Cloudera AI 1.5.5

Using Cloudera AI Registry

Date published: 2020-07-16
Date modified: 2025-12-18

CLOUDERA

https://docs.cloudera.com/

Legal Notice

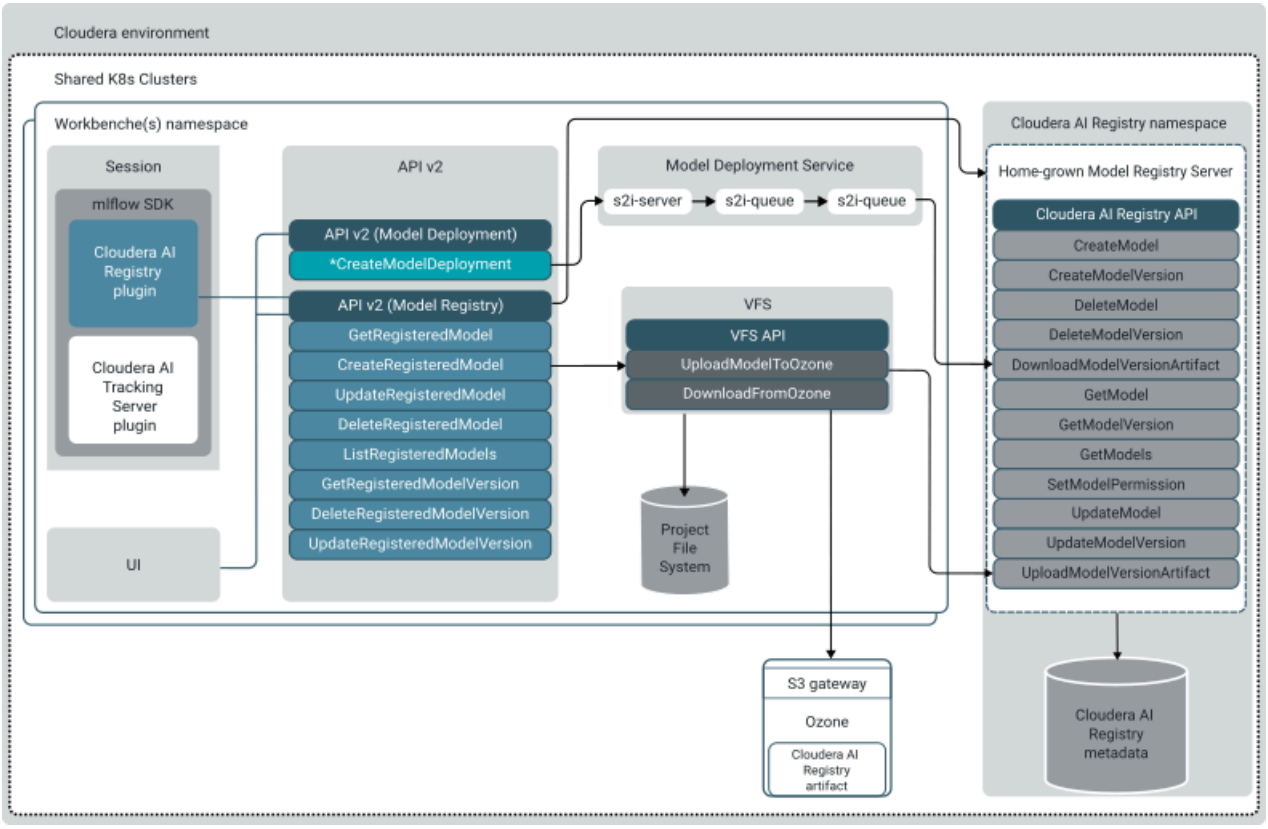
Contents

Using Cloudera AI Registry

Cloudera AI Registry is the core enabler for MLOps, or DevOps for machine learning.

Cloudera AI Registry stores and manages machine learning models and associated metadata, such as the model's version, dependencies, and performance. The registry enables MLOps and facilitates the development, deployment, and maintenance of machine learning models in a production environment.

Figure 1: Cloudera AI Registry on premises



Cloudera AI Registry includes functionality for the following tasks:

- Storing and organizing different versions of a machine learning model and its associated metadata.
- Tracking the lineage of a model, including who created it, when it was created, and any changes made to it over time.
- Providing APIs for accessing and deploying models, as well as for querying and searching the registry.
- Integrating with CI/CD pipelines and other tools used in the MLOps workflow.

