

Environment Setup Instructions

Introduction

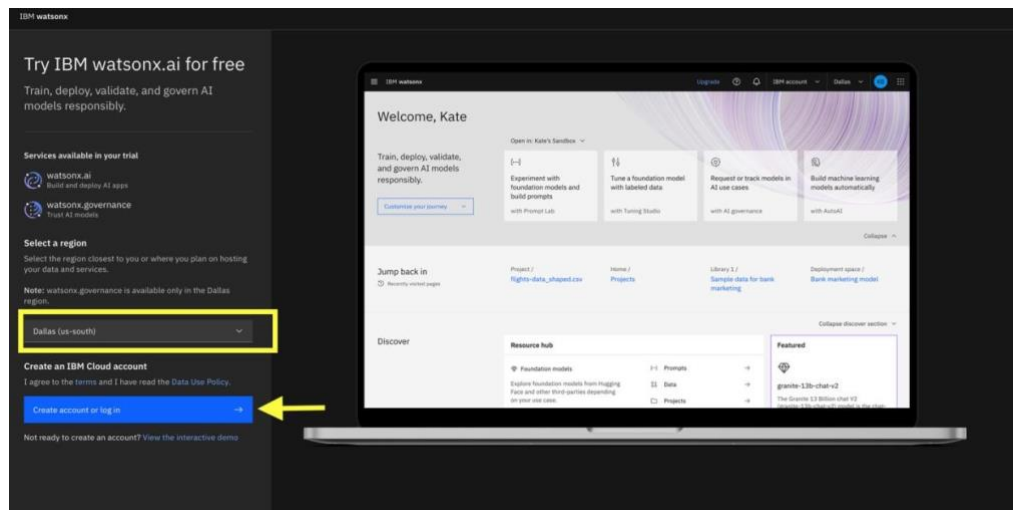
Complete the steps in this guide to ensure your desktop environment has all the required tools/libraries installed and ensure you have the necessary IBM Cloud access.

1. Obtaining your free IBM watsonx ai account

1.1 Create an IBM watsonx account

1.1.1 Register your free account without credit card required here:

https://dataplatform.cloud.ibm.com/registration/stepone?context=wx&preselect_region=true



Make sure to choose the region as “**Dallas (us-south)**” and create account or log in.

1.1.2 You will receive a 7-digits verification code through your email. Type them in and filling personal information.

Create an IBM Cloud account

Already have an IBM Cloud account? Log in

Account information [Edit](#)

mewibmtest4@gmail.com

Verify email

We sent a verification code to mewibmtest4@gmail.com

Verification code

1

Kandana.Lee@phong@ibm.com - saved password

[Resend code](#)

Personal information

Account notice

[Continue](#)

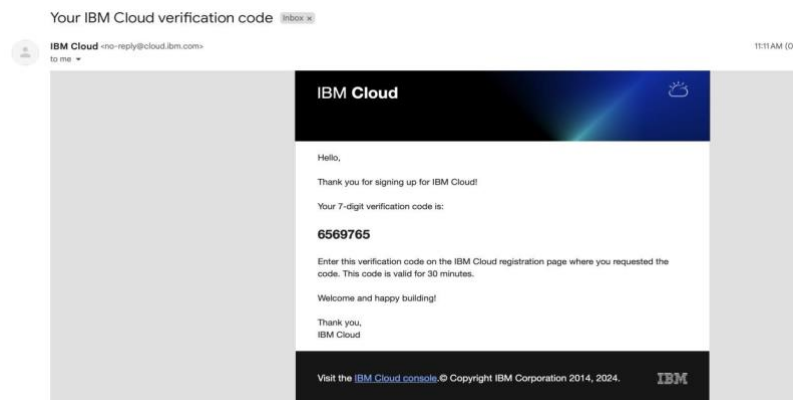
Get started for free with **watsonx.ai™**

Build and deploy generative AI and machine learning solutions.

