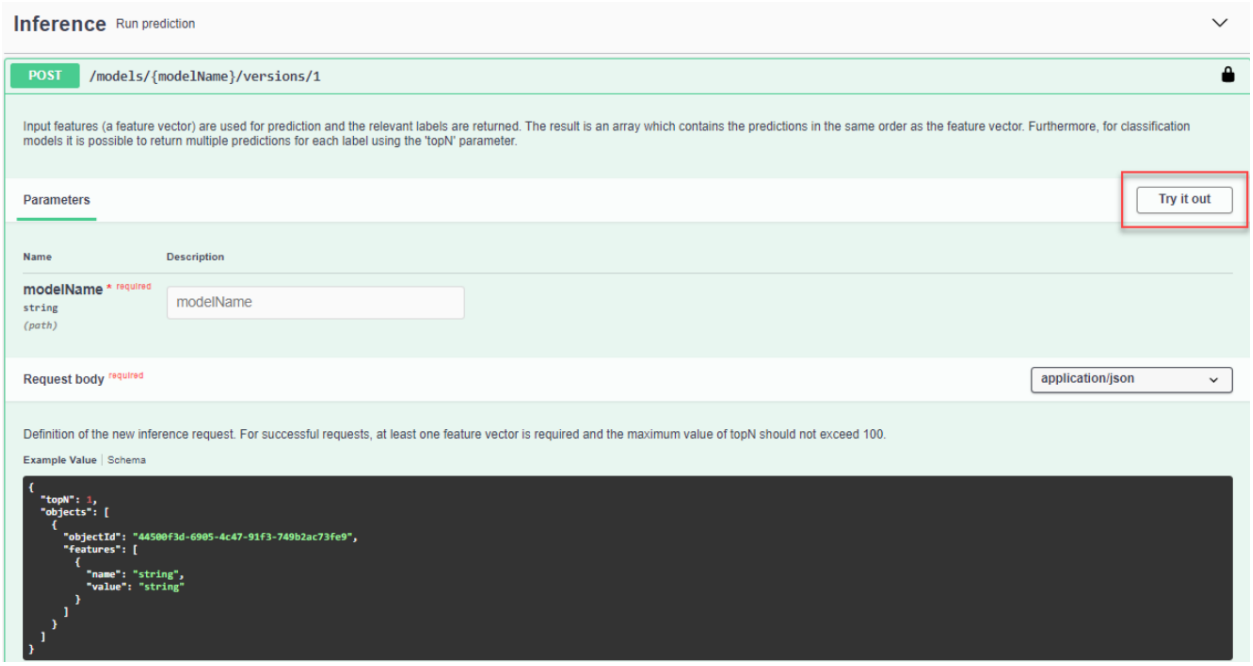
3. Click **Authorize** and then click **Close**.

5.2

Predict financial objects

To get the machine learning model predictions, proceed as follows:

1. Expand the endpoint `POST /models/{modelName}/versions/1` by clicking on it. Then click **Try it out**.



2. Fill the parameter `modelName` with the name of your machine learning model (`ior_tutorial_model`).

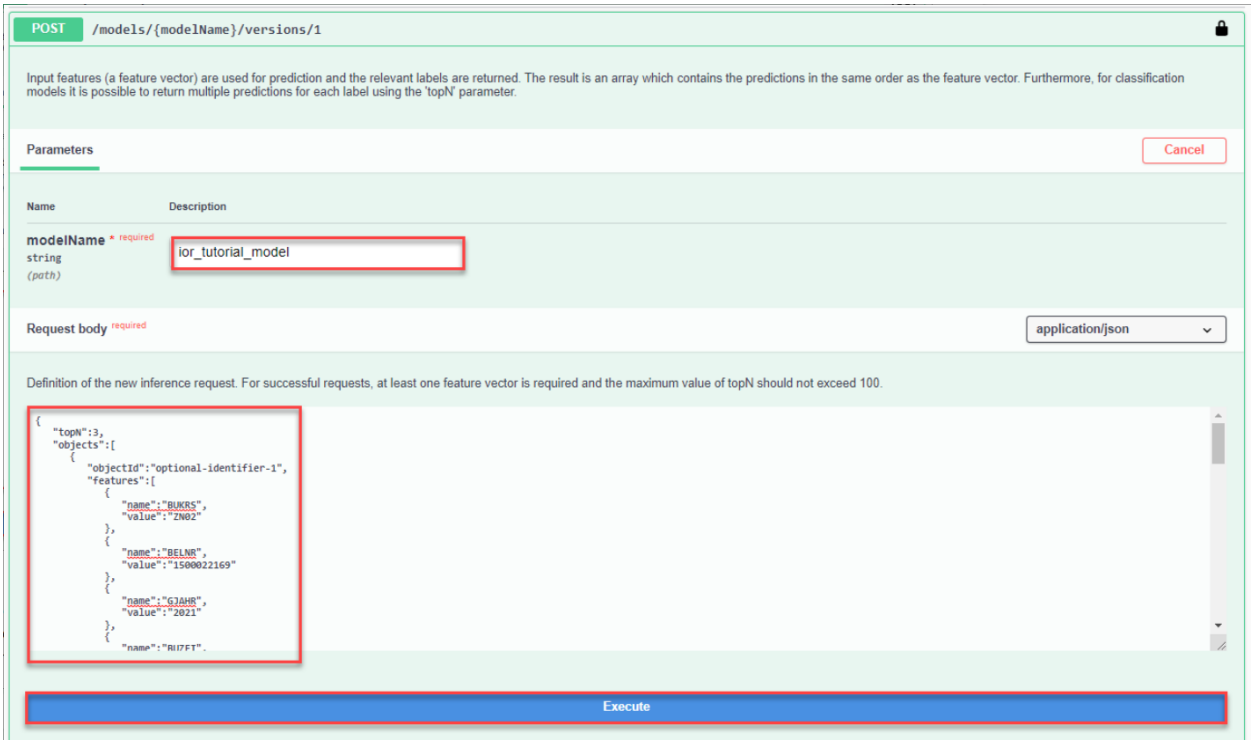
3. In the parameter `body`, you have to provide the data that needs to be predicted. According to the dataset schema that you have created in step 3, the machine learning model takes the training fields such as BUKRS for company code as input and predicts G/L accounts (HKONT). Replace the text in the parameter `body` with the following:

```
{
  "topN":3,
  "objects":[
    {
      "objectId":"optional-identifier-1",
      "features":[
        {
          "name":"BUKRS",
          "value":"ZN02"
        },
        {
          "name":"BELNR",
          "value":"1500022169"
        },
        {
          "name":"GJAHR",
          "value":"2021"
        },
        {
```