Cloudera AI Registry instances help organizations improve the quality and reliability of their machine learning models by providing a centralized location for storing and managing models, as well as enabling traceability and reproducibility of model development. They also make deploying and managing models in a production environment easier by providing a single source for model versions and dependencies.

The Cloudera AI Registry integrates MLFlow and maintains compatibility with the open source ecosystem.

Cloudera AI Registry standalone API

You can use the standalone Cloudera AI Registry API to communicate with the Cloudera AI Registry using the REST client or CLI client.

The Cloudera AI Registry standalone API supports the following functionalities:

- GET/PATCH/DELETE for the model and model version
- GET a curated list of NGC models
- Import external model from NVIDIA NGC or HuggingFace to Cloudera AI Registry through the POST method

Currently, the Cloudera AI Registry Standalone API does not support uploading the models through POST method from the local machine.

API definition

The Swagger definition is available in the Cloudera AI API documentation.

Prerequisites for Cloudera AI Registry standalone API

To set up the Cloudera AI Registry standalone API, configure the Cloudera AI Inference service and import pretrained Models.

Prerequisites for Cloudera AI Inference service

Consider the following prerequisites before setting up Cloudera AI Inference service

- Cloudera Manager supported versions: JSON Web Token-based authentication from Cloudera Control Plane to Cloudera AI Inference service requires Cloudera Manager version 7.12 or higher.
- LDAP Authentication: User authentication is performed by the Knox service running on Cloudera AI Inference service, which relies on the LDAP configuration defined in the Cloudera Control Plane. Without this LDAP integration, access to APIs and model endpoints is denied.
- Ozone Credentials: Cloudera AI Inference service requires read-only Ozone S3 credentials to access Ozone for model downloads. Both Ozone and Cloudera AI Inference service must reside within the same Cloudera Manager, as Ozone certificates are dynamically retrieved from the base cluster during Cloudera AI Inference service provisioning.

Prerequisites to import pretrained models

You must add the URL details to allow them in the firewall rules.

NVIDIA GPU Cloud (NGC)

Add the following URL details so they can be allowed in the firewall's rules.

- prod.otel.kaizen.nvidia.com (NVIDIA open telemetry)
- api.ngc.nvidia.com
- files.ngc.nvidia.com

Hugging Face

Add the following URL details so they can be allowed in the firewall's rules.

- huggingface.co
- cdn-lfs.huggingface.co
- *.cloudfront.net (CDN)

Note: If required, you must allow more URLs based on your requirements.

Authenticating clients for interacting with Cloudera AI Registry API

Clients that interact with the Cloudera AI Registry Standalone API and with model endpoints must obtain a JSON Web Token (JWT) from the Cloudera control plane, which must be passed as a Bearer token in HTTP requests sent to the serving API and endpoints.

To obtain JWT, run the following Cloudera CLI command:

```
$ CDP_TOKEN=$(cdp iam generate-workload-auth-token --workload-name DE | jq -
r
   '.token')
```

In this comment, *DE* is the workload name.

Then pass CDP_TOKEN in the HTTP request header as follows

```
$ curl -H "Authorization: Bearer ${CDP_TOKEN}" <URL>
```

The token obtained using this method expires in one hour.

Role-based authorization

Cloudera AI Registry implements role-based access control.

Users must have the following roles to create an instance of the service in a Cloudera environment:

- EnvironmentAdmin
- MLAdmin (admin user)

Registered Models can be viewed, created, deleted, and modified by users having EnvironmentUser role along with either one of the following roles:

- MLAdmin (admin user)
- MLUser

For more information about the access control for the registered models, see *Model access control*.