Work with a range of open source or proprietary foundation models to build generative AI solutions with minimal data.

Choose from a range of tools like open-source code editors (Studio, Jupyter Notebooks), automated builders and visual modeling tools.

Manage your entire lifecycle from model development to management from a single collaborative studio.



1.1.3 After click continue, you will be directed here. Please agree to terms and press continue

Review your account privacy notice

About your IBMid Account

By creating, or using, an IBMid to access an IBM Web site or other service, you acknowledge that you have read, understand, and agree to be bound by the below terms, in addition to the terms in the IBM Terms of Use, the IBM Privacy Agreement and the "About your IBMid Account Privacy" section of this notice provide details on how IBM protects your personal data.

IBMid accounts are individual IDs, meaning you manage your own account information and access your account through the IBMid login service (a IBMid is the authenticating agency).

If your account is created using an email address containing a domain owned by an organization which you are employed by, contacted to, or volunteer for, your organization can:

- inquire about the status of your ID
- request your account settings (including your personal information)
- at its option, convert it to an enterprise ID.

If your account was created as (or is converted) to an enterprise ID, your organization manages your account information. It is also the authenticating agency for your ID, meaning your access is controlled by your organization's login service.

(Last updated: 2023-09-30)

What data does IBM collect?

Why IBM needs your data

How your data is obtained

How IBM uses your data

How IBM protects your data

How long we keep your data

About your IBMid Account Privacy

Your rights

Our Privacy Statement provides more information about your personal data rights. It also provides contact information if you have questions or concerns regarding our handling of your personal data.

☒ I acknowledge that I understand how IBM is using my Basic Personal Data and (if applicable) how my organization may become the authenticating agency for my IBMid account. I certify that my age is at least the age of consent for my country of residence.

If I am from the People's Republic of China, I agree my personal information may be transferred outside of the People's Republic of China for the purpose of providing me relevant products and services.

If I have an existing IBMid account and I wish to remove my personal information, I can contact the [IBM Privacy Office](#).

[Continue](#)

Get started for free with **watsonx.ai™**

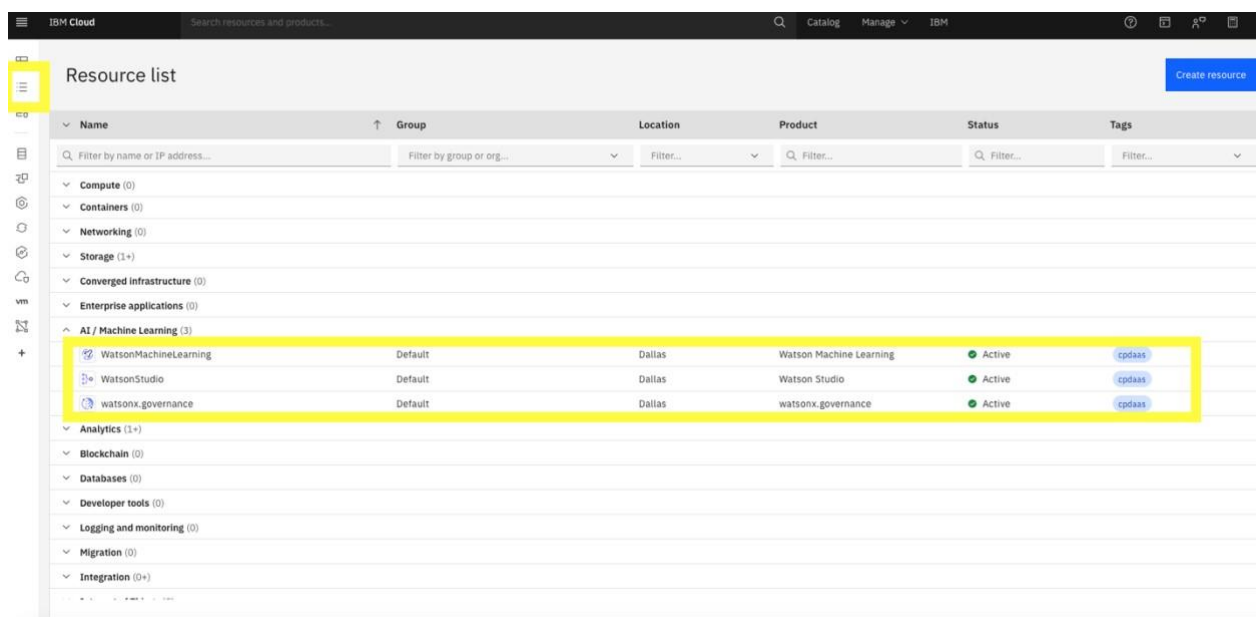
Build and deploy generative AI and machine learning solutions.