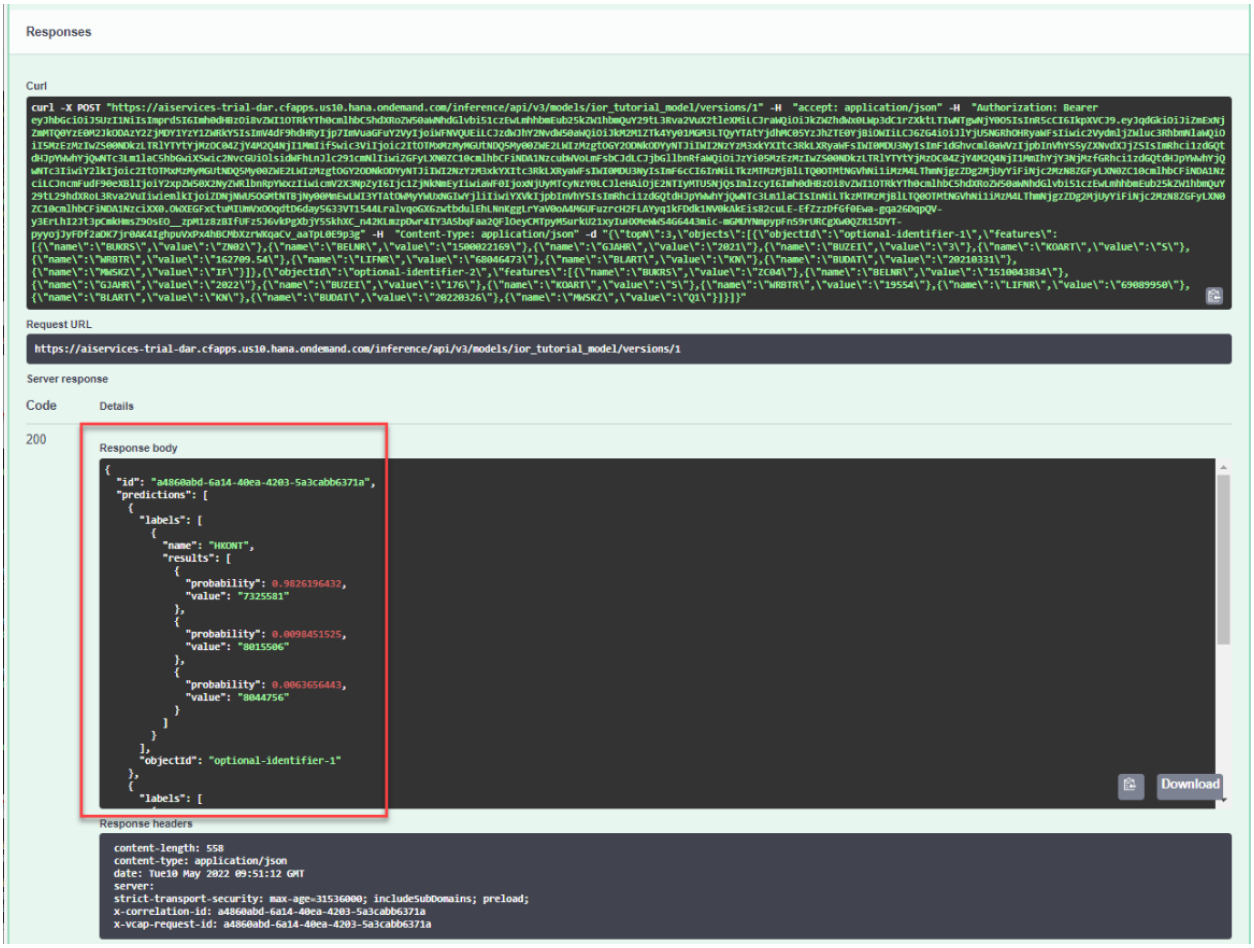
```
      "name":"BUZEI",
      "value":"3"
    },
    {
      "name":"KOART",
      "value":"S"
    },
    {
      "name":"WRBTR",
      "value":"162709.54"
    },
    {
      "name":"LIFNR",
      "value":"68046473"
    },
    {
      "name":"BLART",
      "value":"KN"
    },
    {
      "name":"BUDAT",
      "value":"20210331"
    },
    {
      "name":"MWSKZ",
      "value":"IF"
    }
  ]
},
{
  "objectId":"optional-identifier-2",
  "features":[
    {
      "name":"BUKRS",
      "value":"ZC04"
    },
    {
      "name":"BELNR",
      "value":"1510043834"
    },
    {
      "name":"GJAHR",
      "value":"2022"
    },
    {
      "name":"BUZEI",
      "value":"176"
    },
    {
      "name":"KOART",
      "value":"S"
    },
    {
      "name":"WRBTR",
      "value":"19554"
    },
    {
      "name":"LIFNR",
```

```
        "value":"69089950"
      },
      {
        "name":"BLART",
        "value":"KN"
      },
      {
        "name":"BUDAT",
        "value":"20220326"
      },
      {
        "name":"MWSKZ",
        "value":"Q1"
      }
    ]
  }
 ]
}
```