Using the REST Client

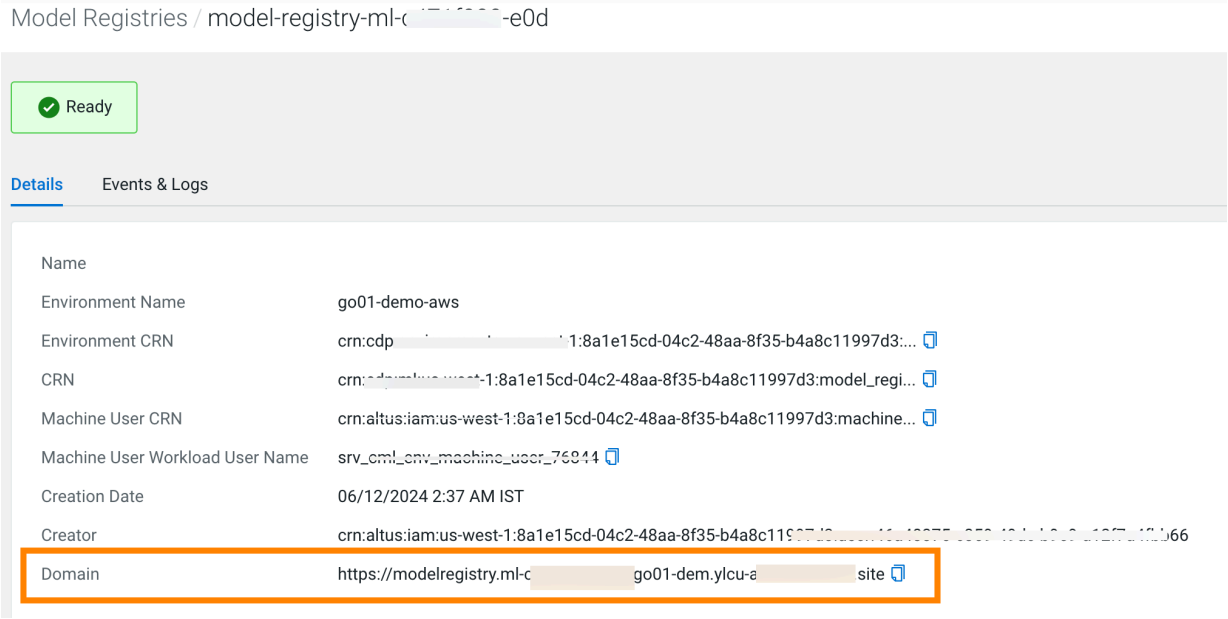
You need the domain information to use the REST client to interact with the registry.

Before you begin

To obtain the domain information, perform the following:

1. In the **Cloudera** console, click the **Cloudera AI** tile.
2. Click AI Registries in the left navigation menu. The AI Registries page displays.

3. Click on the name of the Cloudera AI Registry to display the Cloudera AI Registry information. The Domain name is displayed in the Details tab.



Get all Models

```
curl -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/models | jq
        {
        "models": [
        {
        "created_at": "2024-04-18T15:54:15.543Z",
        "creator": {
        "user_name": "csso_cheyuanl"
        },
        "id": "5bwt-qqe2-e1vg-chqj",
        "name": "foo",
        "tags": null,
        "updated_at": "2024-04-18T15:54:15.543Z",
        "visibility": "private"
        },
```

Get a Model

```
curl -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/models/fx0k
-baf7-yszl-jrt2 | jq
          {
          "created_at": "2024-04-18T15:54:24.940Z",
          "creator": {
          "user_name": "csso_cheyuanl"
          },
          "id": "fx0k-baf7-yszl-jrt2",
          "model_versions": [
          {
          "artifact_uri": "abfs://data@engmldevenvazuresan.dfs.core.windows
.net/modelregistry/fx0k-baf7-yszl-jrt2/y8d8-qluc-00md-h2pw/model.tar.gz",
          "created_at": "2024-04-18T15:54:24.942Z",
          "model_id": "fx0k-baf7-yszl-jrt2",
          "status": "READY",
          "tags": null,
          "updated_at": "2024-04-18T15:54:24.942Z",
```

```
          "user": {
          "user_name": "csso_cheyuanl"
          },
          "version": 1
          }
          ],
          "name": "foo2",
          "tags": null,
          "updated_at": "2024-04-18T15:54:24.940Z",
          "visibility": "private"
          }
```

Get a Model version

```
curl -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/models/fx0k
-baf7-yszl-jrt2/versions/1 | jq
          {
          "artifact_uri": "abfs://data@engmldevenvazuresan.dfs.core.windows.
net/modelregistry/fx0k-baf7-yszl-jrt2/y8d8-qluc-00md-h2pw/model.tar.gz",
          "created_at": "2024-04-18T15:54:24.942Z",
          "model_id": "fx0k-baf7-yszl-jrt2",
          "status": "READY",
          "tags": null,
          "updated_at": "2024-04-18T15:54:24.942Z",
          "user": {
          "user_name": "csso_cheyuanl"
          },
          "version": 1
          }
```

