

IBM watsonx.ai

Technical Hands-on Lab

Custom Foundation Model

Felix Lee (felix@ca.ibm.com)
Worldwide Technology Enablement

Contents

1. Introducing watsonx.ai Custom Foundation Model.....	3
2. About this Lab	4
2.1 Disclaimer.....	5
3. Getting Help.....	6
4. Prerequisites & Getting Started	7
5. Custom Foundation Model.....	9
5.1 Verifying your model is supported	9
5.2 Downloading the asset from Hugging Face.....	12
5.2.1 Getting a Hugging Face Access Token	12
5.2.2 Downloading a model from Hugging Face	15
5.3 Converting the model to the desired format (if necessary)	18
5.4 Making a TechZone reservation and uploading model files	19
5.4.1 Reserving a specific TechZone pattern	19
5.4.2 Logging into your TechZone environment.....	25
5.4.3 Creating a project in watsonx.ai	27
5.5 Uploading the model to Cloud Object Storage.....	31
5.6 Creating a connection to COS from watsonx.ai.....	40
5.6.1 Getting your Access key and Secret key.....	40
5.6.2 Setting up your connection to COS using Access key and Secret key	43
5.7 Creating a model asset	51
5.8 Deploying a model asset.....	54
5.9 Using the Custom Foundation Model.....	59
5.9.1 Considerations for Custom Foundation Models	63
6. Removing a deployment	64
Appendix A. Revision History.....	66

1. Introducing watsonx.ai Custom Foundation Model

IBM's generative AI strategy is open, trusted, and performant. IBM watsonx.ai is designed to work with models from different sources: IBM-developed models, open source, and other third-party models.

IBM watsonx.ai provides a set of carefully curated models which includes IBM's Granite series models, Meta's llama models, Goggle's flan models, and Mistral AI's mistral and mixtral models. These models provide a wide range of generative AI use cases. For example:

- Generation
- Extraction
- Classification
- Summarization
- Question Answering
- Code generation and translation

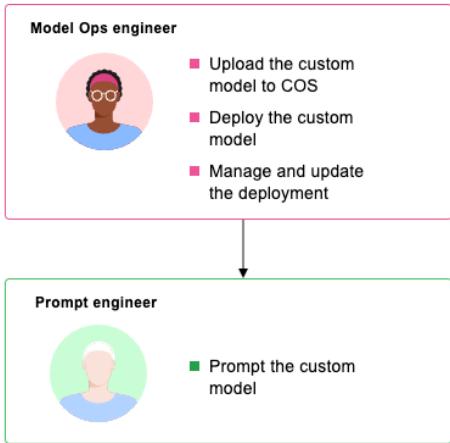
IBM has recognized that clients may have been working with models that they have found satisfactory or might have tuned a model to support their business use cases. Clients will want to deploy these models within watsonx.ai to leverage the rich features. For example:

- Model governance, AI automation, and deployment
- Integration with watsonx.data and watsonx.governance
- Integration with various AI Assistants

The watsonx.ai Custom Foundation Model feature allows clients to bring their own model and deploy it on watsonx.ai. This feature provides the flexibility for clients to implement the AI solutions that are right for their use case.

2. About this Lab

The process for deploying a foundation model and making it available for inferencing includes tasks that are performed by a Model Ops engineer and a Prompt Engineer. The following diagram shows a high-level flow of tasks typically performed by a Model Ops engineer and a Prompt engineer:



This lab focuses on the first 2 tasks of the Model Ops engineer: using the watsonx.ai Custom Foundation Model feature to upload a custom model to your Cloud Object Storage (COS) and then deploying a custom model to watsonx.ai.

The process covered in this lab applies to deploying a model in the watsonx.ai Software as a Service (SaaS) environment in the cloud. For deploying a custom model on-premises, consult the documentation at: <https://tinyurl.com/br7mfnb2>.

2.1 Disclaimer

IBM watsonx.ai is developed and released in an agile manner. In addition to adding new capabilities, the web interface is likely to change over time. Therefore, the screenshots used in this lab may not always look exactly like what you see. You can expect to encounter some of the following:

- Additional foundation models in the library list.
- Changes in the user interface (location of buttons, text for various fields)

These should not affect how the labs work, but have patience and explore.

However, the list of model architectures supported for **Custom Foundation Model** may change. You should always consult **Supported model architectures** on [Planning to deploy a custom foundation model](#) to ensure the model you want to add is based on a supported architecture. Other prerequisites may also change (for example, the format of the model content).

Please raise questions on the [#data-ai-demo-feedback](#) Slack channel (IBMer only). IBM partners can request help at the [Partner Plus Support](#) website.

3. Getting Help

Lab guide help: If you require assistance in interpreting any of the steps in this lab, please post your questions to the [`#data-ai-demo-feedback`](#) Slack channel (IBMer only). IBM partners can request help at the [Partner Plus Support](#) website.

IBM watsonx.ai: Assistance with the watsonx.ai product itself is available in the [`#watsonx-ai-feedback`](#) (IBMer) and the [`#watsonx-ai-enablement`](#) Slack channels (IBMer only). Additionally, please refer to the [watsonx.ai documentation](#) as needed.

4. Prerequisites & Getting Started

- This lab downloads a model from Hugging Face. You will need to have a Hugging Face account to do so. You can go to <https://huggingface.co/join> and provide the necessary information to sign up for an account.
- You need a Mac OS or Linux environment with sufficient disk space (2 GB) to download the model used for this exercise from Hugging Face.
- **Python 3.9 or above** installed (more details below).
- This lab was verified using the **Chrome** browser. It does NOT work with **Safari**.
- You should be comfortable working with basic Linux/Unix navigation (switching directories, listing files, etc.).
- There are many steps in this lab. Your local workstation may also have existing Python environments. Be patient and be prepared to work through minor issues.

4.1 Examine your Python version

You must meet the minimum Python version requirements for this lab to work. Simply run the **python** command to verify what version you are using.

```
[felixl@Felixs-MacBook-Pro ~ % python
Python 3.7.4 (main, Jul  5 2023, 08:40:20) [Clang 14.0.6 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> ]
```

In the screen capture above, the Python version is 3.7.4, which is below the minimum requirement. This means that the lab will not work.

The process of updating Python on your workstation is beyond the scope of this lab. However, you will need to update to at least 3.9 for this lab to work. In this example, once Python is updated to 3.11, rerunning the **python** command shows the following:

```
[felixl@Felixs-MacBook-Pro ~ % python
Python 3.11.4 (main, Jul  5 2023, 08:40:20) [Clang 14.0.6 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> ]
```

The **python** command shows that the version is > 3.9. It is fine to proceed with the lab.

4.1 Using TechZone

- You must use a TechZone account to perform this lab, or you will run into missing service or out-of-tokens issues. When you use TechZone, you will not (and should not) need to subscribe to any additional services.
- Your TechZone account must support the Standard Plan for Watson Machine Learning. This is a higher requirement than most TechZone patterns. You must use the specific pattern created for this lab (outlined in Section 5.4). This lab will NOT work if you reserve the standard watsonx.ai/.governance SaaS pattern.

5. Custom Foundation Model

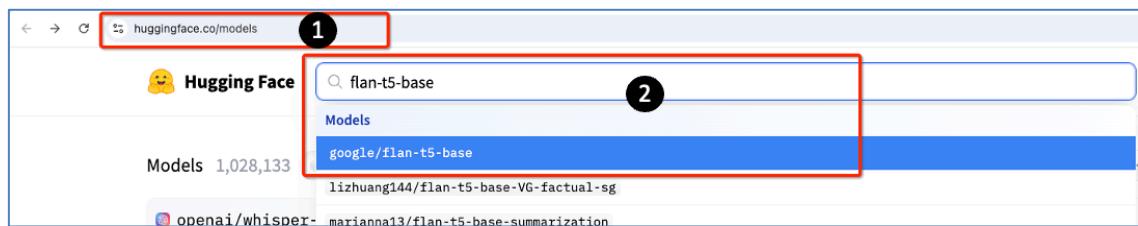
At a high level, adding your custom foundation model to watsonx.ai involves these steps:

1. Verifying that watsonx.ai supports the model you want to add
2. Getting the model files (in this lab, you will get it from Hugging Face)
3. Converting the model file into the desired format (if necessary)
4. Creating a project (a sandbox in this exercise) in watsonx.ai (to do so requires making a specific TechZone reservation)
5. Storing the model files in a supported Cloud Object Storage (COS) using the storage bucket associated with your project
6. Creating a deployment space and setting up a connection to your COS instance.
7. Creating a model asset from the COS content in the deployment space
8. Deploying the model asset
9. Using the Custom Foundation Model

5.1 Verifying your model is supported

First, you need to ensure the model you want to use is supported by watsonx.ai. For this lab, you will use the **flan-t5-base** model as an example. The purpose of this lab is to demonstrate the process. The steps are identical for other models.

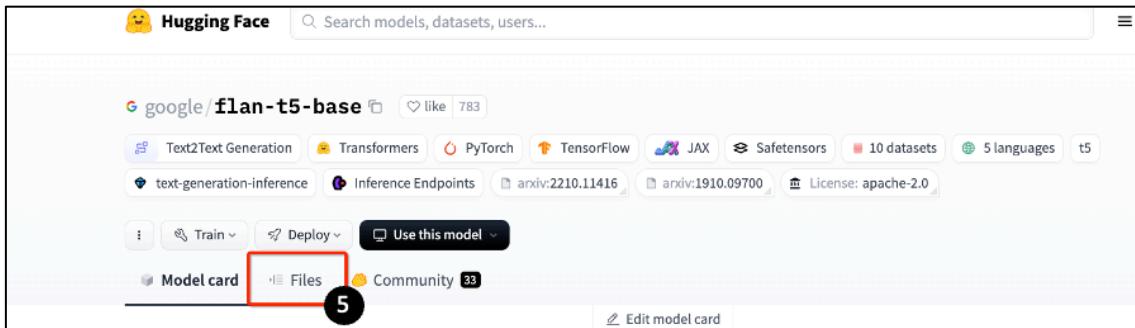
1. Go to <https://huggingface.co/models>
2. Enter **flan-t5-base** in the search field. Hugging Face will pull up the relevant entry.



3. Select **google/flan-t5-base** and its associated Hugging Face page appears, displaying the model card for the **flan-t5-base** model.

Note: **google/flan-t5-base** is the model name (including the “google” part).

4. Click the **Files** tab.



5. The **Files** tab opens. For this lab, you require the following entries (or the model cannot be deployed to watsonx.ai):
 - a. **config.json** - required to load the model in the **Text Generation Inference Service** (TGIS) runtime.
 - b. **tokenizer.json**
 - c. **model.safetensors** (the model must be in this format)
6. Click on the **config.json** file and open it. Examine the content and look for **model_type** which indicates the model family/architecture. You must ensure that this belongs to the list of supported architecture.

```

1  {
2    "architectures": [
3      "T5ForConditionalGeneration"
4    ],
5    "d_ff": 2048,
6    "d_kv": 64,
7    "d_model": 768,
8    "decoder_start_token_id": 0,
9    "dropout_rate": 0.1,
10   "eos_token_id": 1,
11   "feed_forward_proj": "gated-gelu",
12   "initializer_factor": 1.0,
13   "is_encoder_decoder": true,
14   "layer_norm_epsilon": 1e-06,
15   "model_type": "t5", 6
16   "n_positions": 512,

```

In this case, the **model_type** is **t5**.

7. Always check the model against the list of [supported architecture](#). The model you want to deploy to watsonx.ai must have a supported architecture. As of November 2024, the list reads:

Supported model architectures, quantization methods, parallel tensors support, and configuration sizes

Model architecture type	Supported quantization method	Supports parallel tensors (multiGpu)	Supported configurations
bloom	N/A	Yes	Small, Medium, Large
codegen	N/A	No	Small
falcon	N/A	Yes	Small, Medium, Large
gpt_bigcode	gptq	Yes	Small, Medium, Large
gpt-neox	N/A	Yes	Small, Medium, Large
gptj	N/A	No	Small
llama	gptq	Yes	Small, Medium, Large
mixtral	gptq	No	Small
mistral	N/A	No	Small
mt5	N/A	No	Small
mpt	N/A	No	Small
t5	N/A	Yes	Small, Medium, Large

As shown above, **t5** is a supported architecture with a Small, Medium, or Large watsonx.ai configuration (you will use this information later).

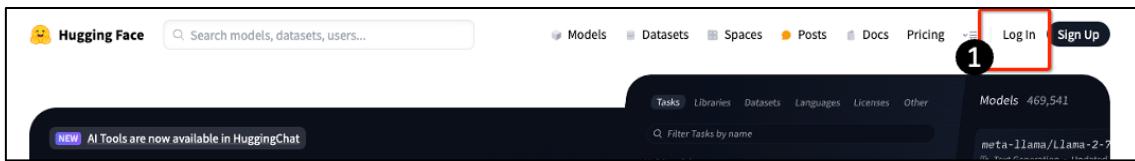
Now that you have verified the model you will use in this lab is supported, you need to download the model files from Hugging Face.

5.2 Downloading the asset from Hugging Face

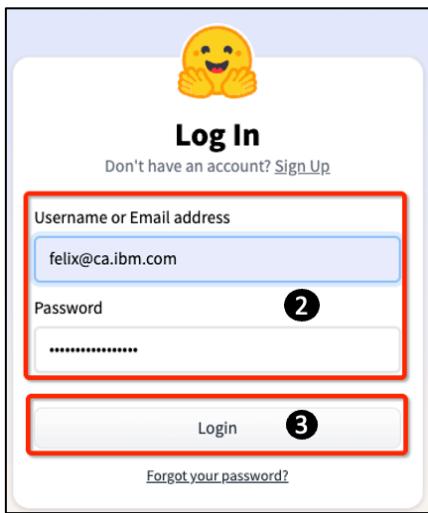
You need to have a Hugging Face account to perform the instructions in this section. If you do not already have one, you can go to <https://huggingface.co/join> and provide the necessary information to sign up.

5.2.1 Getting a Hugging Face Access Token

1. You need a Hugging Face access token to download model files from Hugging Face. Go to <https://huggingface.co/> and click **Log in** to log into your Hugging Face account.



2. Enter your **Username/email, Password**.
3. Click **Login**



4. Click **Settings** on the left.

Hugging Face is way more fun with friends and colleagues! 😊 [Join an organization](#)

+ New

Following 0 ▾

All Models Datasets Spaces Papers Collections Community Posts Upvotes Likes

NEW Follow your favorite AI creators

wangfuyun · Video editing and synthesis

Xenova · Enabling AI in the browser

Refresh List X

Follow Follow

5. Click Access Tokens on the left.

Felix Lee
felix2017

Profile Settings

Profile

Account

Authentication

Organizations

Billing

Access Tokens 5

SSH and GPG Keys

Full name: Felix Lee

Avatar (optional):

Homepage (optional):

AI & ML interests (optional):

6. Click + Create new token.

Note: You may already have an access token. However, you may still want to create a new one for this lab.

Access Tokens

User Access Tokens

Access tokens authenticate your identity to the Hugging Face Hub and allow applications to perform actions based on token permissions. ⓘ Do not share your **Access Tokens** with anyone; we regularly check for leaked Access Tokens and remove them immediately.

You have no Access Token

+ Create new token 6

7. Enter CFM token in the Token name field.
8. Select Read access to contents of all public gated repos you can access (you can select other permissions as well, but use this permission for the lab).

Create new Access Token

Token type
 Fine-grained Read Write
This cannot be changed after token creation.

Token name
 7

User permissions (felix2017)

Repositories	Inference
<input type="checkbox"/> Read access to contents of all repos under your personal namespace	<input type="checkbox"/> Make calls to the serverless Inference API
<input checked="" type="checkbox"/> Read access to contents of all public gated repos you can access	<input type="checkbox"/> Make calls to Inference Endpoints
<input type="checkbox"/> Write access to contents/settings of all repos under your personal	<input type="checkbox"/> Manage Inference Endpoints

9. Scroll down to the bottom and click **Create token**

Org settings

Read access to organizations settings
 Write access to organizations settings / member management

Collections

Read access to all collections in selected organizations
 Write access to all collections in selected organizations

Create token 9

10. The **Save your Access Token** page opens. As you can see below, the actual token is intentionally redacted in this lab for obvious reasons. Click **Copy** and save the tokens somewhere safe where you will be able to find and copy. Once you close this window, you will not be able to retrieve this token. If you did not save it, you will have to create another one.