1.2 Ensure you have access to all the IBM Cloud services.

After successful registration, sign in through <https://cloud.ibm.com>

Login into IBM cloud account IBM Cloud and ensure you can see the following services in the resource lists

1. watsonx.governance
2. Watson Studio
3. Watson Machine Learning



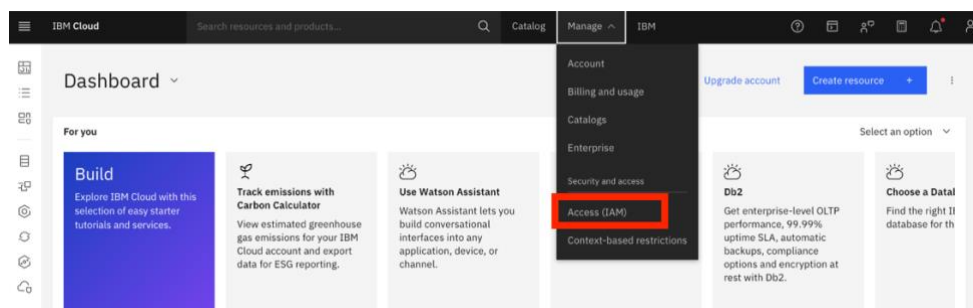
2. Obtain your IBM keys

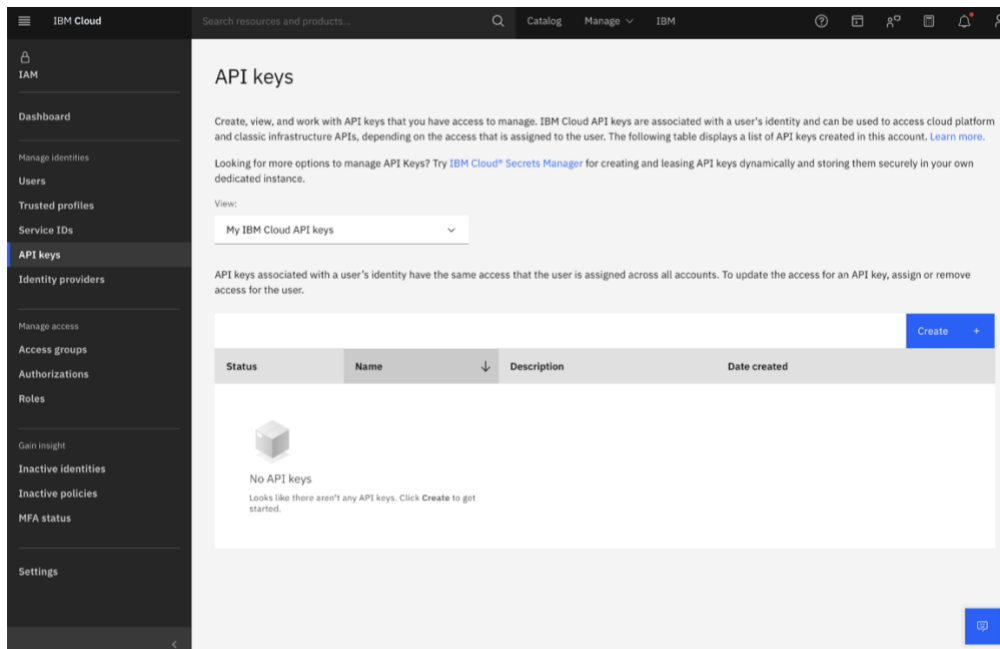
2.1 Obtain your IBM Cloud API key

If you already have an existing API key please use it, otherwise follow [these instructions](#) to generate a new one in the [IBM cloud](#). You will need this API key for next steps.

2.1.1 Login to your IBM cloud account through this [link](#). If you have not had an IBM cloud account, please create one