Patch a Model

```
curl -XPATCH -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/mod
els/fx0k-baf7-yszl-jrt2 -d '{
  "visibility": "public"
}'
```

Patch a Model Version

```
curl -XPATCH -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/mod
els/fx0k-baf7-yszl-jrt2/version/1 -d '{
  "tags": [{"key": "k1", "value": "v1"}, "key": "k2", "value": "v2"}]
}'
```

Delete a Model

```
curl -XDELETE -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/mo
dels/vuu6-gcfx-ydio-rit0
```

Delete a Model version

```
curl -XDELETE -s -H "Authorization: Bearer ${CDP_TOKEN}" ${DOMAIN}/api/v2/mo
dels/vuu6-gcfx-ydio-rit0/versions/1
```

Troubleshooting issues with Cloudera AI Registry API

Learn about some of the recommended series of steps to perform when troubleshooting issues related to the Cloudera AI Registry API.

Debugging the model import failure

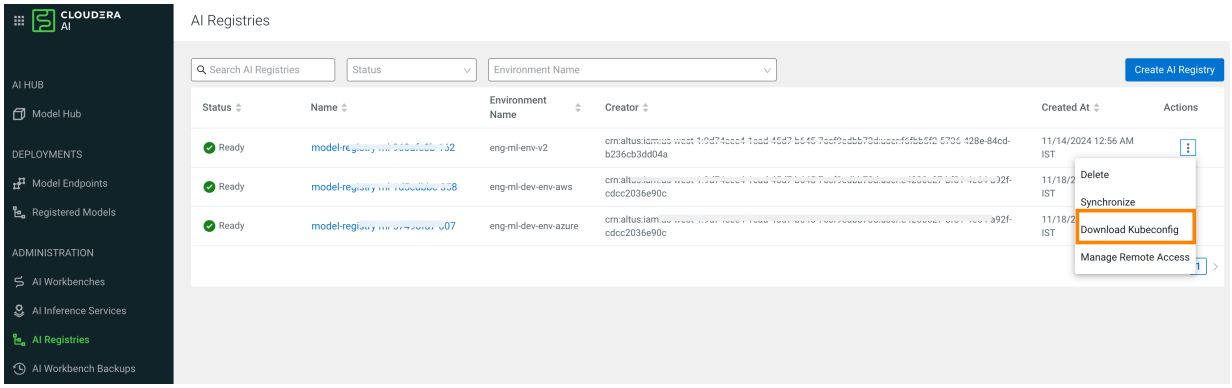
To debug errors that occurred on the Cloudera AI Registry server, you can access the logs found in the API v2 pod.

About this task

Access logs from Cloudera AI Registry Kubernetes cluster.

You can obtain the kubeconfig for the Cloudera AI Registry cluster.

1. In the **Cloudera** console, click the **Cloudera AI** tile.
2. Click AI Registries in the left navigation menu. The AI Registries page displays.
3. In the **Actions** menu, click  and select **Download Kubeconfig**.



In AWS, you need to add your identity under Manage Remote Access to access the Kubernetes cluster.

You must add your identity under Manage Remote Access. For information on granting remote access, see *Granting Remote Access to Cloudera AI Workbench*. After the kubeconfig is set up, run the following `kubectl` command to get logs for the Cloudera AI Registry pod:

```
kubectl logs <AI registry pod name> -n mlx
```

Importing a Hugging Face Model

If your desired Hugging Face model is unavailable on the Model Hub page, you can import those models from the *Hugging Face* website. After you import the model, the newly imported model will be listed on the Registered Models page.