11. Click **Done**.

Save your Access Token

Save your token value somewhere safe. You will not be able to see it again after you close this modal. If you lose it, you'll have to create a new one.

10

Copy

Name	Permissions
CFM	FINEGRAINED

11

Done

Next, you need to download the **flan-t5-base** model from Hugging Face

5.2.2 Downloading a model from Hugging Face

1. On your Mac OS or Linux workstation, open a terminal session.
2. You should create a special subdirectory and create a new Python environment. In this example, a subdirectory **CFM** is created under the **/Users/felixl/Downloads** directory. You will substitute in your appropriate values (for example, **/home/<username>**, or **/home/<username>/Downloads**); where you use your name as opposed to **felixl**.

Enter these commands:

```
cd /Users/<username>/Downloads  
mkdir CFM  
python -m venv /Users/<username>/Downloads/CFM/venv  
ls /Users/<username>/Downloads/CFM
```

You should now see a **venv** directory under the **CFM** directory.

```
felixl@Felixs-MacBook-Pro ~ % cd /Users/felixl/Downloads  
felixl@Felixs-MacBook-Pro Downloads % python -m venv /Users/felixl/Downloads/CFM/venv  
felixl@Felixs-MacBook-Pro Downloads % ls /Users/felixl/Downloads/CFM  
venv  
felixl@Felixs-MacBook-Pro Downloads %
```

3. Use the newly created Python environment by issuing the following commands:

```
cd /Users/<username>/Downloads/CFM  
source venv/bin/activate
```

Note how the left part of the terminal changes to reflect you are in the virtual environment (**venv**).

```
felixl@Felixs-MacBook-Pro Downloads % cd /Users/felixl/Downloads/CFM  
felixl@Felixs-MacBook-Pro CFM % source venv/bin/activate  
venv) felixl@Felixs-MacBook-Pro CFM %  
(venv) felixl@Felixs-MacBook-Pro CFM %
```

4. Create a subdirectory under the **CFM** directory to host the model you are going to download by entering the following commands:

```
mkdir flan-t5-base  
ls -l
```

You should see this:

```
(venv) felixl@Felixs-MacBook-Pro CFM % mkdir flan-t5-base
(venv) felixl@Felixs-MacBook-Pro CFM % ls -l
total 0
drwxr-xr-x  2 felixl  staff   64  3 Oct 12:51 flan-t5-base
drwxr-xr-x  6 felixl  staff  192  3 Oct 12:30 venv
(venv) felixl@Felixs-MacBook-Pro CFM %
```

5. Install the **huggingface-cli** package using the following command:

```
pip install -U "huggingface_hub[cli]"
```

You will see something similar to this:

```
(venv) felixl@Felixs-MacBook-Pro CFM % pip install -U "huggingface-hub[cli]"
Requirement already satisfied: huggingface-hub[cli] in /Users/felixl/miniconda3/lib/python3.11/site-packages (0.23.0)
Collecting huggingface-hub[cli]
  Using cached huggingface_hub-0.24.6-py3-none-any.whl (417 kB)
Requirement already satisfied: filelock in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (3.14.0)
Requirement already satisfied: fsspec>=2023.5.0 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (2024.3.1)
Requirement already satisfied: packaging>=20.9 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (23.2)
Requirement already satisfied: pyyaml>=5.1 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (6.0.1)
Requirement already satisfied: requests in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (2.31.0)
Requirement already satisfied: tqdm>=4.42.1 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (4.65.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from huggingface-hub[cli]) (4.11.0)
Collecting InquirerPy==0.3.4 (from huggingface-hub[cli])
  Using cached InquirerPy-0.3.4-py3-none-any.whl (67 kB)
Collecting pfzy<0.4.0,>=0.3.1 (from InquirerPy==0.3.4->huggingface-hub[cli])
  Using cached pfzy-0.3.4-py3-none-any.whl (8.5 kB)
Requirement already satisfied: prompt-toolkit<4.0.0,>=3.0.1 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from InquirerPy==0.3.4->huggingface-hub[cli]) (3.0.39)
Requirement already satisfied: charset-normalizer<4,>=2 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from requests->huggingface-hub[cli]) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from requests->huggingface-hub[cli]) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from requests->huggingface-hub[cli]) (1.26.16)
Requirement already satisfied: certifi>=2017.4.17 in /Users/felixl/miniconda3/lib/python3.11/site-packages (from requests->huggingface-hub[cli]) (2023.5.7)
Requirement already satisfied: wcwidth in /Users/felixl/miniconda3/lib/python3.11/site-packages (from prompt-toolkit<4.0.0,>=3.0.1->InquirerPy==0.3.4->huggingface-hub[cli]) (0.2.6)
Installing collected packages: pfzy, InquirerPy, huggingface-hub
  Attempting uninstall: huggingface-hub
    Found existing installation: huggingface-hub 0.23.0
    Not uninstalling huggingface-hub at /Users/felixl/miniconda3/lib/python3.11/site-packages, outside environment /Users/felixl/Downloads/CFM/venv
      Can't uninstall 'huggingface-hub'. No files were found to uninstall.
Successfully installed InquirerPy-0.3.4 huggingface-hub-0.24.6 pfzy-0.3.4
(venv) felixl@Felixs-MacBook-Pro CFM %
```

Note: If your Python version is below 3.9, you may not have the pip command available in your venv. If this is an issue, check your Python version and upgrade it before proceeding.

6. Verify that **huggingface-cli** is set up by running the following command:

```
huggingface-cli --help
```

You should see help output for this command as follows:

```
(venv) felixl@Felixs-MacBook-Pro CFM % huggingface-cli --help
usage: huggingface-cli <command> [<args>]

positional arguments:
  {env,login,whoami,logout,repo,upload,download,lfs-enable-largefiles,lfs-multipart-upload,sca
n-cache,delete-cache,tag}
      huggingface-cli command helpers
  env                  Print information about the environment.
  login                Log in using a token from huggingface.co/settings/tokens
  whoami               Find out which huggingface.co account you are logged in as.
  logout               Log out
  repo                 {create} Commands to interact with your huggingface.co repos.
  upload               Upload a file or a folder to a repo on the Hub
  download             Download files from the Hub
  lfs-enable-largefiles
      Configure your repository to enable upload of files > 5GB.
  scan-cache           Scan cache directory.
  delete-cache         Delete revisions from the cache directory.
  tag                 {create, list, delete} tags for a repo in the hub

options:
  -h, --help            show this help message and exit
(venv) felixl@Felixs-MacBook-Pro CFM %
```

Note: Hugging Face CLI will deploy a version based on your Python version. If you run the **huggingface-cli –help** command and you do NOT see a download option, it is indicative that you have the wrong Python version. Check and upgrade your Python before proceeding.

7. You will need to use the Hugging Face token you saved in Section 5.2.1 Step 10. But first, you need to set up a few environment variables by entering the several commands in the terminal as follows:

```
export HF_TOKEN=<your Hugging Face token> (intentionally redacted below)
export MODEL_NAME="google/flan-t5-base"
export MODEL_DIR=<your base directory>/CFM/flan-t5-base"
```

Note: you need to use the full path name you saw in Section 5.1, Step 3:
google/flan-t5-base.

For this example, the last variable is set to **/Users/felixl/Downloads/CFM/flan-t5-base**. (the token is redacted in the example below).

```
(venv) felixl@Felixs-MacBook-Pro CFM % export HF_TOKEN="redacted"
(venv) felixl@Felixs-MacBook-Pro CFM % export MODEL_NAME="google/flan-t5-base"
(venv) felixl@Felixs-MacBook-Pro CFM % export MODEL_DIR="/Users/felixl/Downloads/CFM/flan-t5-base"
(venv) felixl@Felixs-MacBook-Pro CFM %
```

On your terminal, log into Hugging Face and download the model with these commands:

```
huggingface-cli login --token ${HF_TOKEN}
huggingface-cli download ${MODEL_NAME} --local-dir ${MODEL_DIR} --cache-dir
${MODEL_DIR}
```

You should see something like this (the output is partially captured here):

```
(venv) felixl@Felixs-MacBook-Pro CFM % huggingface-cli login --token ${HF_TOKEN}
The token has not been saved to the git credentials helper. Pass `add_to_git_credential=True` in this function directly or `--add-to-git-credential` if using via `huggingface-cli` if you want to set the git credential as well.
Token is valid (permission: fineGrained).
(venv) felixl@Felixs-MacBook-Pro CFM % huggingface-cli download ${MODEL_NAME} --local-dir ${MODEL_DIR} --cache-dir ${MODEL_DIR}
Fetching 12 files:  0%|██████████| 0/12 L00:00<?, ?it/s
Downloading 'pytorch_model.bin' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/pytorch_model.bin.521ad557548700c7f342a804b51c7c17440c33662c387883c0a39ebbdec17a28.incomplete'
Downloading 'special_tokens_map.json' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/special_tokens_map.json.2c19eb6e3b583f52d34b93b5978d3d30b667682.incomplete'
Downloading 'model.safetensors' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/model.safetensors.1dfb70afcdcedceb9f9fae2f9b68e004ad934361fb35b9b2bd50b45ea90790fc8.incomplete'
Downloading '.gitattributes' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/.gitattributes.e2e68308705c59d0e047f13c1116ed223f9b7fec.incomplete'
Downloading 'generation_config.json' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/generation_config.json.d52815623b046b7db1c4b957b5a83a8ad30b146a.incomplete'
Downloading 'config.json' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/config.json.d41e16cd4cd58621695da4f94132f516d07251af.incomplete'
special_tokens_map.json: 100%|██████████| 2.20k/2.20k [00:00<00:00, 35.5MB/s]
Download complete. Moving file to /Users/felixl/Downloads/CFM/flan-t5-base/special_tokens_map.json      | 0.00/2.20k [00:00<?, ?B/s]
Downloading 'README.md' to '/Users/felixl/Downloads/CFM/flan-t5-base/.huggingface/download/README.md.0926373f7a2843e99ce6a2d19f3aa834ebf1f944.incomplete'
```

A total of 12 files should be downloaded.

Note: the **flan-t5-base** model is relatively small (about 4 GB in size). Be aware that some models are huge (think hundreds of GBs).

- Verify the download with the command (sub in your appropriate directory):

```
ls /Users/felixl/Downloads/CFM/flan-t5-base
```

You should see this:

```
(venv) felixl@Felixs-MacBook-Pro CFM % ls /Users/felixl/Downloads/CFM/flan-t5-base
README.md           generation_config.json    pytorch_model.bin        tf_model.h5
config.json         model.safetensors       special_tokens_map.json   tokenizer.json
flax_model.msgpack  models--google--flan-t5-base spiece.model        tokenizer_config.json
```

5.3 Converting the model to the desired format (if necessary)

To deploy a custom foundation model on watsonx.ai, that models need to be in **safetensors** format. In the example here, the **flan-t5-base** model artifacts include a **model.safetensors** file. If the model you downloaded does not have this format, you will need to convert the model to **safetensors** format.

The instructions to do so are documented in [Downloading a custom foundation model and setting up Storage](#) - look under the section titled [Converting a model to the required format](#).

5.4 Making a TechZone reservation and uploading model files

Once you have downloaded the model files, you are ready to set up watsonx.ai. As mentioned in Section 4, using **Custom Foundation Model** requires a specific TechZone pattern. In this section, you will make a reservation here and then upload the model files you just downloaded into the Cloud Object Storage (COS).

5.4.1 Reserving a specific TechZone pattern

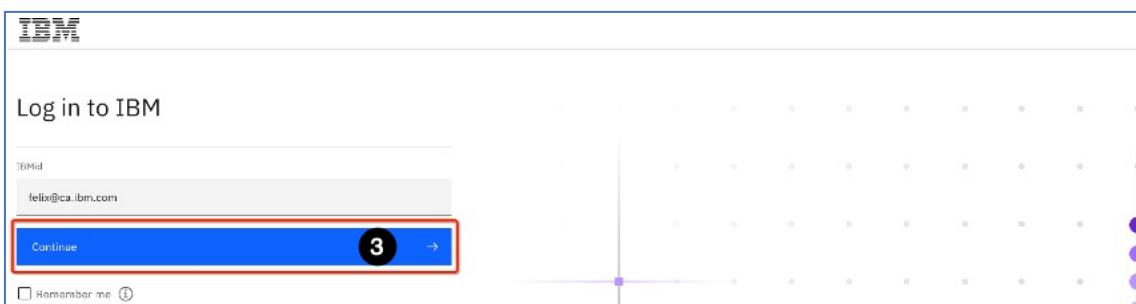
The detailed setup instructions are as follows.

1. Go to the [IBM TechZone](#) website.
2. If you already have an IBMid, skip to Step 3. If you do not have one, click [Create an IBMid](#).

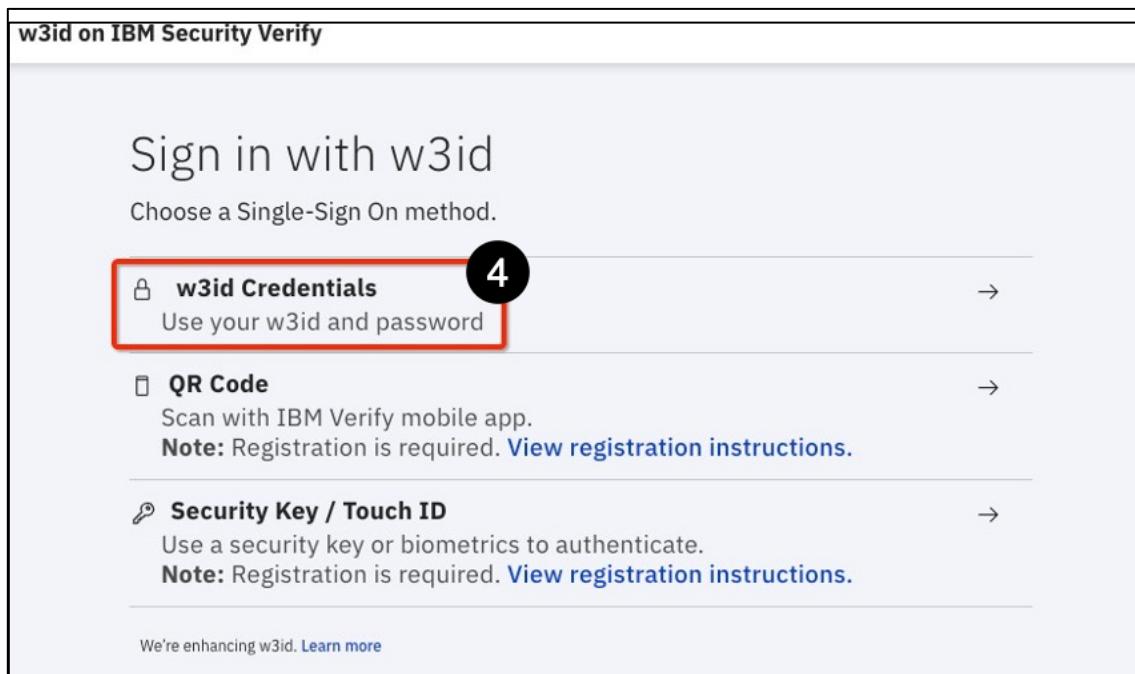


Simply follow the instructions from the subsequent panels to create your IBMid.