4. Click Execute to send the above input to the service to get financial object predictions.



In the response of the service, you find the probability and the values for G/L account numbers (HKONT). The probability represents how certain the model is about its prediction. The higher the probability the more confident the model is that the prediction is actually correct. If the probability is close to 1, the model is very certain. The service provides one main prediction and two alternative predictions for each G/L account.



{
"id": "a4860abd-6a14-40ea-4203-5a3cabb6371a",
"predictions": [
{
"labels": [
{
"name": "HKONT",
"results": [
{
"probability": 0.9826196432,
"value": "7325581"
},
{
"probability": 0.0098451525,
"value": "8015506"
},
{
"probability": 0.0063656443,
"value": "8044756"
}
]
}

```
    ],
    "objectId": "optional-identifier-1"
   },
   {
    "labels": [
     {
      "name": "HKONT",
      "results": [
       {
        "probability": 0.9783372283,
        "value": "7515556"
       },
       {
        "probability": 0.018033715,
        "value": "7325581"
       },
       {
        "probability": 0.0034737068,
        "value": "8044756"
       }
      ]
     }
    ],
    "objectId": "optional-identifier-2"
   }
  ],
  "processedTime": "2022-05-10T09:51:12.253115",
  "status": "DONE"
}
```


You have successfully used a machine learning model to predict financial objects. Feel free to adapt the examples above and retry the prediction.

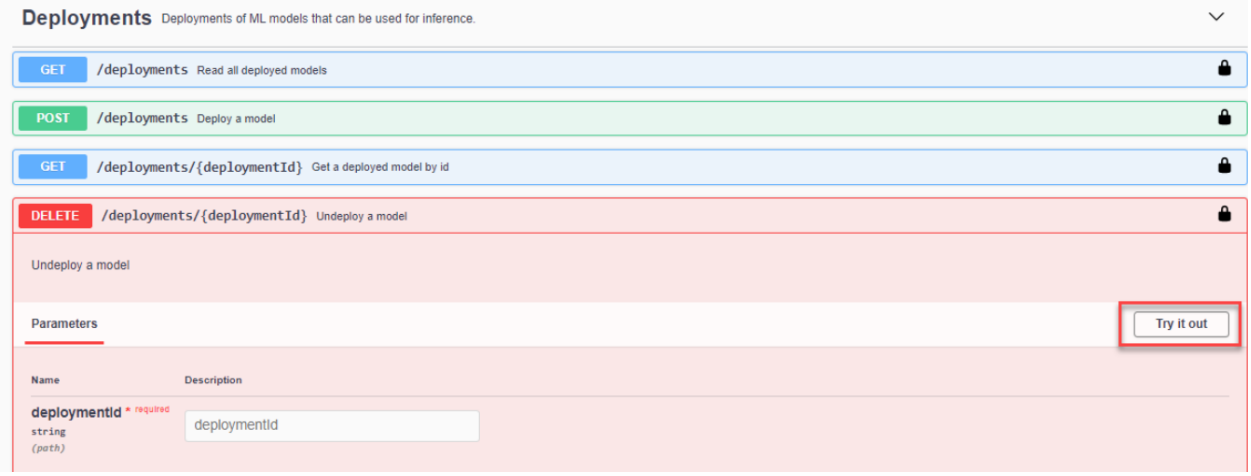
5.3

Undeploy your model (Optional)

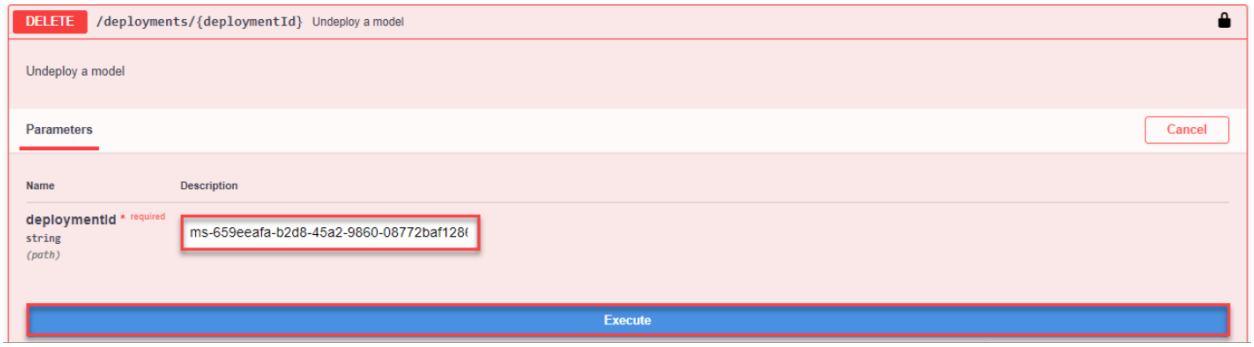
Now that you have learned the whole process about how to use the Invoice Object Recommendation (IOR) Business Blueprint from the Data Attribute Recommendation service, it's time to clean up.