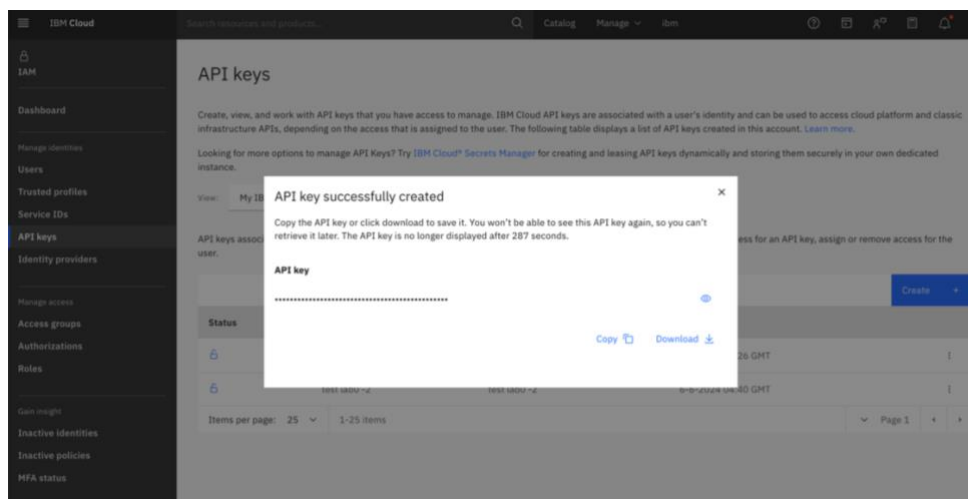
2.1.2 In the IBM Cloud console, go to **Manage > Access (IAM) > API keys**.





- 2.1.3 Click **Create** to create an IBM Cloud API key.
- 2.1.5 Enter a name and description for your API key.
- 2.1.5 Click **Create**.
- 2.1.6 Click **Show** to display the API key.

Please make sure to click **Copy** to copy and save it for later, you will not be able to access them again.
Save this key for later step in this set up.



2.2 Connect to your watsonx.ai instance

Ensure that you can log into to watsonx.ai.

2.3 Locate the Watsonx.ai Project Id.

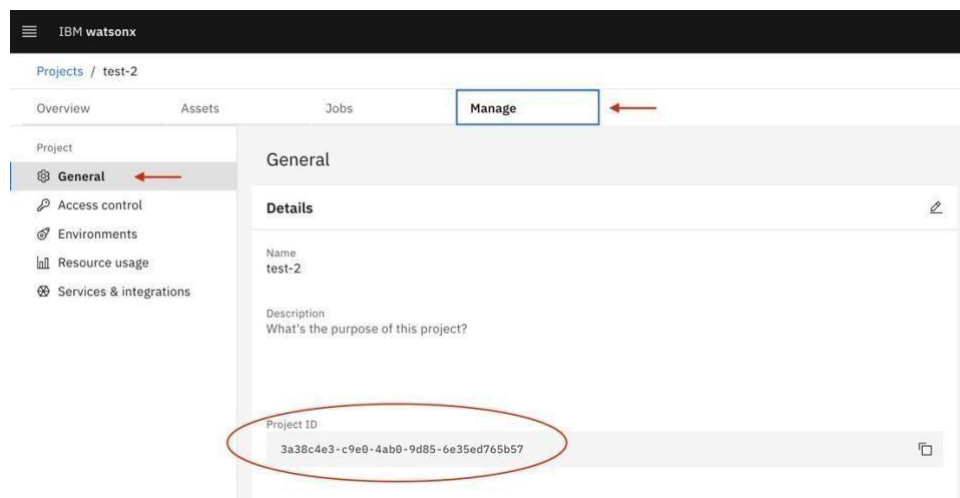
2.3.1 Ensure you are logged into to watsonx.ai.