3. Provide your IBMid and click **Continue**.



4. Select your Single-Sign-On method. The **w3id Credentials** option is selected in this example:



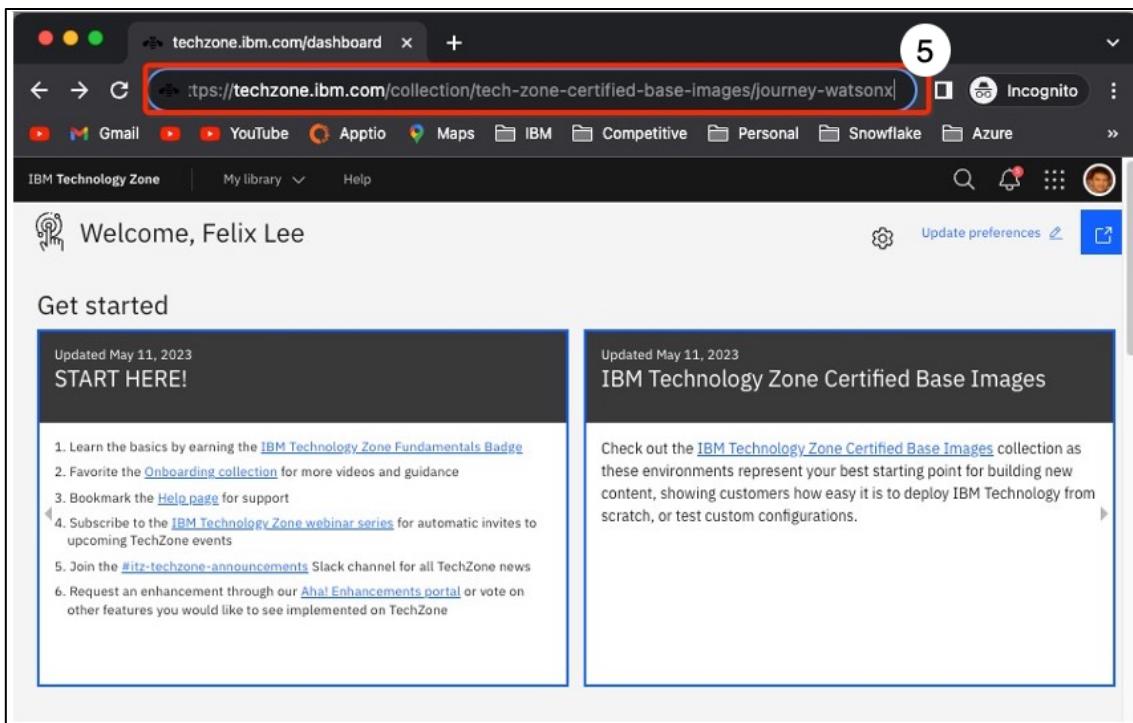
5. Provide the necessary information (w3id and password, QR code, etc.) and sign in.

If this is the very first time you are logging onto the TechZone, you will be brought to the TechZone dashboard. Copy and paste the following HTTP address in the browser's search field (do not use the magnifying glass icon) and press **Enter**:

<https://techzone.ibm.com/collection/tech-zone-certified-base-images/journey-watsonx>

This will take you to the **TechZone Certified Base Images** page.

If you have been to TechZone before, you might be taken directly to the TechZone Certified Base Images page already. In that case, proceed to Step 6.



6. Your browser will open to the TechZone Certified Base Images page. On the left-hand side, the **watsonx** section should be highlighted. If not, click on it.

TechZone Certified Base Images
watsonx

★★★★★ (25) Rate this resource

Developer Base Image - docker-based image best suited for education and test activities.

Stand Alone Base Image - OpenShift-based image best suited for education and demo activities.

SaaS Offerings - Available for Customer Demo and Proof-of-Technology purposes only.

Jul 6, 2023 watsonx.data Developer Base Image This image contains the required software and prerequisites that are needed to run the IBM watsonx.data software.	Sep 21, 2023 Ibmcloud 2: us-east, us-east, eu-de, eu-gb, eu-gb, jp-tok watsonx.data Standalone - OCP on VMWare Self-Managed watsonx.data Standalone UPI OpenShift cluster (VMware on IBM Cloud) with ODF (OCS) support. watsonx.data: <ul style="list-style-type: none">• watsonx.data operator• GA Version 1.0.0• Deployed via OpenShift GitOps and OpenShift Pipelines Infrastructure details: <ul style="list-style-type: none">• UPI - User Provided Infrastructure• Cloud platform independent (Bare-metal)	Aug 11, 2023 watsonx.data standalone login and usage This video will show you how to find your watsonx.data credentials, url login information and how to open the lakehouse portal.
---	---	--

7. Scroll down until you find the **watsonx.ai for custom models (Student ID version)** tile (highlighted in red below). You **MUST** use this pattern.

Watsonx AI with a shared WML standard plan for custom model imports. Only use this environment if you require this capability.

Services on IBM Cloud including WS, COS, DB2, WML.

****Note: This reservation creates Student ID credentials and does no...

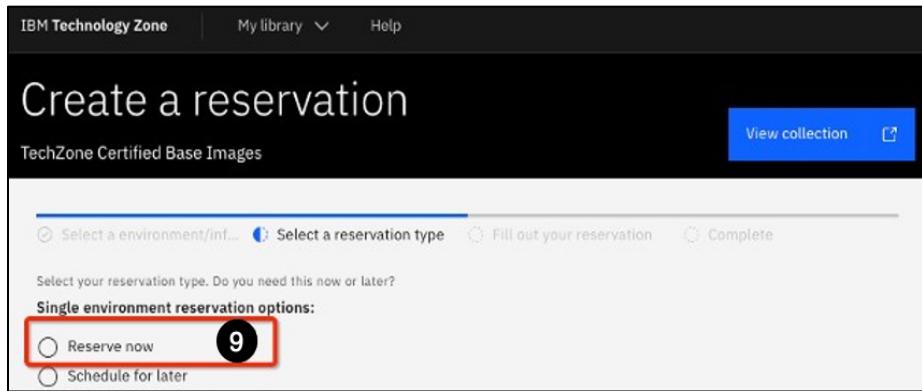
- Hover over the **IBM Cloud environment Reserve** button at the bottom and the text will change to **Reserve it**. Click **Reserve it**.

Watsonx AI with a shared WML standard plan for custom model imports. Only use this environment if you require this capability.

Services on IBM Cloud including WS, COS, DB2, WML.

****Note: This reservation creates Student ID credentials and does no...

- On the **Create a reservation** page, select **Reserve now**. If you are not ready to use the environment in the next hour or so, you can select the **Schedule for later** radio button (the steps that follow are the same).



10. You can leave the **Name** as is, or change the **Name** of the reservation to <your id> watsonx.ai for custom models (Student ID version).

11. Click the **Education** tile.

12. In the **Purpose description** field, enter Completing L3 Lab.

13. Select your **preferred Geography**. You can choose any one you wish. In the example here **AMERICAS** is chosen.

Name
felixwatsonx.ai for custom models (Student ID version) 10

Purpose

- Demo**
Deliver a client specific demonstration based on discovery with the client and aligns to the identified architecture. Automatically captures a Technical Sales Activity in IBM Sales Cloud on the Opportunity code provided.
- Pilot**
Rapid co-creation build that proves IBM technologies can deliver business value to clients' end users. Serves as a foundation to build a production solution. Automatically captures a Technical Sales Activity in IBM Sales Cloud on the Opportunity code provided.
- Education** 11
Gaining experience with specific technology, product, or solution.
- Test**
Need to test a specific function, configuration, or customization.

Please ensure to select the correct purpose as this can **NOT** be updated or changed after this reservation has been created. Review the [Reservation Duration Policy](#) to understand default durations allowed for specific infrastructures based on purpose.

Sales Opportunity number
Enter an opportunity number
Providing an [IBM Sales Cloud opportunity number](#), [Gainsight Relationship ID](#), or a [Project Work ID](#) will allow you to extend your reservation date.

Purpose description
12
Completing L3 Lab

What are you doing? Why do you need this? What are you trying to accomplish?

Preferred Geography
itz-watsonx - AMERICAS - us-south region - dal10 datacenter 13

14. Leave the **End date and time** as is.
15. Select **No** (should be the default) for installing Db2.
16. You can optionally provide some additional details in the **Notes** field if you wish.
17. On the right-hand side at the bottom, accept the **Terms and Conditions**, then click **Submit**.

The screenshot shows a reservation form with the following highlighted areas:

- Step 14:** The "End date and time" section, which includes fields for "Select a date" (03/11/2025), "Select a time" (11:36 AM), and "Time zone" (America/Toronto). A red box surrounds this entire section.
- Step 15:** The "Install DB2?" dropdown menu, which has "No" selected. A red box surrounds this field.
- Step 16:** The "Notes" text area, which is currently empty. A black circle surrounds the number 16 next to it.
- Step 17:** The "Submit" button, which is highlighted with a blue rectangle.

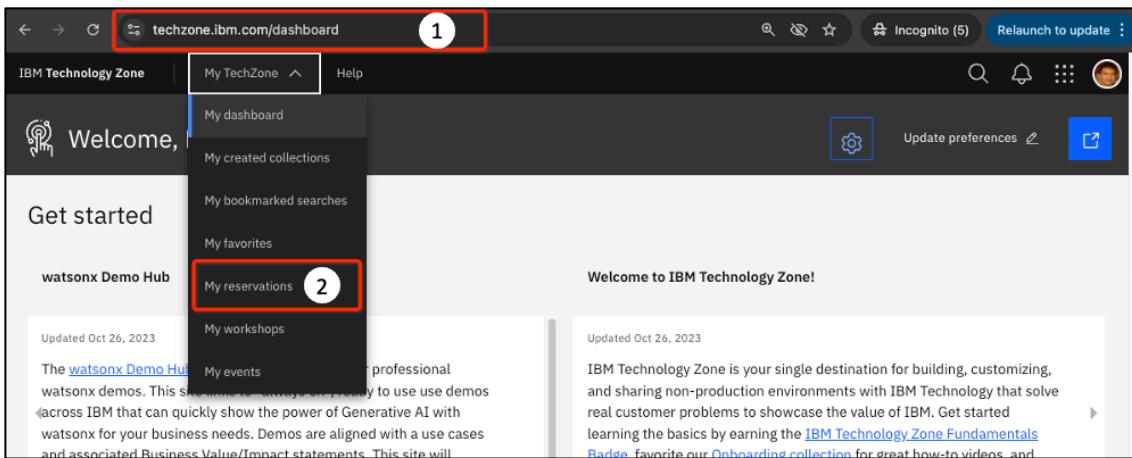
Other visible elements include a note about reservation policy: "Reservation policy: Recommended 6 months, but can be reserved up to 6 months on this reservation form. Extend later for null increments up to null total. Max time 6 months total." and a checkbox for accepting terms and conditions.

18. You will get an email sent to your IBMid email address letting you know that your reservation is being provisioned. Once your watsonx.ai instance provisioning is completed, you will receive a second email telling you that it is ready for use.
- Note:** Unlike other TechZone reservations, you do NOT need to accept an invitation to work with these accounts. You will be logging in via a “student login” described in the next section.

5.4.2 Logging into your TechZone environment

To log into the **Custom Foundation Model** TechZone environment, you require information available on your reservation. You **DO NOT** log in using your IBM ID.

1. Go to <http://techzone.ibm.com/dashboard>
2. Click **My TechZone**, then click **My reservations**.



3. The **My reservations** page opens. Look for the **watsonx.ai for custom models** tile (the **Password** is intentionally redacted in the image below).

You can copy the **Username** and **Password** and keep them handy, or you can always return to this page to get it.

In this example, the **Username** is **user_kzijc**.

My reservations

Search Filter by infrastructure Filter by status

Education
watsonx.ai for custom models (Student ID version)
Sep 25, 2024 7:25 AM
Mar 24, 2025 7:28 AM
Expires in: 179 days, 10 hours, 43 minutes
Extend limit: 0
Username user_kzljz
Password [REDACTED]
Status: Ready

Education
InstructLab opensource CLI
Sep 22, 2024 5:56 PM
Oct 6, 2024 6:09 PM
Expires in: 11 days, 1 hour, 13 minutes
Extend limit: 0
Username itzuser
Password USE SSH KEY
Status: Ready

Education
watsonx.ai/.governance SaaS
Sep 11, 2024 9:59 PM
Sep 25, 2024 10:05 PM
Expires in: 14 hours, 18 minutes
Extend limit: 1
Username felix@ca.ibm.com
Password <your ibm cloud password>
Status: Ready

4. Click on the tile to open its details.
5. Scroll down to the **Reservation Details** section. Click <https://cloud.ibm.com/logout> to log out of other sessions.

Reservation Details

First make sure you are logged out of IBM Cloud.
<https://cloud.ibm.com/logout>

6. Return to your browser tab with the reservation details. Click on the link provided for **IBM Cloud Login** (your link will be different from the example shown below).

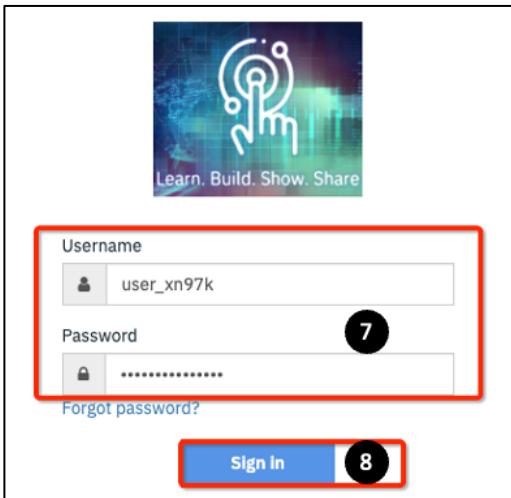
Reservation Details

First make sure you are logged out of IBM Cloud.
<https://cloud.ibm.com/logout>

IBM Cloud Login
<https://cloud.ibm.com/authorize/itzwatsonx14>

7. A new login panel appears. Use the **Username** and **Password** provided in your TechZone reservation to log in.

8. Click Sign in



9. You have now logged into the IBM Cloud using the Student ID. You can stay logged in, or simply log back in later using the same procedure.

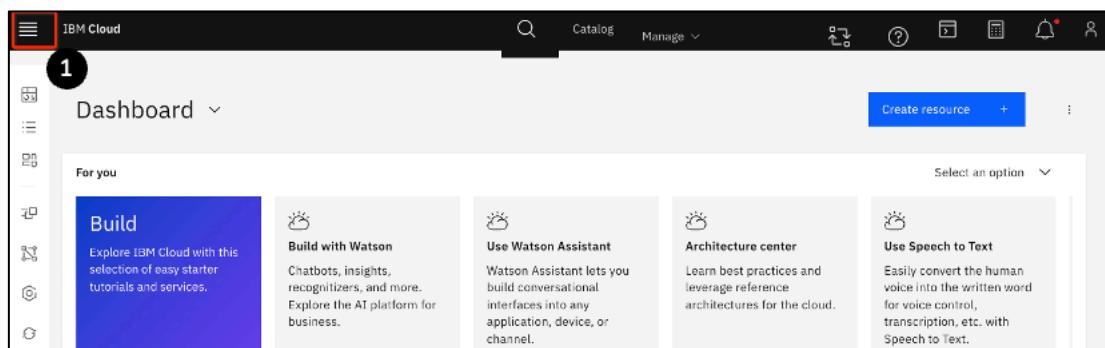
You must always be logged into your TechZone reservation in this manner for this lab.

5.4.3 Creating a project in watsonx.ai

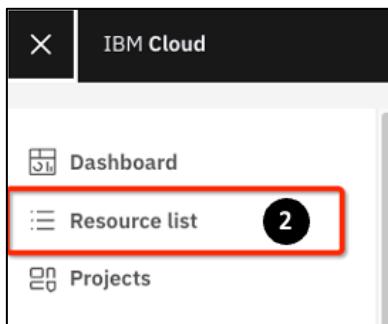
Now you will create a sandbox project that will include a Cloud Object Storage (COS) instance that you can use to upload your model.

If you have logged out of IBM Cloud (or it has timed out), you can log back in using the process in Section 5.4.2. Do NOT login with your IBM ID.