First, `undeploy` your model. For that, go back to the Swagger UI for `mm` and:

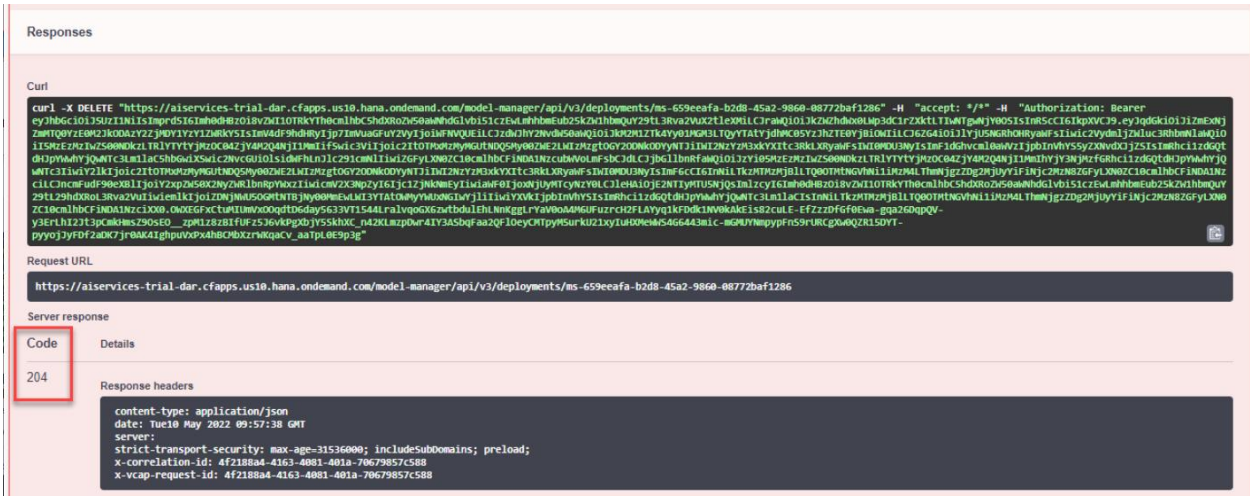
1. Expand the endpoint `DELETE /deployments/{deploymentId}` by clicking on it. Then click **Try it out**.



2. Fill the parameter `deploymentId` with the ID of your deployment. Use the `GET /deployments` endpoint in case you no longer have the deployment `id` in hand.



3. If the response code is `204`, the model has been successfully `undeployed`.



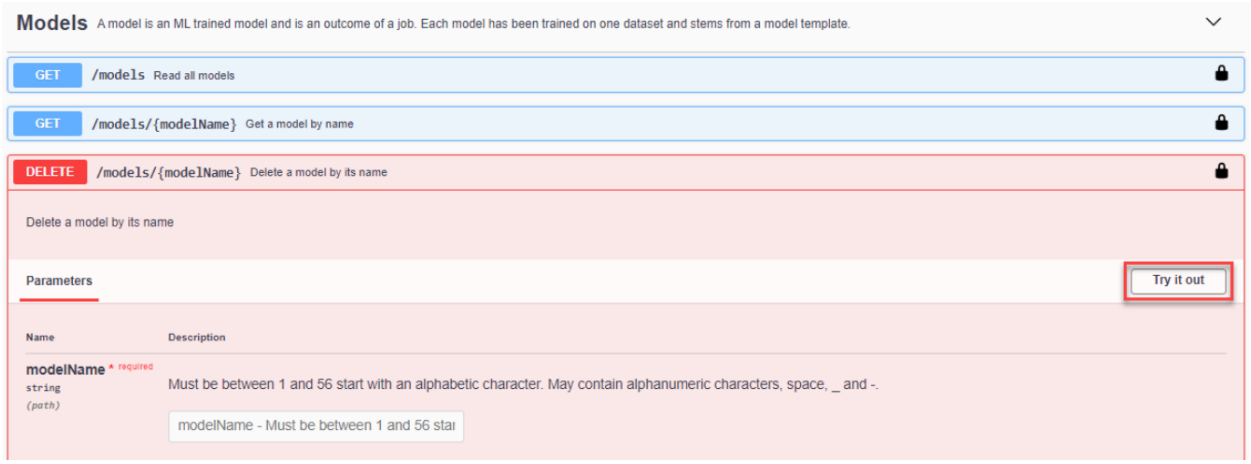
You have successfully `undeployed` your model, but the model is not yet deleted. Instead it isn't in production which means that you cannot make inference requests. You can deploy it again at any time using the `POST /deployments` endpoint.