2.3.2 Select the project under your organization name.

2.3.3 Select the "Manage" tab from your Project's main page.

You will see your Project ID under the "General" tab as shown below.

Save this key for later step in this set up.

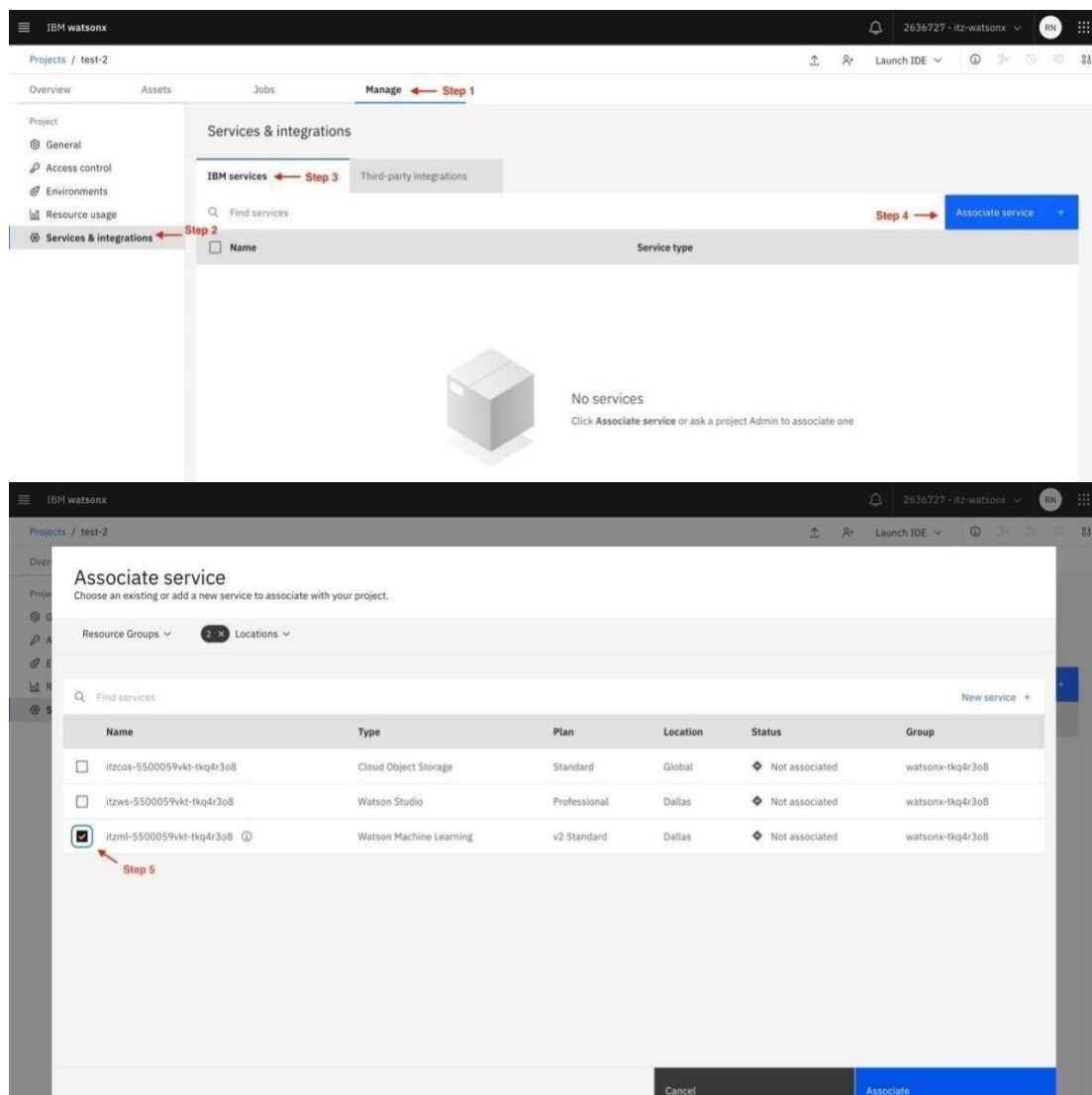


2.4 Associate your project with a WML instance

A watsonx.ai project must always be associated with an instance of Watson Machine Learning (WML) before you can use the Prompt Lab or interact with the WML Python SDK.

If the WML instance is not associated for some reason complete the following steps

1. Within your newly created watsonx.ai project, click the Manage tab
2. Select Services & integrations from the side navigation menu
3. Select the IBM Services tab
4. Select Associate service +
5. Choose a WML instance and select Associate



3. Clone the workshop Git repo

If you are familiar with git command, you can directly clone this GitHub [repository](https://github.com/edsml-kl121/compact-watsonx-ai-incubation-program) by create a new directory in your local computer and run the following command

```
git clone https://github.com/edsml-kl121/compact-watsonx-ai-incubation-program
```

Otherwise, we recommend downloading and installing the [Github Desktop](#) and then follow these instructions on [how to clone a repository using Github Desktop](#).

4. Install Visual Studio Code (VS Code)