1. On the **Dashboard**, click on the main menu on the upper left.



2. Click **Resource list**



3. Expand the AI/Machine Learning section by clicking on it.

This screenshot shows the 'Resource list' page in the IBM Cloud interface. At the top, there's a title 'Resource list' and a 'Create resource' button. Below the title, there's a table header with columns: Name, Group, Location, Product, Status, and Tags. There are also several filter input fields. The main content area shows collapsed sections for Compute, Containers, Networking, Storage, Converged infrastructure, Enterprise applications, and Analytics. The 'AI / Machine Learning' section is expanded, showing one entry: 'itcos-110000f3jk-xn97k' with 'watsonxai-xn97k' under 'Group', 'Global' under 'Location', 'Cloud Object Storage' under 'Product', and 'Active' under 'Status'. A red box highlights the 'AI / Machine Learning' section, and a black circle with the number 3 is placed over the first item in the list. The background is white.

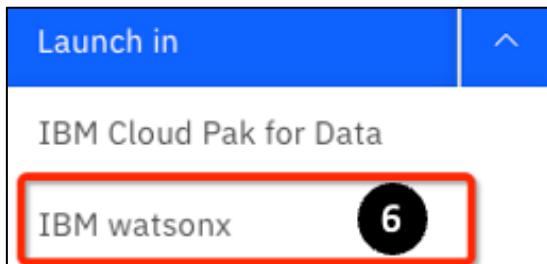
4. Click on the Watson Studio instance highlighted below (your entry will have a different itzws-*** label).

This screenshot shows a table of Watson Studio instances. The table has columns: Name, Group, Location, Product, Status, and Tags. There are four entries: 1. Watson Knowledge Catalog-itz, Group: Dallas, Product: IBM Knowledge Catal..., Status: Active. 2. Watson Machine Learning-itz, Group: Dallas, Product: Watson Machine Lear..., Status: Active. 3. Watson OpenScale-itz, Group: Dallas, Product: watsonx.governance, Status: Active. 4. itzws-110000f3jk-xn97k, Group: watsonxai-xn97k, Location: Dallas, Product: Watson Studio, Status: Active. A red box highlights the last row, and a black circle with the number 4 is placed over the 'Dallas' location cell. The background is white.

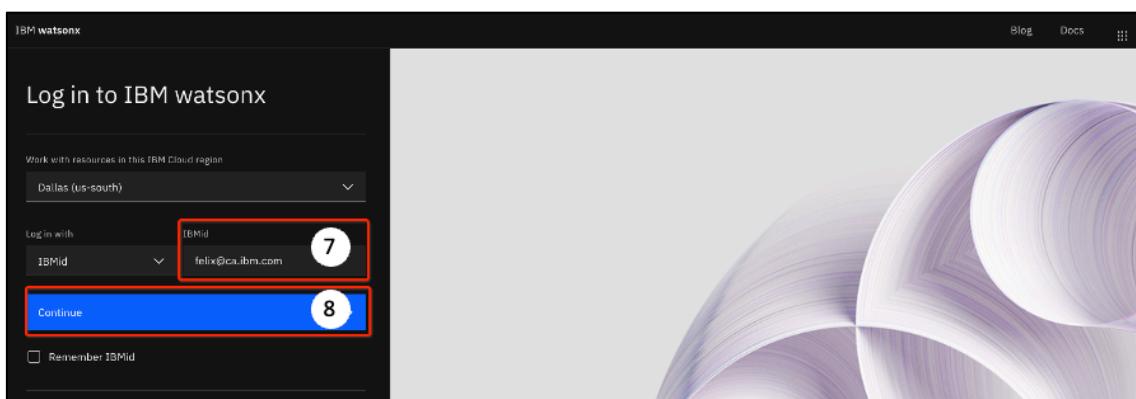
5. On the Watson Studio page, click to expand and see options for Launch in.



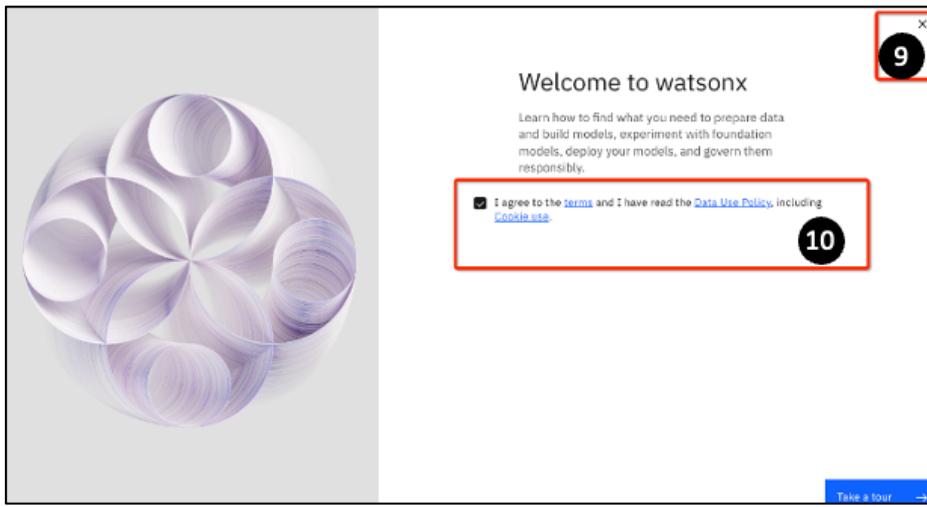
6. You will see 2 options: **IBM Clock Pak for Data** and **IBM watsonx**. Select **IBM watsonx**.



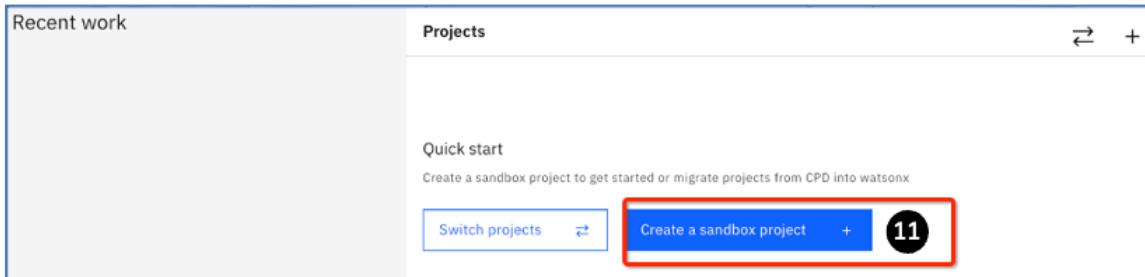
7. You may be asked to log into the IBM cloud. If so, provide your IBMid this time only. Note that you have already logged into your Student ID (**user_kzijc**). This is an additional step to get you into the watsonx.ai console.
8. Click **Continue**.



9. The **Welcome to watsonx** page may open, asking you to agree to terms. If it does, agree to its terms.
If TechZone assigns an account you have used before, your **Welcome to watsonx** page may not ask you to agree to the terms.
10. In either case, click **X** to close this window.



11. You are now in the watsonx.ai console. Scroll down to the **Projects** area and click **Create a sandbox project**.



12. A project called **notset's sandbox** is created. Note that you are using a Student ID, not your personal ID so the project would not have your name (or an **itz-watsonx** prefix like in other watsonx.ai L3 labs).

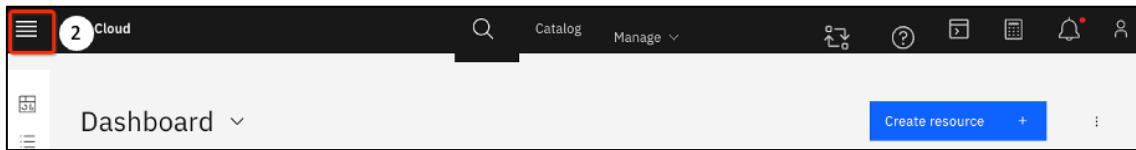
The sandbox project may take some time to create, so please be patient.

When you create the sandbox project, a Cloud Object Storage (COS) bucket is created. You will upload the model files into this bucket in the next section.

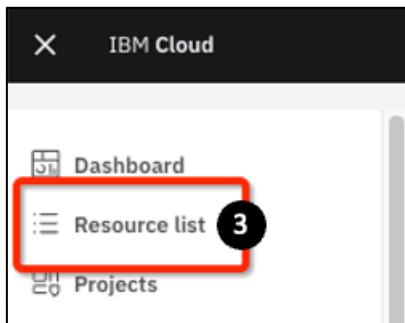
5.5 Uploading the model to Cloud Object Storage

Now you will upload the model files to the COS bucket associated with your project.

1. If you have logged out of IBM Cloud (or it has timed out), you log back in using the process in Section 5.4.2. Do NOT login with your IBM ID.
2. On the **Dashboard**, click on the main menu on the upper left.



3. Click **Resource list** on the slide-out.



4. Expand the **Storage** section by clicking on it.

A screenshot of the Resource list interface. The "Storage" section is expanded, showing one entry: "itzcos-110000f3jk-kzliz" (Name), "watsonxai-kzliz" (Group), "Global" (Location), "Cloud Object Storage" (Product), and "Active" (Status). A red box highlights this entry, and a black circle containing the number "4" is placed over the "Storage" section header.

The entry with **Name** similar to **itzcos-*-*** (as shown above) is the bucket available to you for this lab. In this example, it is **itzcos-110000f3jk-kzliz**. Click on this **itzcos-*-*** instance.

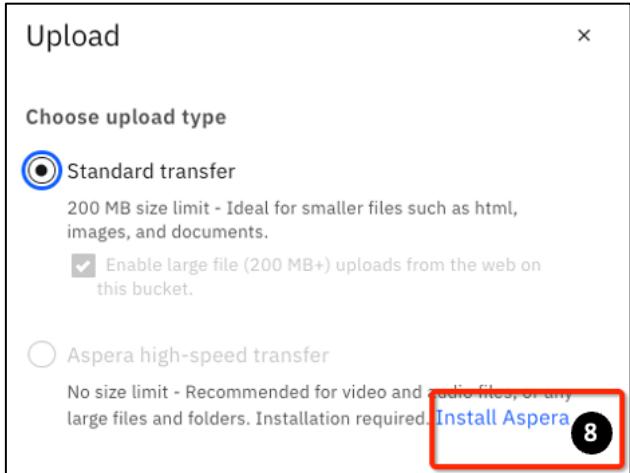
- The **itzcos-*-*** instance page opens, showing the bucket associated with it. There might be multiple buckets, but you should see an entry with a name like **notsetsandbox-donot-delete-***

Click on that bucket to open it.

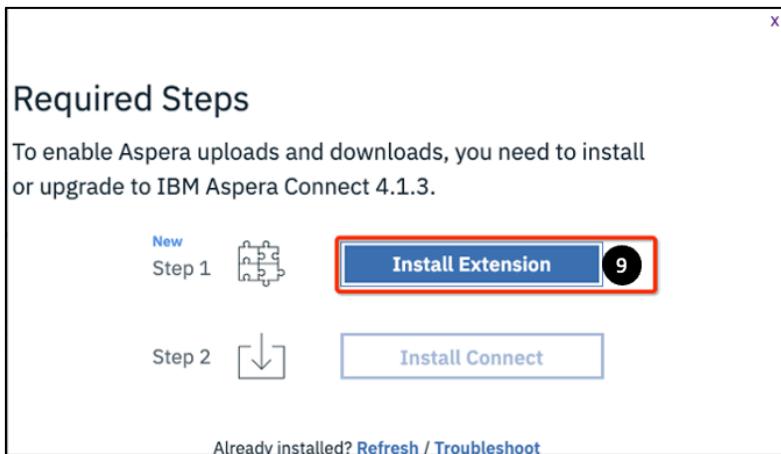
Name	Public access	Location	Storage class	Created
notsetssandbox-donotdelete-pr-nccbwx8s...	No	United States - Dallas (us-south)	Standard	2024-09-25 8:22 AM

- Click **Upload** on the right.

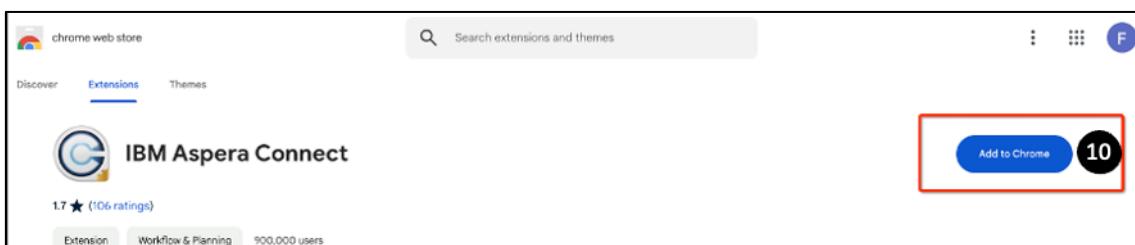
- The **Upload** slide-out opens. Notice how the **Standard transfer** is proposed for file transfers within the 200 MB limit. For this lab, since the **flan-t5-base** model is bigger you should use the **Aspera high-speed transfer** option. If you can select this option, select it and skip to Step 21
- Click the **Install Aspera** hotlink associated with the **Aspera high-speed transfer** option.



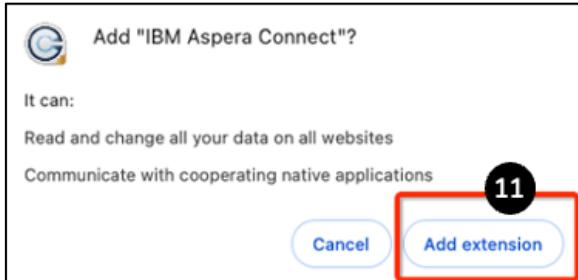
9. The Required Steps page appears. Click **Install Extension**. If the Extension is already installed, the **Install Connect** button is highlighted instead. If so, skip to Step 13.



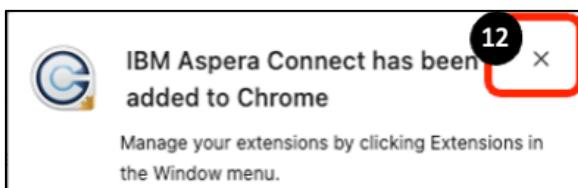
10. The Chrome web store page opens. Click **Add to Chrome**.



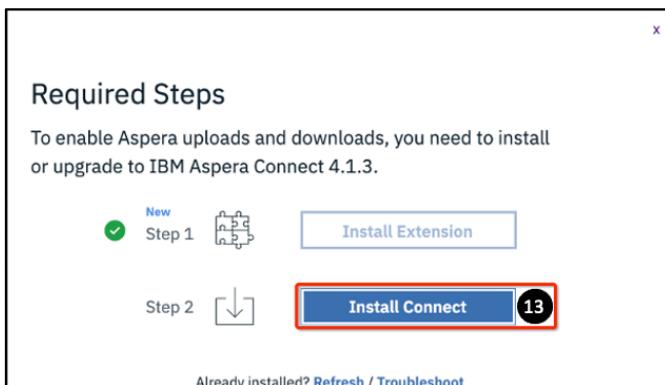
11. On the Add "IBM Aspera Connect" dialog box, Click **Add extension**.



12. You will see a message that Aspera has been successfully installed. Click **x** to dismiss this window.



13. You are back to the **Required Steps** page. Click **Install Connect**.



14. You will see the **IBM Aspera Connect Installer** downloaded. Depending on the OS, you may see different files. In the case of Mac OS, you see a file like **ibm-aspera-connect_4.1.3.93_macOS_x86_64.dmg** (version number 4.1.3.93 may differ).

15. Navigate to the **Downloads** (or your equivalent) folder and click on the file.

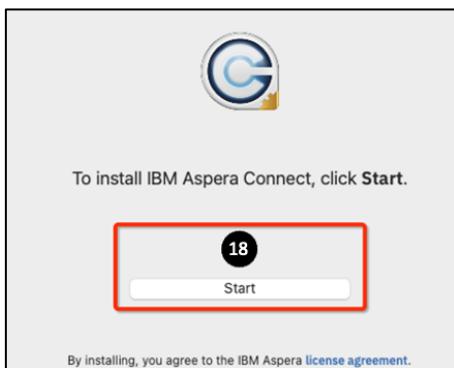
16. A folder opens up. Open the **IBM Aspera Connect Installer** icon.