5.4

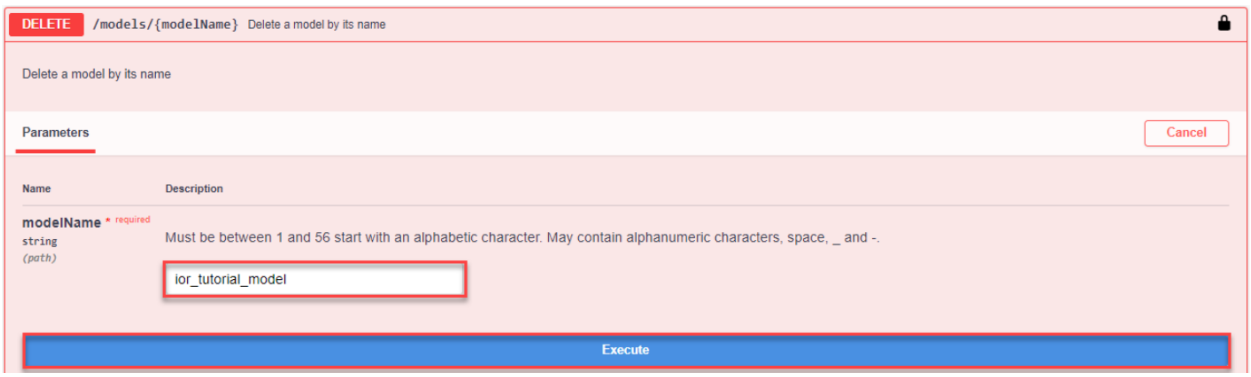
Delete your model (Optional)

Once `undeployed`, you can delete your model.

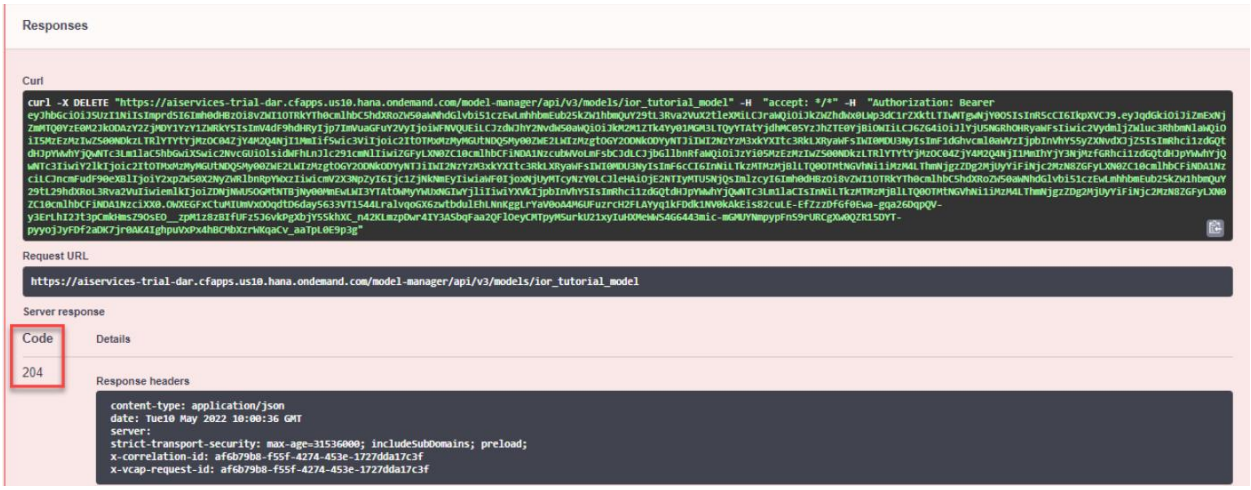
1. Expand the endpoint `DELETE /models/{modelName}` by clicking on it. Then click **Try it out**.



2. Fill the parameter `modelName` with the name of your machine learning model (`ior_tutorial_model`). Use the `GET /models` endpoint in case you no longer have the model `name` in hand.