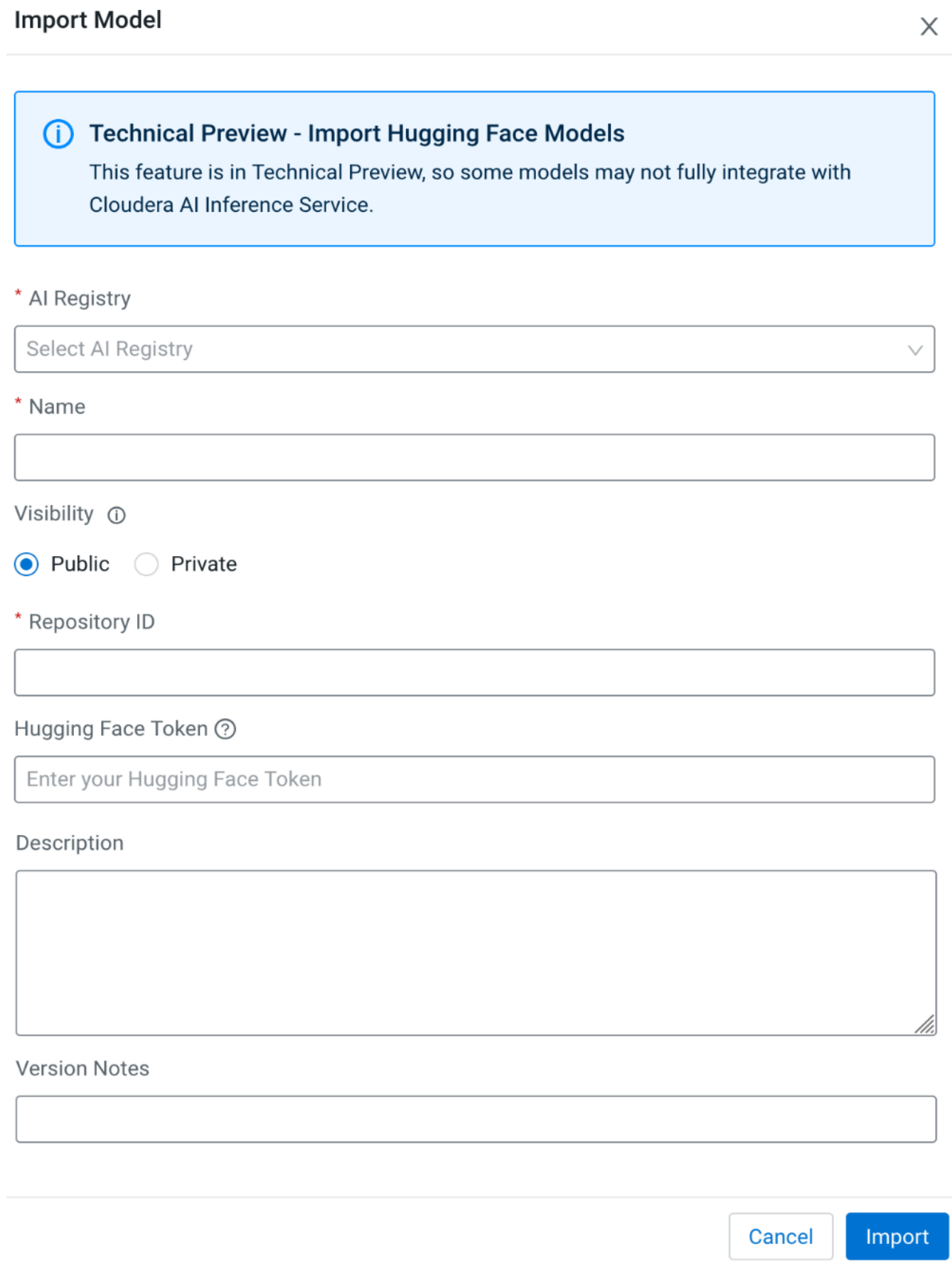
Procedure

1. In the **Cloudera** console, click the **Cloudera AI** tile.

   The **Cloudera AI Workbenches** page displays.
2. Click **Registered Models** under **Deployments** in the left navigation menu.

   The **Registered Models** page displays. The page lists all the models of different Cloudera AI Registries along with the associated metadata.

3. Click Import Model. The Import Model page displays.


Import Model
Technical Preview - Import Hugging Face Models
This feature is in Technical Preview, so some models may not fully integrate with Cloudera AI Inference Service.
* AI Registry
Select AI Registry
* Name
Visibility
Public
Private
* Repository ID
Hugging Face Token
Enter your Hugging Face Token
Description
Version Notes
Cancel
Import

4. In the AI Registry drop-down list, select the AI registry to which you want to import the model.
5. In the Name field, enter a new name for the model you are importing.
6. Select the Visibility as Public or Private. If you select Public, the model is available for other users. If you select Private, the model is displayed on the Registered Models page only for the user who imported it.
7. In the Repository ID field, enter the ID of the Hugging Face model. You can obtain the ID of a model from the Hugging Face website.
8. In the Hugging Face Token field, enter the token obtained from the Hugging Face website.
9. In the Description field, enter a description for the model.
10. In the Version Notes field, enter notes about this version of the model.
11. Click Import.

Results

You can view this newly imported model on the Registered Models page.