17. On Mac OS, you will see a warning panel. Click Open.



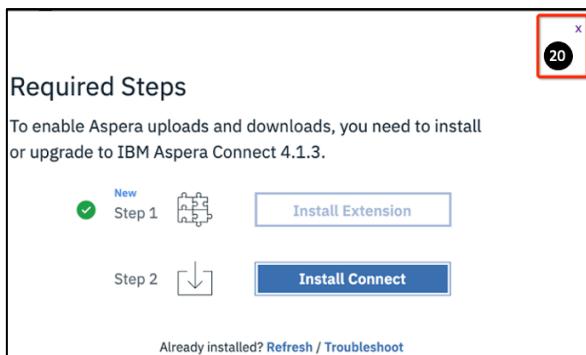
18. The To install IBM Aspera Connect page opens. Click Start.



19. When installation is completed, click Close.



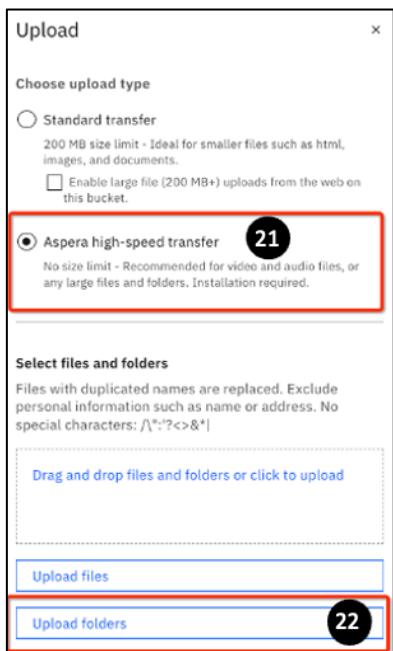
20. Click X to close the **Required Steps** window (if still open).



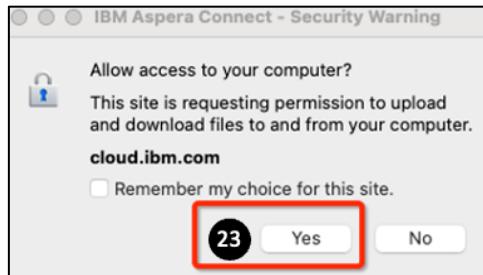
21. On the watsonx.ai console, click **Upload** again. Now, the **Aspera high-speed transfer** option is available to you. Select **Aspera high-speed-transfer**.

Note: if you have successfully installed the plug-in and the Aspera image but the **Aspera high-speed transfer** option is still not available, the plug-in may be blocked by the browser.

22. Click **Upload folders**.

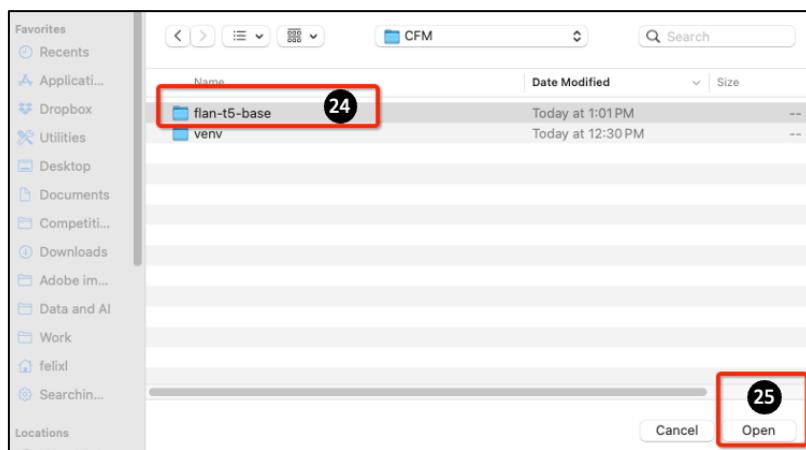


23. You will be asked if you **Allow access to your computer**. Click **Yes**.



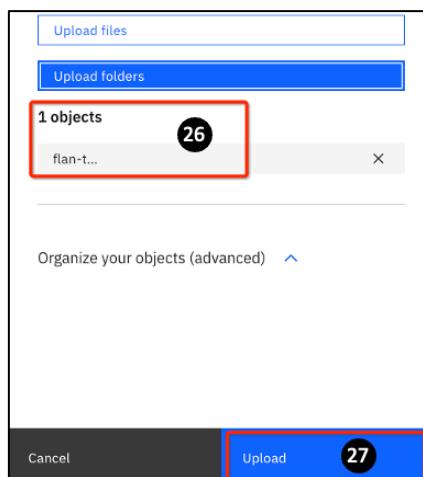
24. Navigate to the CFM folder (/Users/felixl/Downloads/CFM in this example) and select the flan-t5-base folder.

25. Click Open to upload.

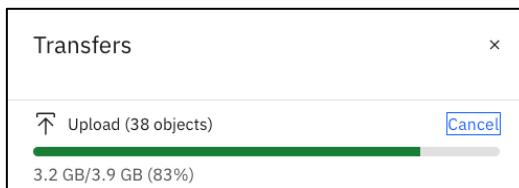


26. You will see “1 objects” as shown below - the folder flan-t5-base.

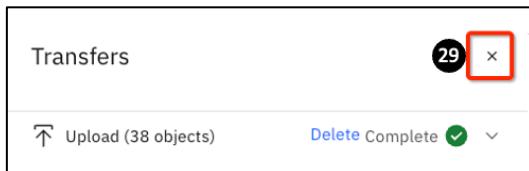
27. Click Upload.



28. You will see the following message. Aspera provides high-speed transfer, so this should be quick. It does depend on your connection speed to IBM Cloud.



29. When the upload is completed, you will see the image below. Click X to close the **Transfers** panel.



30. You will see a lot of files now in your COS bucket. You will have multiple pages.

Object name	Archived	Size	Last modified
flan-t5-base/		0 bytes	2024-10-03 2:53 PM
flan-t5-base/.gitattributes		1.4 KB	2024-10-03 2:53 PM
flan-t5-base/.huggingface/.gitignore		1 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/.gitattributes.lock		0 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/.gitattributes.metadata		100 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/README.md.lock		0 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/README.md.metadata		100 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/config.json.lock		0 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/config.json.metadata		100 bytes	2024-10-03 2:53 PM
flan-t5-base/.huggingface/download/flax_model.msgpack.lock		0 bytes	2024-10-03 2:53 PM

You are now ready to connect to this from your watsonx.ai instance. To do so, you need additional information on the COS bucket.

31. Click the Configuration tab.

The screenshot shows the IBM Cloud Object Storage interface for a bucket named 'notsetssandbox-donotdelete-pr-ncbwx8sra3jygk'. The 'Configuration' tab is highlighted with a red box and the number 31. The interface includes tabs for 'Objects', 'Configuration', and 'Permissions'. A message at the top says, 'If you're seeing more usage than expected, versions count towards your usage or you may have incomplete uploads Learn more'. Below is a file list with a single entry: 'medical_summarization/README.md' (44.1 KB, 2024-09-25 8:57 AM). There are 'Upload' and 'Actions...' buttons at the top right.

32. Scroll down to the **Endpoints** section and find the **Public** endpoint to this bucket. Click the **Copy** button on the right of the entry to put the information on the clipboard. Save this information somewhere you can retrieve later.

The screenshot shows the 'Endpoints' section in the IBM Cloud Object Storage interface. It includes sections for 'Regular Endpoints' (with a note about REST API requests) and 'Private' (with a note about IBM cloud services). The 'Public' section is highlighted with a red box and the number 32. It contains a note about pointing to external services or Cloud Foundry applications. Below is a text input field with the value 's3.us-south.cloud-object-storage.appdomain.cloud'. A 'Copy' button is visible to the right of the input field. Other options like 'Direct' are also listed.

In this example the public endpoint is
s3.us-south.cloud-object-storage.appdomain.cloud

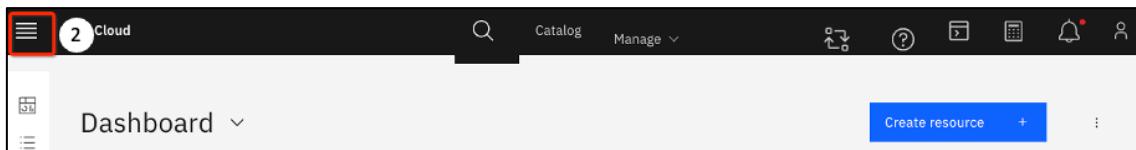
5.6 Creating a connection to COS from watsonx.ai

You have uploaded a model into the COS instance associated with your TechZone reservation. Now you need to create a connection from watsonx.ai to access the model's files.

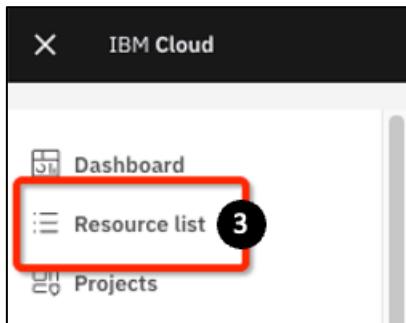
5.6.1 Getting your Access key and Secret key

When you connect to your COS instance to access your model files for Custom Foundation Model, you need to connect using **Access key and Secret key**.

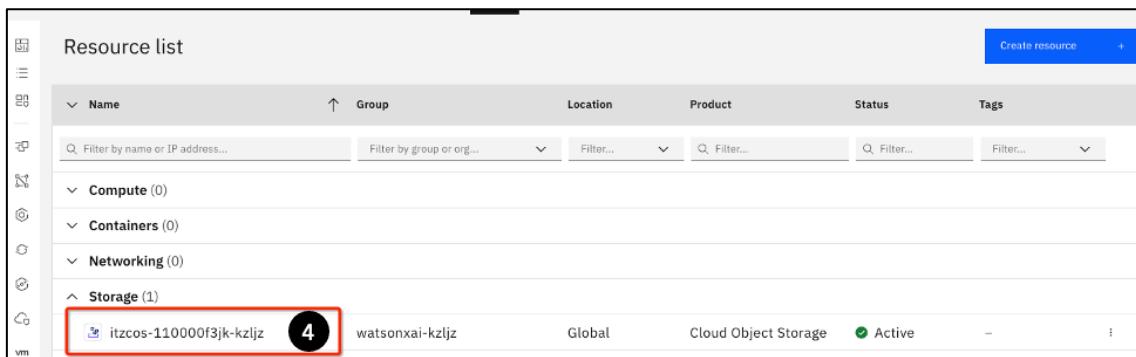
1. If you have logged out of IBM Cloud (or it has timed out), you can log back in using the process in Section 5.4.2. Do NOT login with your IBM ID.
2. On the **Dashboard**, click on the main menu on the upper left.



3. Click **Resource list** on the slide-out



4. Expand the **Storage** section and click on the COS instance with name like **itzcos-***.



5. Select the **Service credential** tab (if not already selected)

6. You may already have several keys. Click **New Credential** to create a new set.

7. The **Create Credentials** page opens. Use the following values:

- For **Name**: use **CFM_demo**
- For **Role**: select **Manager**
- For **Include HMAC Credential**: set it **On** by clicking on the button and slide it to **On**.

8. Click **Add**.

9. Scroll down (if necessary) to find the **CFM_demo** key and click it to get its details.
10. Click the **copy** icon to place that information on your clipboard. Paste it in a file so you can easily retrieve later.



11. The content of this sample key is pasted below. Note the values of **access_key_id** and **secret_access_key** (you need the values within the quotation marks). In this example:

- a. **access_key_id** is 0912c828a1704903884590e9eb46f1d3
- b. **secret_access_key** is 11e50e250e0440ba40b46add3cfbc6e8146dfb5f04e0570e

```
{
  "apikey": "yb1y0C5P07m_dtF0aPe-L3oz1pCPy08nyQ4k7TC1op_0",
  "cos_hmac_keys": {
    "access_key_id": "0912c828a1704903884590e9eb46f1d3",
    "secret_access_key": "11e50e250e0440ba40b46add3cfbc6e8146dfb5f04e0570e"
  },
  "endpoints": "https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints",
  "iam_apikey_description": "Auto-generated for key crn:v1:bluemix:public:cloud-object-storage:global:a/baf3554ebf3c4225ac4c4e9efb79516b:9d8a528f-da3a-4469-b30f-d1b80e944671:resource-key:0912c828-a170-4903-8845-90e9eb46f1d3",
  "iam_apikey_id": "ApiKey-b6c65d0c-fbca-4939-802b-b1e001bf4f32",
  "iam_apikey_name": "CFM_demo",
  "iam_role_crn": "crn:v1:bluemix:public:iam::::serviceRole:Manager",
  "iam_serviceid_crn": "crn:v1:bluemix:public:iam-identity::a/baf3554ebf3c4225ac4c4e9efb79516b::serviceid:ServiceId-a9e0b637-f363-4193-82b6-9f4310216b61",
  "resource_instance_id": "crn:v1:bluemix:public:cloud-object-storage:global:a/baf3554ebf3c4225ac4c4e9efb79516b:9d8a528f-da3a-4469-b30f-d1b80e944671::"
}
```

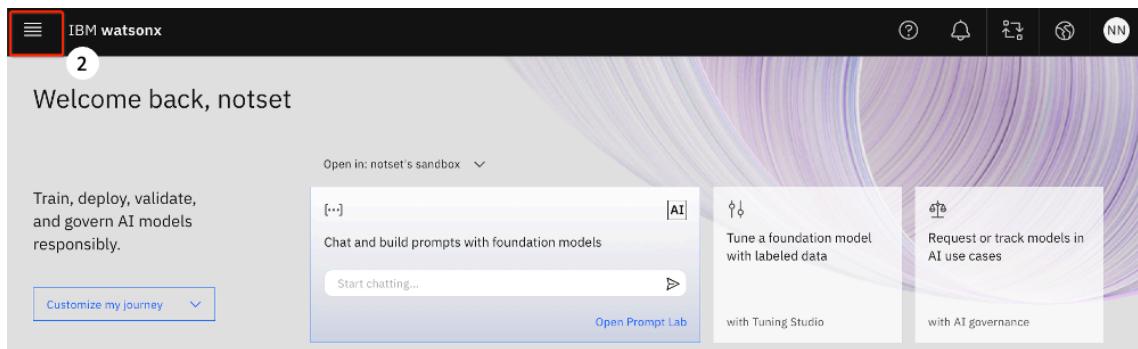
You will use these values when you create the connection to the COS instance.

5.6.2 Creating a Deployment Space and connection to COS using Access key and Secret key

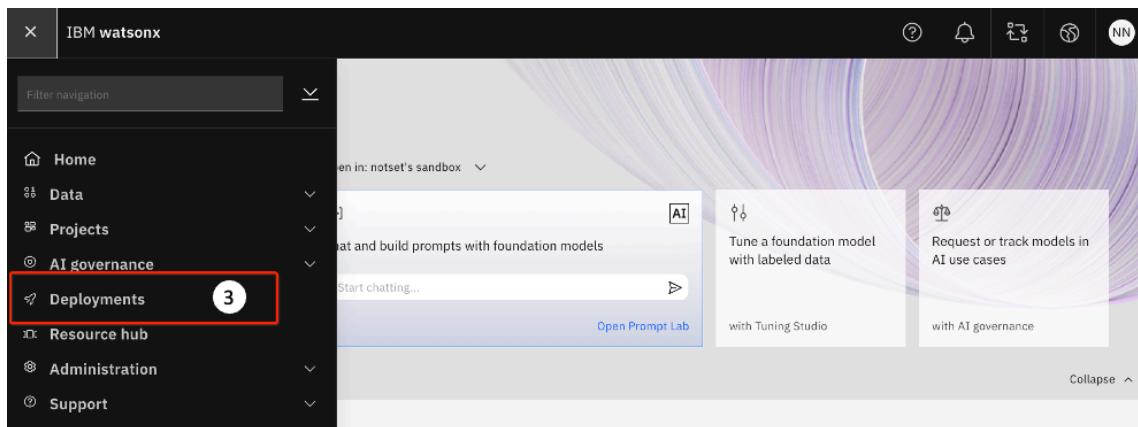
You can deploy a model to a project or to a deployment space. A deployment space is not associated with a project. Assets from different projects can be deployed in the same space. In this section, you will

- Create a deployment space
- Define a connection from your deployment space to your COS instance.