3. If the response code is `204`, the model has been successfully deleted.

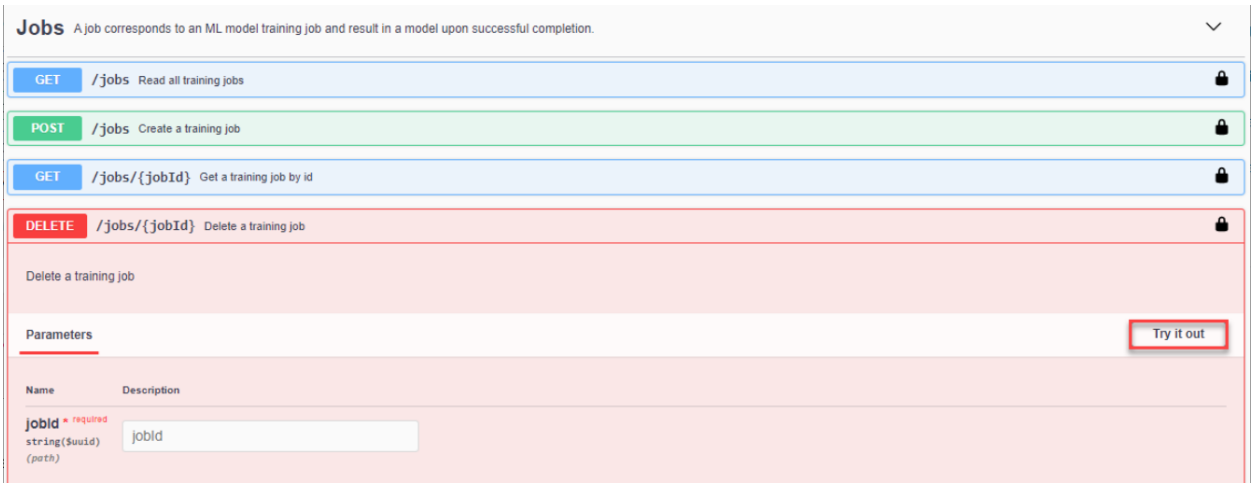


5.5

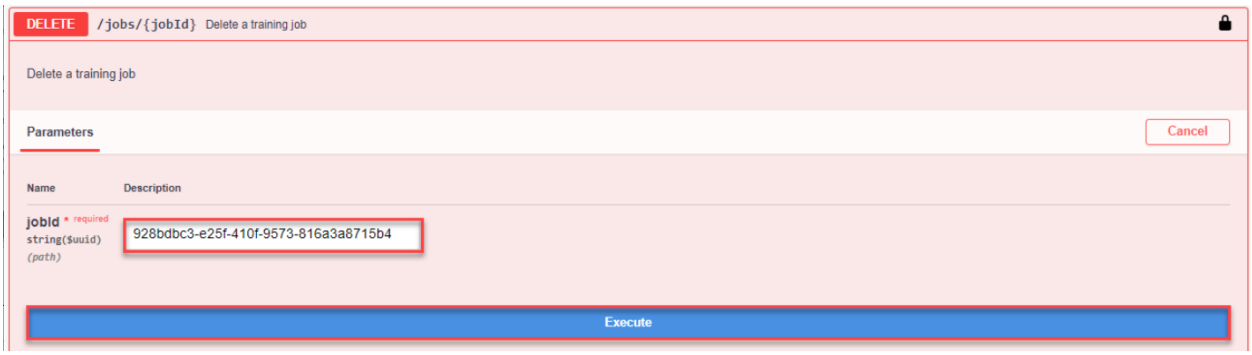
Delete your training job (Optional)

Now that the model is deleted, you can delete the training job that created the model.

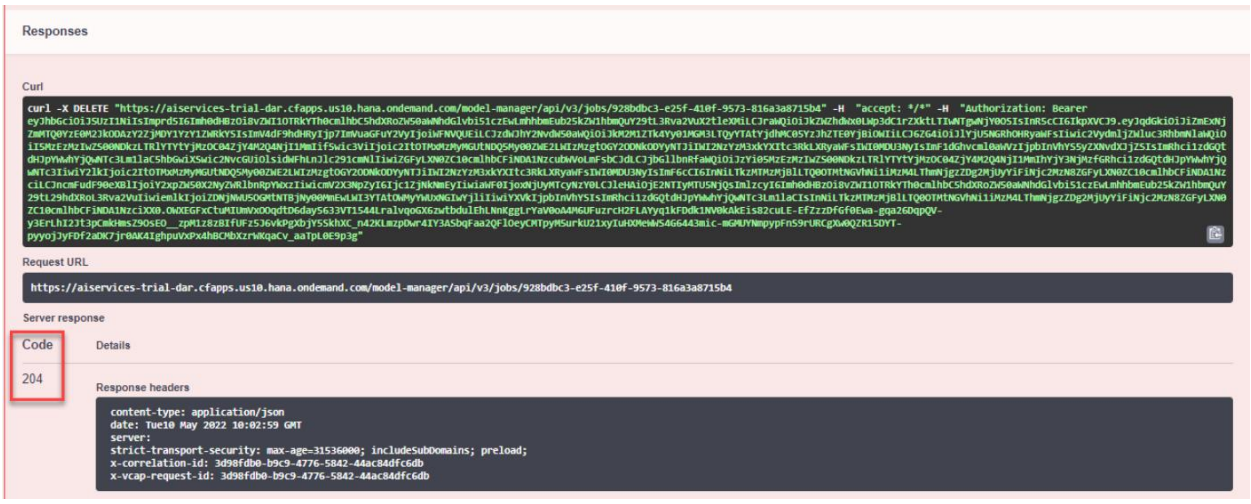
1. Expand the endpoint `DELETE /jobs/{jobId}` by clicking on it. Then click **Try it out**.



2. Fill the parameter `jobId` with the ID of your training job. Use the `GET /jobs` endpoint in case you no longer have the job `id` in hand.