1. You can log back into watsonx.ai using Steps 1-8 of Section 5.4.3.
2. The watsonx.ai main console opens. Click the main menu on the upper left.



3. On the slide-out, click **Deployments**.



4. The **Deployments** page opens. Click the **New deployment space** button.

The screenshot shows the IBM WatsonX interface. At the top, there's a navigation bar with icons for help, notifications, and user profile. Below it, the title 'IBM WatsonX' is followed by 'Deployments' and '0 spaces'. A blue button labeled 'New deployment space' with a counter '4+' is prominently displayed. Below these, there are tabs for 'Activity' and 'Spaces', with 'Activity' being the active tab.

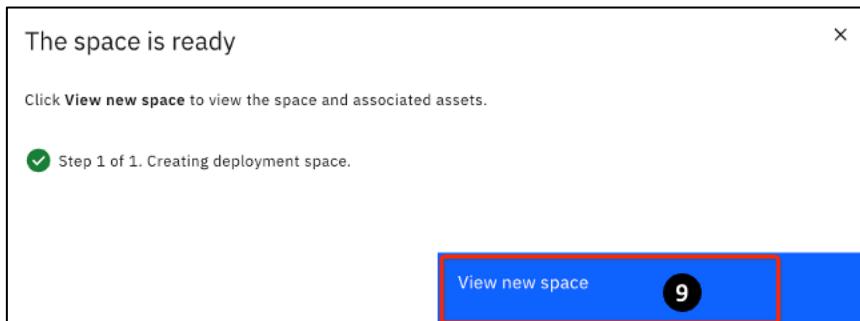
5. The **Create a deployment space** page opens. Enter **CFM_space** as **Name**.
6. Select **Development** for **Deployment stage**.
7. The **Select storage service** field should be automatically populated if you only have one COS instance. If you have more than one, select the one identified in Step 4 of Section 5.5 (with a name like **itzcos-***).
8. Leave everything else as is. Click **Create**.

The screenshot shows the 'Create a deployment space' form. It has two main sections: 'Define details' and 'Select services'. In the 'Define details' section, the 'Name' field is filled with 'CFM_space' (step 5). In the 'Select services' section, the 'Select storage service' dropdown is set to 'itzcos-110000f3jk-kzljz' (step 7). Both sections are highlighted with red boxes. At the bottom right, there are 'Cancel' and 'Create' buttons, with the 'Create' button being highlighted with a red box (step 8).

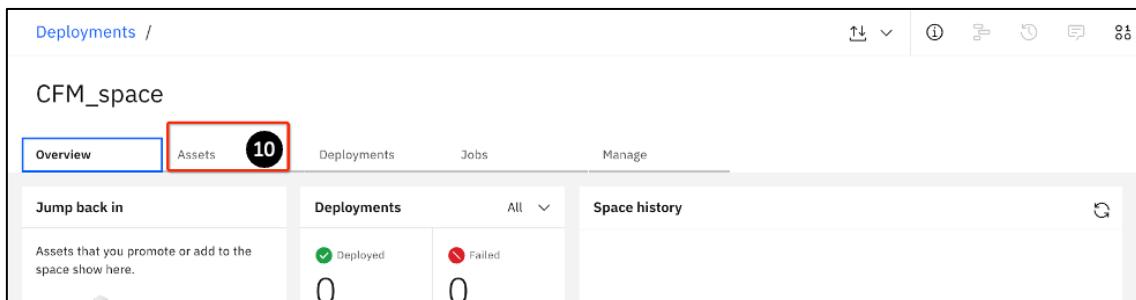
9. It takes some time to create a deployment space. You will see this:



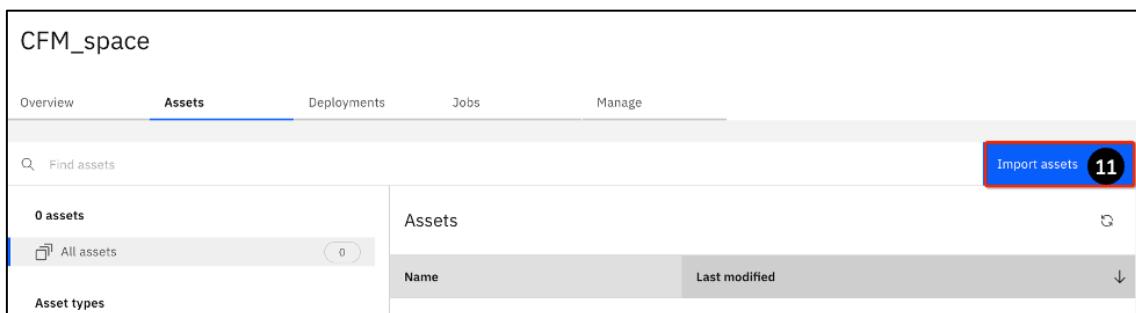
When the process is completed, the **View new space** button becomes active. Click **View new space**.



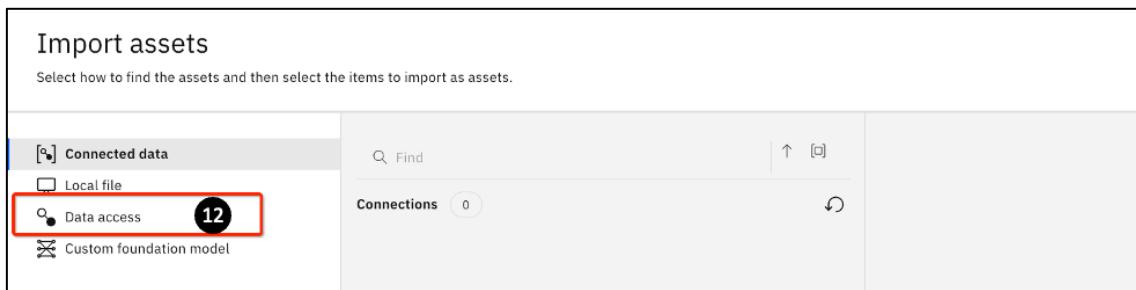
10. The **CFM_space** page opens. Click the **Assets** tab.



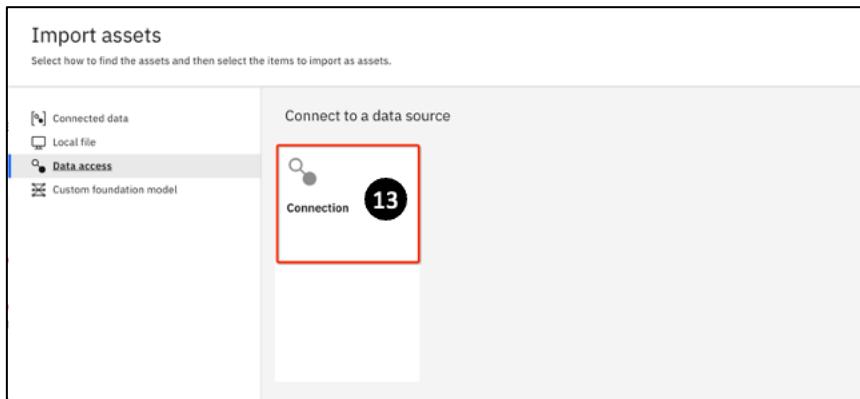
11. On the **Assets** tab, click **Import assets**.



12. The **Import assets** page opens. The first thing you need to do is to define data access to your COS instance where the model files have been uploaded. Click **Data access** on the left.



13. Click the Connection tile.

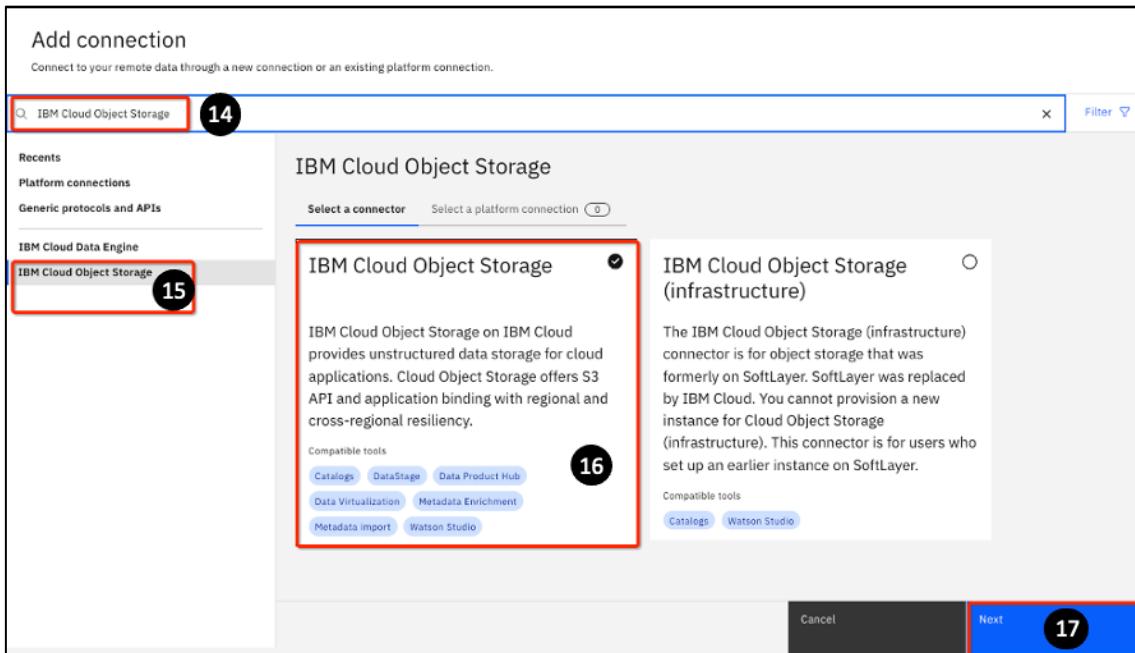


14. The Add connection page opens. If you do not see **IBM Cloud Object Storage**, go to the **Search** field at the top left and enter **IBM Cloud Object Storage**. The list will dynamically filter.

15. Click on **IBM Cloud Object Storage**.

16. You may see 2 tiles: **IBM Cloud Object Storage** and **IBM Cloud Object Storage (infrastructure)**. Select **IBM Cloud Object Storage** (if not already selected).

17. Click **Next**.



18. On the Connect to a data source page:

- a. **DO NOT** click **Select instance**. If you do, the connection will fail.
- b. For **Name**: enter **CFM_connect**
- c. For **Login URL**: enter the value you obtained in Step 32 of Section 5.5. For example: **s3.us-south.cloud-object-storage.appdomain.cloud**
- d. For **Credentials**, click **Personal**.
- e. For the **Authentication method**, select **Access key and Secret key**.

Connect to a data source: IBM Cloud Object Storage
Define the details to create a connection asset.

Test connection

Connection overview

Connection details

Credentials
Certificates
Location and sovereignty

Integrated instance ⓘ
 Select instance ⓘ **18a**

Name (required) **CFM_connect** **18b**

Description
Connection description

Connection details

Bucket ⓘ

Login URL (required) ⓘ
S3 us-south.cloud-object-storage.appdomain.cloud **18c**

Credentials

Credential setting ⓘ
 Personal **Shared** **18d**

Each user enters their own credentials for accessing data.

Authentication method (required) ⓘ
Access key and Secret key **18e**

19. Two additional fields will appear when you select **Access key and Secret key**. Enter the values you received for your **CFM_demo** in Step 11 of Section 5.6.1. In this example:

Access key: 0912c828a1704903884590e9eb46f1d3

Secret key: 11e50e250e0440ba40b46add3cfbc6e8146dfb5f04e0570e

20. Click **Test Connection**.

Connect to a data source: IBM Cloud Object Storage
Define the details to create a connection asset.

Test connection **20**

Connection overview

Connection details

Credentials
Certificates
Location and sovereignty

Credentials

Credential setting ⓘ
 Personal **Shared**

Each user enters their own credentials for accessing data.

Authentication method (required) ⓘ
Access key and Secret key **19**

Access key (required) ⓘ
0912c828a1704903884590e9eb46f1d3

Secret key (required) ⓘ
11e50e250e0440ba40b46add3cfbc6e8146dfb5f04e0570e

You should see a successful test result:

Connect to a data source: IBM Cloud Object Storage

Define the details to create a connection asset.

Connection overview Connection details

✓ The test was successful. Click Create to save the connection information.

[Test connection](#)

21. Click **Create**.

Connect to a data source: IBM Cloud Object Storage

Define the details to create a connection asset.

Connection overview Connection details

Credentials

Access key and Secret key

Access key (required) (1)
0912c828a1704903884590e9eb46f1d3

Secret key (required) (1)
11e50e250e0440ba40b46add3cfbc6e8146dfb5

Certificates

SSL certificate (1)

Validate the SSL certificate (1)

Location and sovereignty

Select the location and sovereignty of this source. If you don't set these properties, data location rules might deny access to this source and its connected data assets.
[Learn more](#)

Location (1)

[Cancel](#) [Back](#) **Create** 21

22. You will get a warning. You can ignore it and click **Create**.

Create connection without setting location and sovereignty?

If you don't set the location and sovereignty, data location rules might deny access to this source and its connected data assets.

Don't show me this message again.

[Cancel](#) **Create** 2

23. On the **CFM_space**, the newly created **CFM_connect** asset appears.

The screenshot shows the 'Assets' tab in the 'CFM_space' deployment space. The interface includes a navigation bar with 'Overview', 'Assets' (selected), 'Deployments', 'Jobs', and 'Manage'. Below the navigation is a search bar labeled 'Find assets' and a blue 'Import assets' button. On the left, there's a sidebar with '1 asset' and 'Asset types' sections. The main area is titled 'Assets' and displays a table with one row. The table has columns for 'Name' (with a value of 'CFM_connect') and 'Last modified' (32 minutes ago). A red box highlights the 'Name' column of the first asset. A black circle with the number '23' is overlaid on the right side of the table.

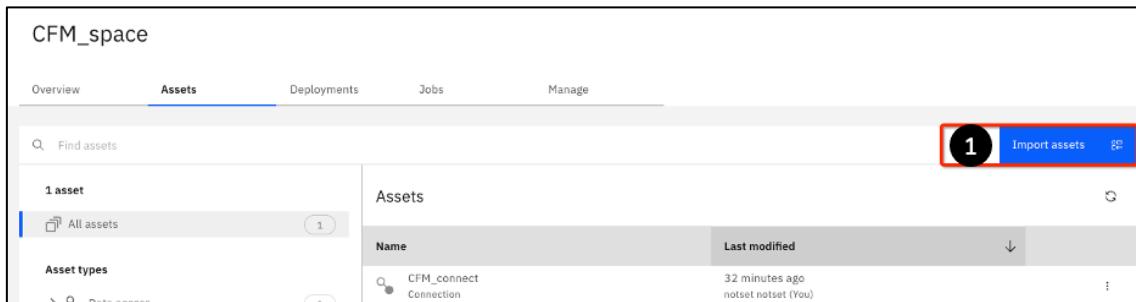
Name	Last modified
CFM_connect	32 minutes ago not yet tested (You)

You have created a deployment space and defined a connection to the COS instance in this section.

5.7 Creating a model asset

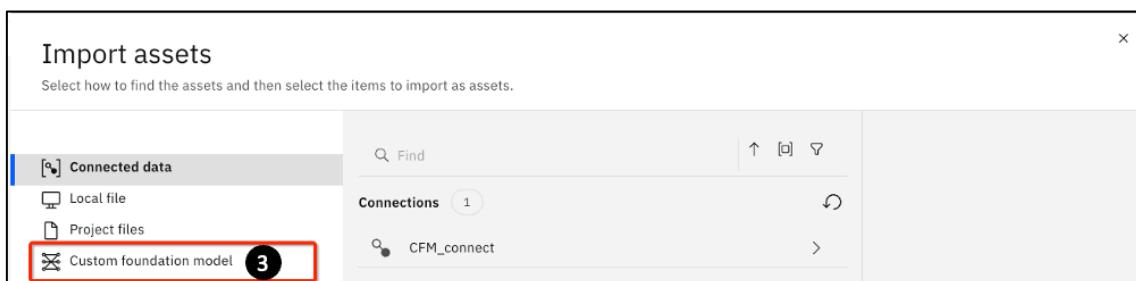
In this section, you import your model asset into the newly created deployment space.

1. On the **CFM_space Assets** tab, click **Import assets**.



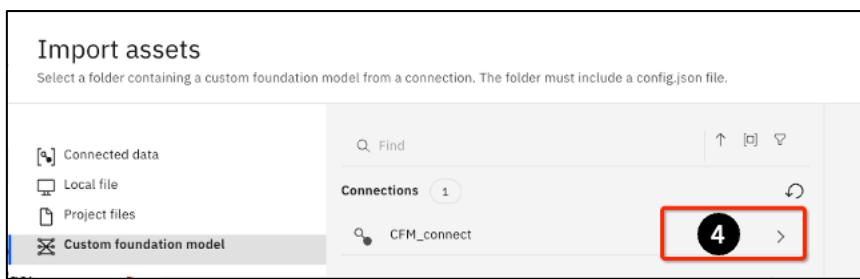
The screenshot shows the 'Assets' tab of the 'CFM_space' interface. At the top right, there is a blue button labeled 'Import assets' with a red box around it and the number '1' in a black circle. Below the button, there is a search bar labeled 'Find assets' and a section titled 'Assets' with a table. The table has columns for 'Name' and 'Last modified'. One row is visible, showing 'CFM_connect' under 'Name' and '32 minutes ago' under 'Last modified'. The 'Asset types' dropdown menu is open, showing options like 'All assets' and 'Data sources'.

2. The **Import assets** page opens (probably to the default **Connected data** menu item). You should see the **CFM_connect** connection.
3. Click the **Custom foundation model** menu item on the left.



The screenshot shows the 'Import assets' page. On the left, there is a sidebar with menu items: 'Connected data' (highlighted with a red box and the number '3'), 'Local file', 'Project files', and 'Custom foundation model'. In the main area, there is a search bar labeled 'Find' and a 'Connections' section. The 'CFM_connect' connection is listed in the connections list. A red box highlights the 'Custom foundation model' menu item.

4. You will see the **CFM_connect** connection created in Step 18 of Section 5.2.6. Click on it to expand.



The screenshot shows the 'Import assets' page with the 'Custom foundation model' menu item selected. In the main area, the 'Connections' section shows the 'CFM_connect' connection. To the right of the connection name, there is a circular arrow icon and a right-pointing arrow icon, both enclosed in a red box with the number '4'. A red box also highlights the 'CFM_connect' connection in the list.

5. You should see the **CFM_connect** COS bucket. If you have more than one bucket, ensure you select the correct one (identified in Step 4 of Section 5.5). Click on it to expand.

Import assets
Select how to find the assets and then select the items to import as assets.

[?] Connected data Find Connections 1
 Local file Data access Custom foundation model

CFM_connect 2
 22f4875e-ebe1-4a0f-a9a2-c564140f2618
 notsetssandbox-donotdelete-pr-ncbw...

5

6. You may need to scroll right to see the details. You should see the content of the bucket: the folder **flan-t5-base**. Click on this folder.

Import assets
Select a folder containing a custom foundation model from a connection. The folder must include a config.json file.

[?] Connected data Find Find
 Local file Data access Custom foundation model

CFM_connect 2
 22f4875e-ebe1-4a0f-a9a2-c564140f2618
 notsetssandbox-donotdelete-pr-ncbw...

notsetssandbox-donotdelete-pr-ncbw... 1
 flan-t5-base

6

7. The folder's content is shown. Click **Next** to import the entire folder.

Import assets
Select a folder containing a custom foundation model from a connection. The folder must include a config.json file.

[?] Connected data Find Find
 Local file Data access Custom foundation model

netsetssandbox-donotdelete-pr-ncbw... 1
 flan-t5-base

flan-t5-base 14
 .distributed
 .buggingface
 config.json
 flan_model.msgpack
 generation_config.json
 model.safetensors
 model->google->flan-t5-base
 pytorch_model.bin
 README.md
 specific_token_map.json
 specmodel

Selected assets (1/1)
 All asset details must load before final selection.
 flan-t5-base
 Type: Folder
 Path: /netsetssandbox-donotdelete-pr-ncbw.../flan-t5-base
 Connection name: CFM_connect
 Connector: IBM Cloud Object Storage (ibmcloudobjectstorage)

Cancel Back Next 7

8. The **Import a custom foundation model** page opens. Note that some values are filled in for you already:

Architecture: t5

Name: flan-t5-base – leave it as is for the exercise. You may want to change the model name in a Proof of Experience (PoX) or other activities with a client to make it easier to remember.

9. Optionally, you can add tags – leave this empty for this exercise.

10. Note the **Model deployment parameters** at the bottom of the page. You can update these by clicking on the associated pencil icon to the right. For this exercise, simply use the default values provided.

11. Click **Import**.

Import a custom foundation model

Define details

Architecture

t5

flan-t5-base

Description

What's the purpose of this model?

Tags

Add tags to make assets easier to find.

Find or create tags

Model deployment parameters

Display name *	Name *	Data type *	Enum	Minimum value	Maximum value	Default value
Data type	dtype	string	float16, bfloat16		float16	
Max batch size	max_batch_size	number		1	1000000000000000	16
Max concurrent requests	max_concurrent_requests	number		1	1000000000000000	1024

Cancel

Back

Import

11

12. The model page for **flan-t5-base** opens. Note the **About this asset** information. You can provide changes/updates (such as adding a description) if you want. For now, leave this as is.

Deployments / CFM_space / flan-t5-base

Deployments Model details

New deployment

Name	Type	Status	Tags	Last modified

This asset doesn't have any deployments yet
Use the New Deployment button to create a deployment for this asset.

About this asset

Name flan-t5-base

Description No description provided.

Asset Details

Type: custom_foundation_model_1.0

Model ID: 91a4c05f-92bf-42...

Software specification: walsone-cfn-caikit-1.0

Tags

Add tags to make assets easier to find.

Last modified 1 second ago by noseti noseti

Created on Oct 3, 2024 by noseti noseti

12

13. The model asset needs to be deployed before it is usable. You will do it in the next section. Do NOT close this window.

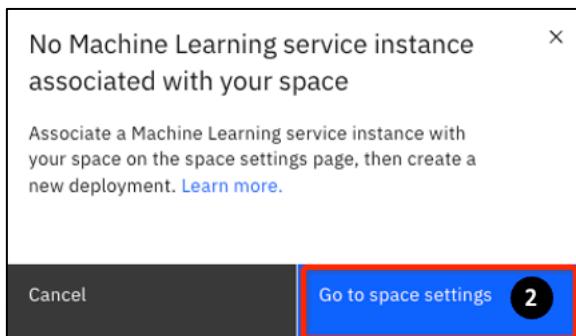
5.8 Deploying a model asset

Now, you will deploy the model that you have imported. This is required before the model can be used in watsonx.ai. Without this step, your model is basically just a set of files in a COS bucket. To deploy your model, you need to first create a user API key.

1. Click **New Deployment** on the flan-t5-base asset page.

The screenshot shows the 'Deployments / CFM_space / flan-t5-base' page. At the top, there are tabs for 'Deployments' and 'Model details'. Below the tabs is a search bar and a table with columns: Name, Type, Status, Tags, and Last modified. A red box highlights the 'New deployment' button in the top right corner of the table area. To the right of the table, there's a sidebar titled 'About this asset' with fields for Name (medical_summarization) and Description (No description provided).

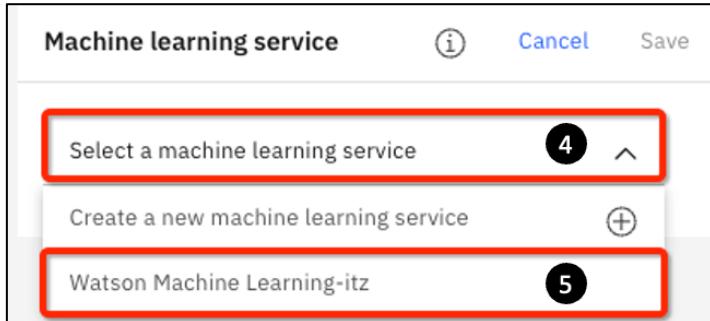
2. You may get the following warning message about Machine Learning Service instance. If so, click **Go to space settings**. If not, skip to Step 10 below.



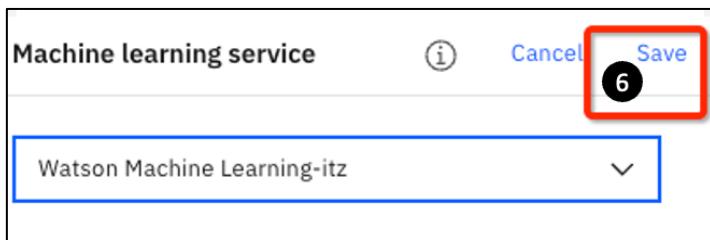
3. The CFM_space page opens. Click **Associate instance** from Machine learning service field on the right.

The screenshot shows the 'Deployments / CFM_space' page. On the left, there's a sidebar with 'General', 'Access control', 'Environments', and 'Resource usage'. The main area has tabs for 'Overview', 'Assets', 'Deployments', 'Jobs', and 'Manage'. Under 'Manage', there's a 'General' section with 'Details' and 'Storage' tabs. In the 'Machine learning service' section, there's a button labeled 'Associate instance' which is highlighted with a red box. Other fields in this section include 'Name' (N/A), 'Bucket' (2f4875e-ebe1-4a0f-a9a2-c564140f2618), and 'Manage in IBM Cloud'.

4. Expand **Select a machine learning service** to show the list.
5. Select (you might have more than one) the **Watson Machine Learning-itz** service.



6. Click **Save**.



7. You are on the **Manage** tab. Click the **Assets** tab.

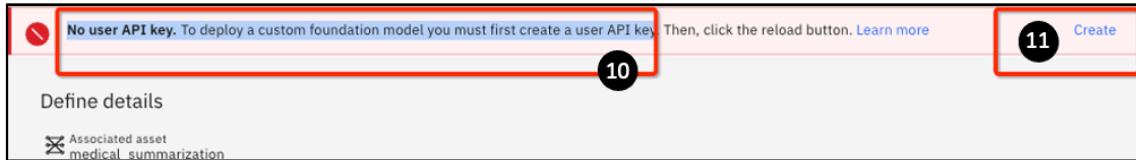
8. Click on the **flan-t5-base** asset.

Name	Last modified
flan-t5-base Custom foundation model	14 minutes ago notset/notset (You)
CFM_connect Connection	5 hours ago notset/notset (You)

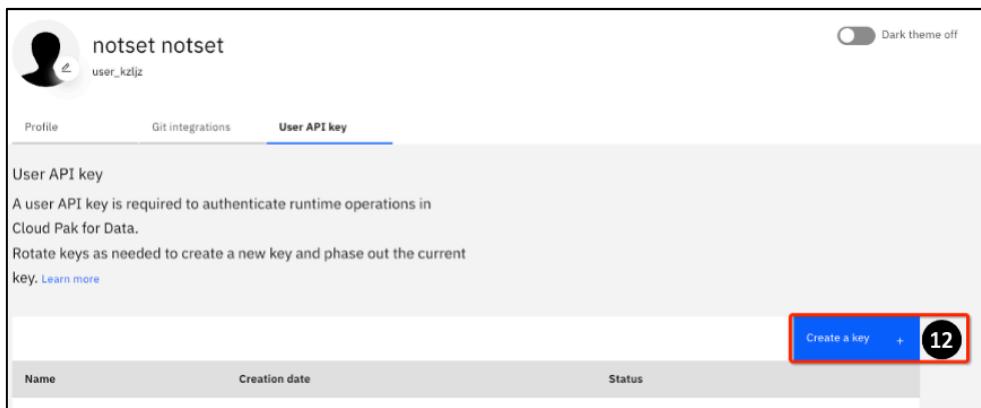
9. On the **flan-t5-base** page, click **New deployment**.

10. The **Create a deployment** page opens. You may see a warning: **No user API key**. To deploy a custom foundation model, you must first create a user API key.

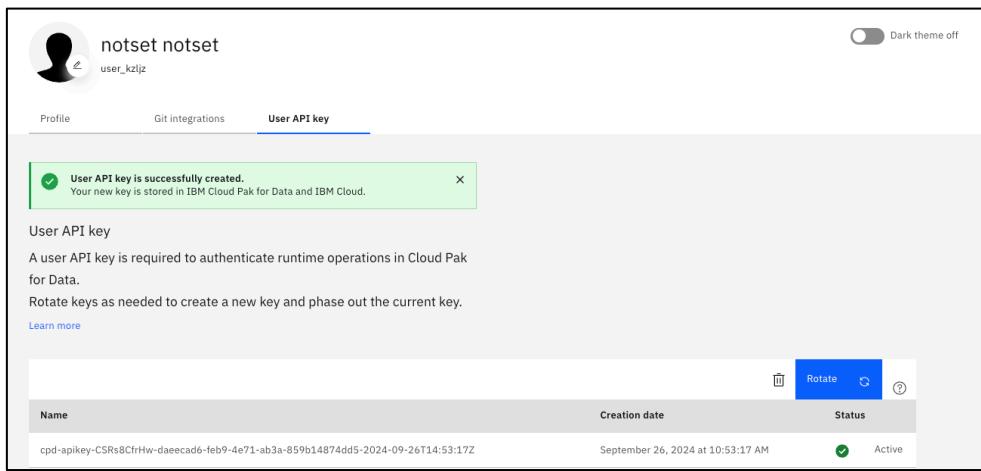
11. Click **Create**.



12. This will open the API creation page. Click **Create a key**. Return to the **Create a deployment** page (likely on a different tab of your browser).



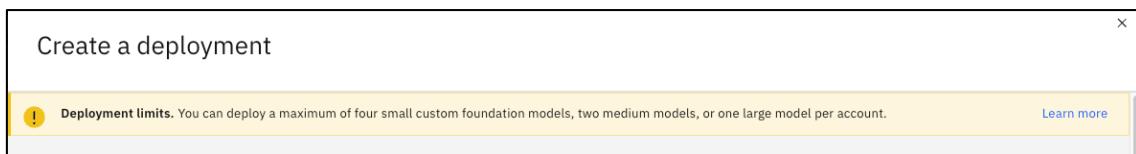
13. Your **User API key** will be created.



14. Go back to the **Create a deployment** window. Instead of **Create**, you will now see **Reload**. Click **Reload**.



The message of the top changes to the following:



15. Note that the message says you can create four small, two medium, or one large model. In this exercise, the model you will deploy is a small one.
16. The **Deployment type** is set to **Online** – this is informational and cannot be changed.
17. For **Name** – this is the model name. Enter **CFM-flan-t5-base**.
18. For **Description**, enter: “**This is a Custom Foundation Model based on flan-t5-base.**”
19. Under **Define configuration**, the **Architecture** is set to **t5**. Leave this as is.
20. Note that there is a cost to model deployment. This means two things:
 - a. Please be considerate – this exercise uses a small model for a small deployment. That is sufficient for a demo. You should not use a medium or large unless you are in a situation that calls for larger-sized models, for example, if a client demands it in a Proof of Experience exercise.
 - b. **You must remove** your deployment **after you have completed the lab** using the instructions in Section 6. Otherwise, these costs will continue to be incurred and charged back to you.
21. You can expand and look at **Model deployment parameters** if so desired.
22. Click **Create**.