3. If the response code is `204`, the training job has been successfully deleted.

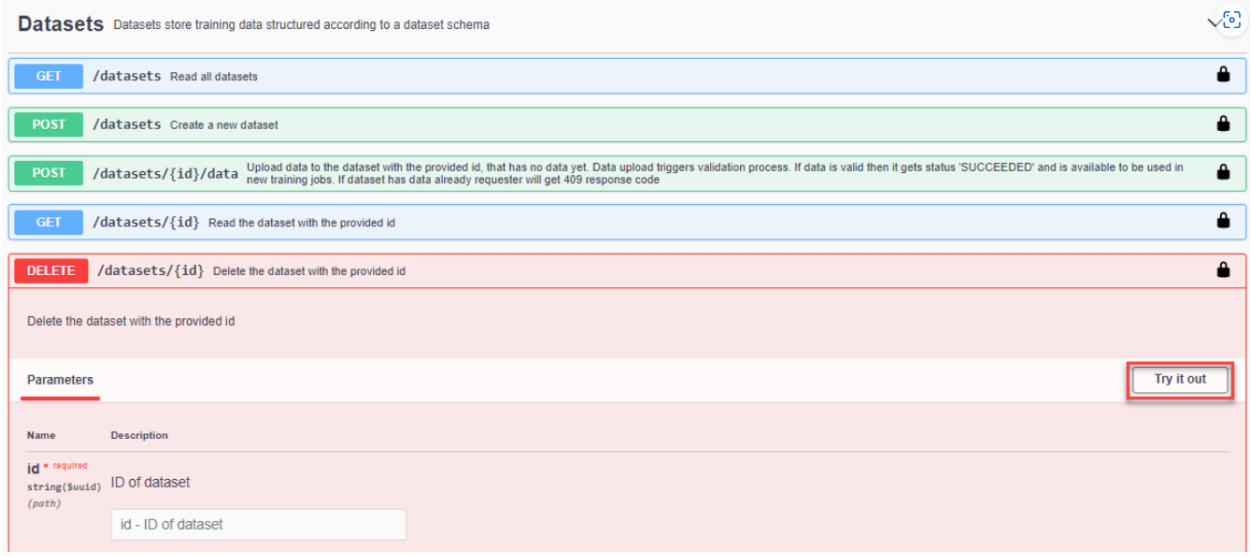


5.6

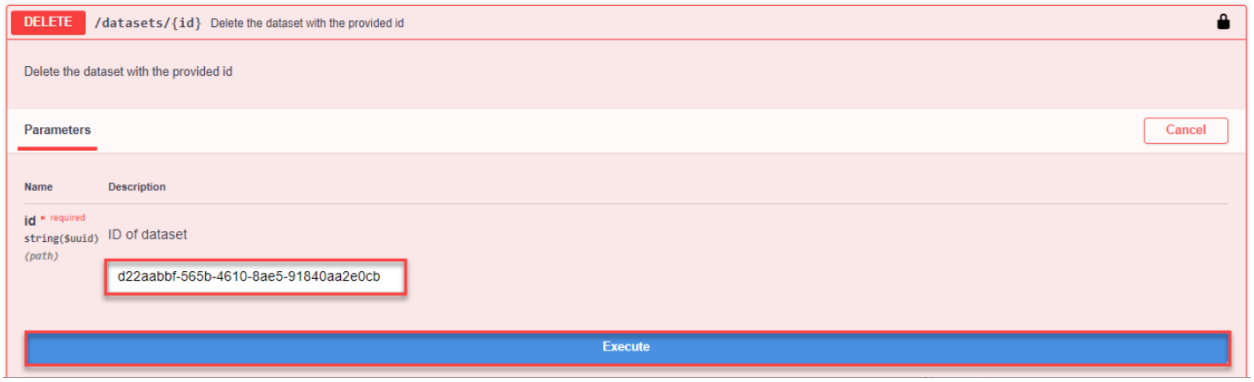
Delete your dataset (Optional)

To clear the uploaded data, you can now delete the dataset as the associated training job is already deleted. For that, go back to the Swagger UI for `dm` and:

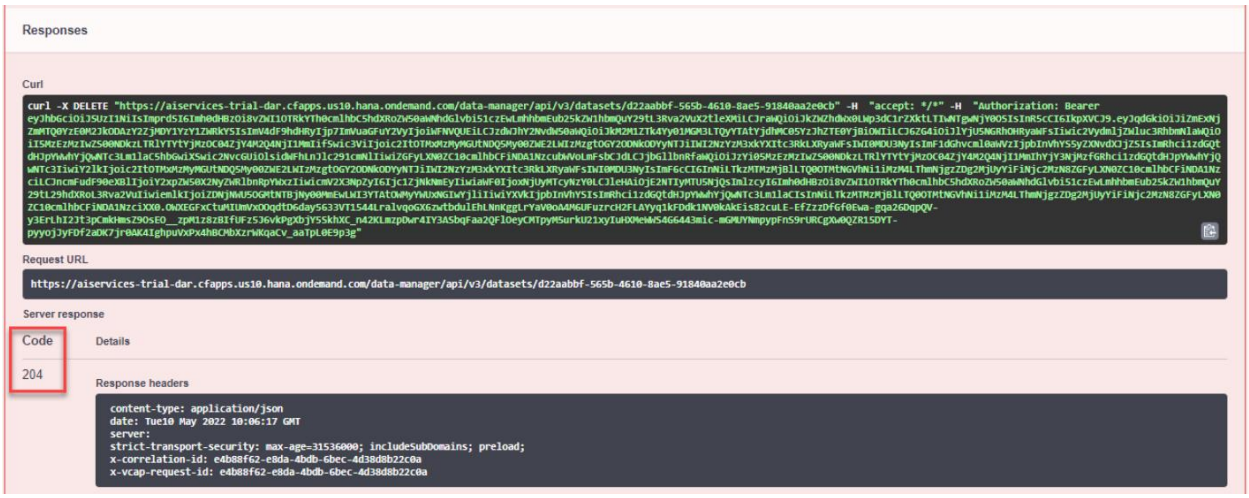
1. Expand the endpoint `DELETE /datasets/{id}` by clicking on it. Then click **Try it out**.



2. Fill the parameter `id` with the ID of your dataset. Use the `GET /datasets` endpoint in case you no longer have the dataset `id` in hand.