Create a deployment

CFM-flan-t5-base.

Deployment limits: You can deploy a maximum of four small custom foundation models, two medium models, or one large model per account. [Learn more](#)

Define details	Define configuration
Associated asset flan-t5-base	Architecture t5
Deployment type Online Run the model on data in real-time, as data is received by a web service.	Select configuration Small: \$6.22 per hour
Name CFM-flan-t5-base	Model deployment parameters
Serving name	
Description This is a Custom Foundation Model based on flan-t5-base.	
Tags	Add tags to make assets easier to find. Find or create tags
<input type="button" value="Cancel"/> <input style="background-color: blue; color: white; border: 2px solid red;" type="button" value="Create"/> 22	

The deployment will take some time to complete. In this example, it took 4 minutes.

Deployments / CFM_space / flan-t5-base				
Deployments		Model details		
Name	Type	Status	Tags	Last modified
(ip) CFM-flan-t5-base	Online	✓ Deployed		4 minutes ago notset notset (You)

This model is now ready to be used.

5.9 Using the Custom Foundation Model

Once a model is deployed, you can use it like other models available in watsonx.ai.

1. Open up the watsonx.ai console. If you have not closed the browser tab from Step 22 of the last section (Section 5.8), click **IBM watsonx**.

The screenshot shows the IBM watsonx interface. At the top, there is a navigation bar with the IBM watsonx logo and a notification icon with the number '1'. Below the navigation bar, the URL 'Deployments / CFM_space / flan-t5-base' is visible. A 'Deployments' tab is selected, showing a table with columns: Name, Type, Status, Tags, and Last modified. There is one entry: 'CFM-flan-t5-base' (Type: Online, Status: Deployed, Last modified: 1 hour ago). A 'New deployment' button is located at the top right of the table area.

2. The watsonx.ai console opens. Click **Open Prompt Lab**.

The screenshot shows the watsonx.ai interface after opening the Prompt Lab. The top navigation bar includes the IBM watsonx logo, a notification icon with the number '2', and location information '2695539 - itz-watsonx-14 Dallas NN'. Below the navigation bar, a welcome message 'Welcome back, notset' is displayed. On the left, there is a sidebar with a 'Customize my journey' button. In the center, there are three cards: 'Chat and build prompts with foundation models' (with 'Start chatting...' button), 'Tune a foundation model with labeled data' (with 'with Tuning Studio' button), and 'Request or track models in AI use cases' (with 'with AI governance' button). A red box highlights the 'Open Prompt Lab' button at the bottom right of the central area.

3. Click the **Structured** tab if watsonx.ai does not open in it.

The screenshot shows the watsonx.ai interface with the Prompt Lab open. The top navigation bar is identical to the previous screenshot. The central area shows the 'Prompt Lab' interface with tabs: 'Chat' (highlighted with a red box), 'Structured' (highlighted with a red box and a number '3'), and 'Freeform'. Below the tabs, there is a text input field with placeholder text 'Enter your prompt text.' and a note: 'Remember: This is not a chat interface. Provide instructions and examples to show the model what to do.' A red box highlights the 'Structured' tab. A note at the bottom states: 'When you prompt a text-generating model, the model responds by appending text to your prompt text or continuing your prompt text.'

4. Click the **Sample prompts** icon.

The screenshot shows the watsonx.ai interface with the Prompt Lab open. The top navigation bar is identical. The central area shows the 'Prompt Lab' interface with tabs: 'Chat' (highlighted with a red box), 'Structured' (selected and highlighted with a red box and a number '4'), and 'Freeform'. Below the tabs, there is a text input field with placeholder text 'Enter your prompt text.' and a note: 'Remember: This is not a chat interface. Provide instructions and examples to show the model what to do.' A red box highlights the 'Sample prompts' icon (a small square icon with a white circle) next to the 'Chat' tab. A note at the bottom states: 'When you prompt a text-generating model, the model responds by appending text to your prompt text or continuing your prompt text.'

5. Click **Earnings call summary** from the list of **Sample prompts** slide-out.

The screenshot shows the 'Sample prompts' section of the Prompt Lab interface. On the left, there's a sidebar with 'Summarization' and 'Classification' sections. The 'Classification' section contains three items: 'Meeting transcript summary', 'Scenario classification', and 'Feedback classification'. The 'Meeting transcript summary' item is under the 'Summarization' section. The 'Earnings call summary' item is highlighted with a red box and a black circle containing the number 5.

6. The example earnings call summary is loaded. Click on the default **flan-ul2-20b** model, then click **View all foundation models**.

The screenshot shows the 'Set up' section of the Prompt Lab interface. At the top, it says 'Instruction (optional)' with a placeholder 'Tell the model what to do. For example: Summarize the transcript.' Below that is 'Examples (optional)'. Under 'Input' and 'Output', there are fields with placeholder text. To the right, there's a 'Recent' dropdown menu. The 'flan-ul2-20b' model is selected. At the bottom right of the dropdown menu, there's a button labeled 'View all foundation models' with a black circle containing the number 6.

7. Scroll down to find and click on your deployed model: **CFM-flan-t5-base**.

Select a foundation model

To choose a model, review characteristics such as tasks that models perform. Compare model benchmarks with scores in the range 0–100. Higher scores are better.

All models Model benchmarks

Search for a model or task

 granite-20b-multilingual	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	 granite-34b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	 granite-3b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	 granite-7b-lab	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.
Provider: IBM	Type: InstructLab	Provider: IBM	Type: Provided model	Provider: IBM	Type: Provided model	Provider: IBM	Type: InstructLab
 llama-8b-code-instruct	The Llama model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	 llama-3-1-70b-instruct	Llama-3-1-70b-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-3-1-8b-instruct	Llama-3-1-8b-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-3-2-11b-vision-instruct	Llama-3-2-11b-vision-instruct is an auto-regressive language model that uses an optimized transformer architecture.
Provider: IBM	Type: Provided model	Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model
 llama-3-2-1b-instruct	Llama-3-2-1b-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-3-2-3b-instruct	Llama-3-2-3b-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-3-2-90b-vision-instruct	Llama-3-2-90b-vision-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-3-405b-instruct	Llama-3-405b-instruct is Meta's largest open-sourced foundation model to date, with 405 billion parameters, optimized for dialogue use cases.
Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model
 llama-3-70b-instruct	Llama-3-70b-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-3-8b-instruct	Llama-3-8b-instruct is an auto-regressive language model that uses an optimized transformer architecture.	 llama-guard-3-11b-vision	Llama-guard-3-11b-vision is an auto-regressive language model that uses an optimized transformer architecture.	 llama3-llava-next-8b-hf	Llama3-llava-next-8b-hf is an auto-regressive language model that uses an optimized transformer architecture.
Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model
 mistral-large	Mistral Large, the most advanced Large Language Model (LLM) from Mistral AI, is an experimental model. Thanks to its state-of-the-art reasoning capabilities it can be applied to any AI-related task, including the most challenging ones.	 mistral-8x7b-instruct-v01	The Mistral-8x7B Large Language Model (LLM) is a pre-trained generative Sparse Mixture of Experts.	 CFM-flan-t5-base			
Provider: Mistral AI	Type: Provided model	Provider: Mistral AI	Type: Provided model	Provider: IBM	Type: Custom model		

- The **CFM-flan-t5-base** model card opens. Note that the **Description** entered in Step 18 or Section 5.8 appears. You can use the **Description** field to provide more details about the Custom Foundation Model if necessary.

- Click **Select model**.

CFM-flan-t5-base

Provider: IBM, tuned by nosetl/nosetl | Type: Custom model

Description

This is a Custom Foundation Model based on flan-t5-base.

8

Back Select model 9

- The **CFM-flan-t6-base** model is selected for this example.

- Click **Generate**.

The screenshot shows the WatsonX AI studio's Prompt Lab interface. On the left, there's a sidebar with various project templates like 'Meeting transcript summary', 'Earnings call summary', 'Scenario classification', 'Feedback classification', 'Marketing email generation', 'Thank you note generation', 'Fact extraction', 'Questions about an article', and 'Questions about insurance'. The main area is titled 'Set up' and contains sections for 'Instruction (optional)' and 'Examples (optional)'. In the 'Try' section, there's a 'Test your prompt' input field with the text 'Financial Highlights' and an output panel showing the generated response: 'I'll start with the financial highlights of the fourth quarter. We deliver... \$8 billion in revenue, \$1.5 billion of operating pre-tax income and earnings per share of \$1.5. Our seasonally strongest quarter, we generated \$2.5 billion of free cash flow.' At the bottom right of the main area, there's a 'Generate' button.

You will see the following output:

The screenshot shows the AI-generated output in a light blue box. The text reads: 'The fourth quarter was a strong year for us. We delivered \$8 billion in revenue, \$1.5 billion of operating pre-tax income and operating earnings per share of \$1.5. Our seasonally strongest quarter, we generated \$2.5 billion of free cash flow.'

As a side note, this may not be the best summary (the default flan-ul2-20b does better), but this lab was only meant to show that once you have imported a Custom Foundation Model, it can be used just like any other models in the watsonx.ai studio.

5.9.1 Considerations for Custom Foundation Models

Custom Foundation Model is a powerful tool that allows clients to bring models that they are familiar with, or have already tuned, to the watsonx.ai platform and leverage the various features and capabilities available there. This lab showed the process of importing a model.

Here are some further considerations.

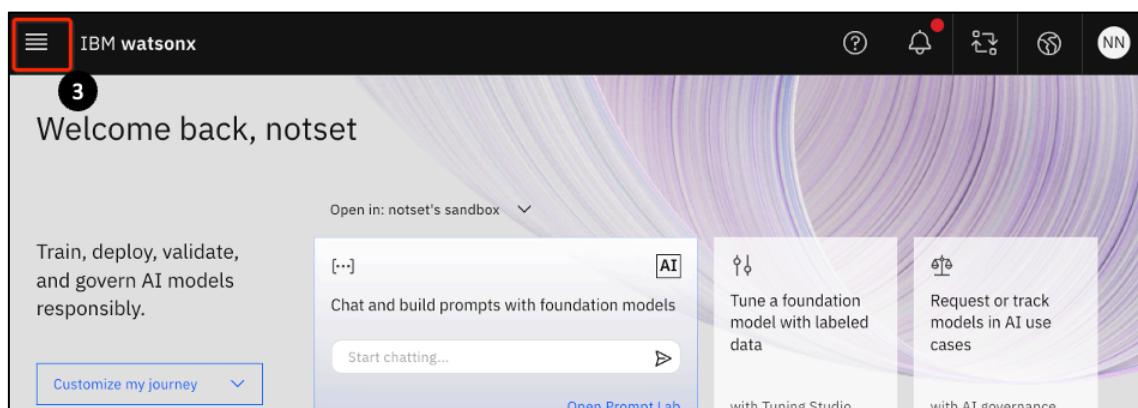
1. Not all models can be imported. Always check the [supported architecture](#) link to ensure the model they have in mind is supported.
2. Hugging Face models typically include a model card. However, the current import process does not pull this in. As described in Step 18 of Section 5.8, you can enter information relevant to the model as the **Description** when the model is deployed.
3. Currently, in a watsonx.ai account, you can deploy a maximum of four small models, two medium models, or a single large model (you can, of course, remove deployments before trying a new one if the limit is reached). This is meant for deploying tried and true, known models versus a way to bring in different models to test/try out if they work for your business use cases.
4. There is a cost associated with hosting a Custom Foundation Model, and that depends on the size of the model. See the Hourly billing rates in the custom foundation models section in [Watson Machine Learning plans and compute usage](#).
5. There are some current limitations (for example, you cannot tune a Custom Foundation Model). See [Planning to deploy a custom foundation model](#) to find the details.

You must complete the steps outlined in Section 6 as a final task for this lab.

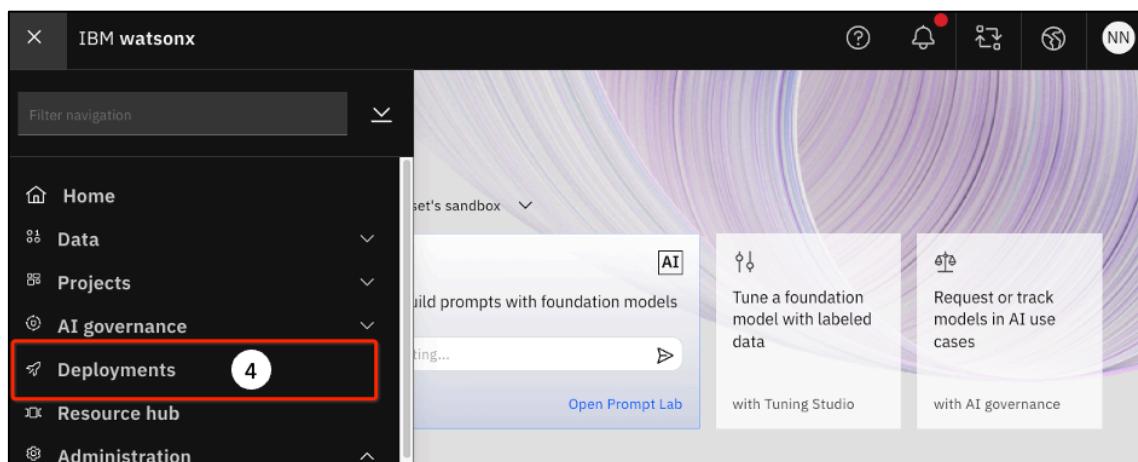
6. Removing a deployment

You must remove your deployment when you are finished with this exercise, or there will be an ongoing cost that will be charged to you.

1. If you have logged out of your Student ID account, log back in using the process outlined in Section 5.4.2. If you have not, skip to point 3.
2. Once logged in, bring up the watsonx.ai console using steps 1-10 of Section 5.3.
3. Click the main menu on the upper left.



4. Scroll down on the slide-out menu and click **Deployments**.



5. The **Deployments** page opens. Click **CFM_space**.

The screenshot shows the 'Deployments' page with the 'Spaces' tab selected. A single deployment space, 'CFM_space', is listed. The row for 'CFM_space' is highlighted with a red box and a black circle containing the number 5.

Name	Last modified	Your role	Collaborators	Tags	Type	Online deployments	Jobs
CFM_space	Sep 26, 2024, 9:53 AM	Admin	NN		Development	1	0

6. The **CFM_space** page opens. Click the **Deployments** tab if it did not open to it.
7. Click on the three vertical dots corresponding to **CFM-flan-t5-base** deployment space.

The screenshot shows the 'CFM_space' deployment page with the 'Deployments' tab selected. A single deployment, 'CFM-flan-t5-base', is listed. The row for 'CFM-flan-t5-base' has a red box around its three-dot menu icon and a black circle containing the number 7.

Name	Type	Status	Asset	Tags	Last modified
CFM-flan-t5-base	Online	Deployed	View		15 hours ago notset notset (You)

8. Click **Delete**.

The screenshot shows the 'CFM_space' deployment page with the 'Deployments' tab selected. A single deployment, 'CFM-flan-t5-base', is listed. The 'Delete' button in the row's three-dot menu is highlighted with a red box and a black circle containing the number 8.

Name	Type	Status	Asset	Tags	Last modified
CFM-flan-t5-base	Online	Deployed	View		15 hours ago notset notset (You)

9. Click **Delete**.

The screenshot shows a confirmation dialog box asking 'Delete the deployment **CFM-flan-t5-base**?'. It states that deleting the deployment will make it unavailable for all consumers. The 'Delete' button at the bottom right is highlighted with a green box and a black circle containing the number 9.

Delete the deployment **CFM-flan-t5-base**?

Deleting the deployment will make it unavailable for all consumers.

Cancel Delete

Appendix A. Revision History

Date	Changes
October 2024	Original version.