3. If the response code is `204`, the dataset has been successfully deleted.

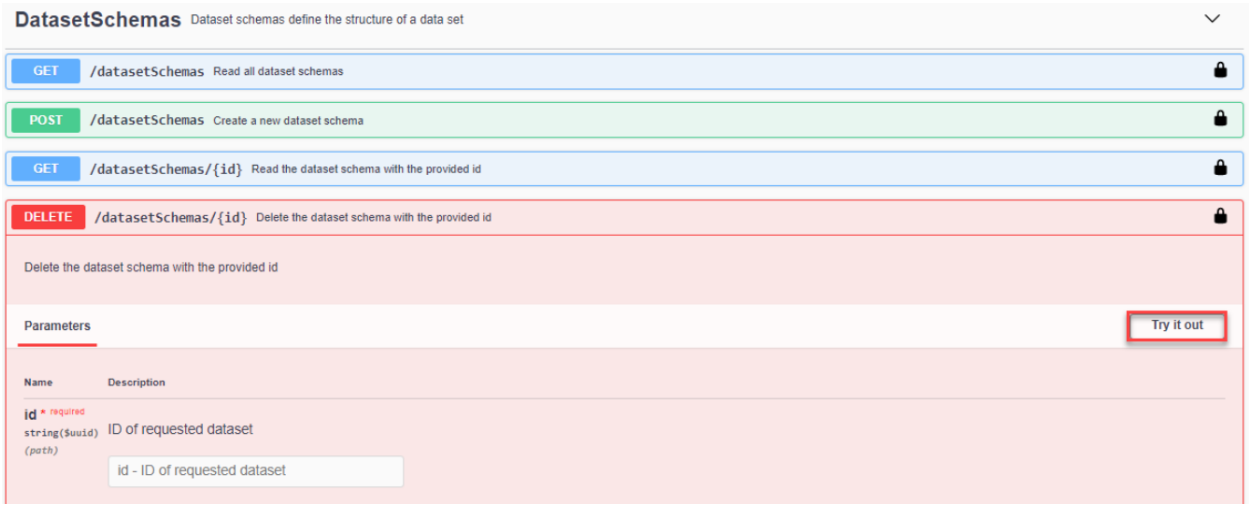


5.7

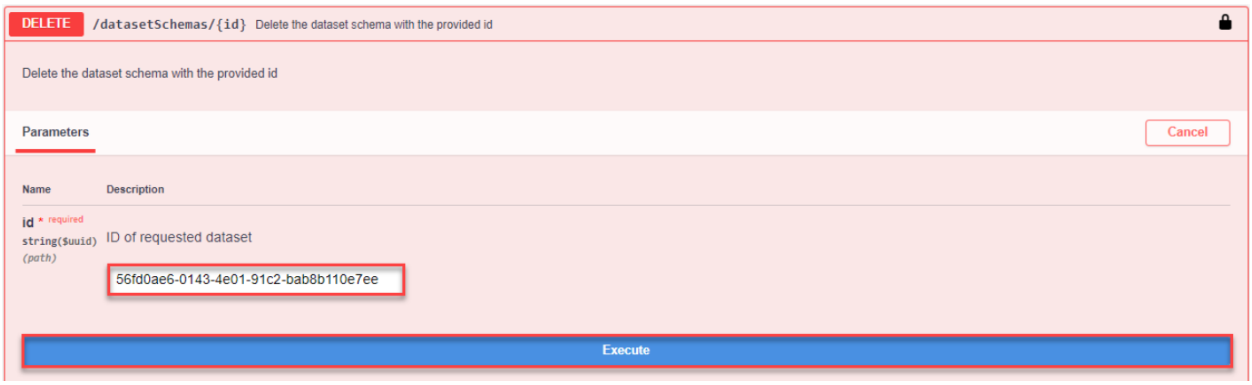
Delete your dataset schema (Optional)

If you do not need your dataset schema anymore, you can delete it as well.

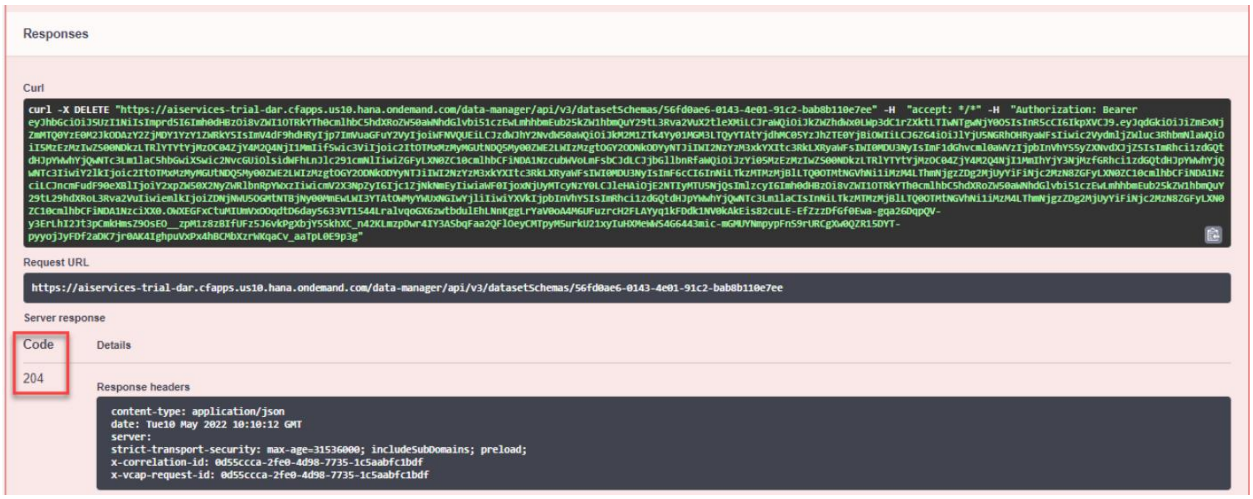
1. Expand the endpoint `DELETE /datasetSchemas/{id}` by clicking on it. Then click **Try it out**.



2. Fill the parameter `id` with the ID of your dataset schema. Use the `GET /datasetSchemas` endpoint in case you no longer have the dataset schema `id` in hand.



3. If the response code is `204`, the dataset schema has been successfully deleted.



Congratulations, you have completed this tutorial.