We recommend installing VS Code for this workshop, so we are on a common platform.

5. Update credentials in .env file

Python provides support for **.env** files through a library called **dotenv** that we will use in this workshop to pass the credentials.

5.1 Create a new file inside the TH folder, and name the file **“.env”**. If you have created the file, but are having trouble viewing it, [learn how to view hidden files on a Mac](#) or [how to view hidden files on Windows](#).

5.2 Open the .env file, add the following content:

```
WATSONX_APIKEY = <your-ibm-cloud-api-key>
IBM_CLOUD_URL = 'https://us-south.ml.cloud.ibm.com'
PROJECT_ID=<your-project-id>
MILVUS_HOST=<TBC>
MILVUS_PORT='8080'
MILVUS_SERVER_NAME='localhost'
MILVUS_USER='root'
MILVUS_PASSWORD='4XYg2XK6sMU4UuBEjHq4EhYE8mSF03Qq'
MILVUS_SERVER_PEM_PATH='./cert.pem'
EMB_IBM_CLOUD_URL=<TBC>
EMB_PROJECT_ID =<TBC>
EMB_WATSONX_APIKEY=<TBC>
EMB_SPACE_ID =<TBC>
EMB_DEPLOYMENT_ID =<TBC>
```

Use the IBM_CLOUD_URL given above. The API_KEY and PROJECT_ID need to be filled in by you.

5.3 Add your IBM Cloud API key from Step 2.1.6 in [API_KEY](#)

5.4 Add your project ID from Step 2.3.3 in [PROJECT_ID](#)

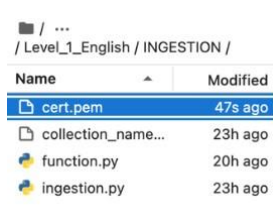
5.5 The remaining will be provided by the watsonx team

Save your changes and close the file.

6. Adding cert.pem file

3.1.2 Add **cert.pem** file that you received in your email inside this lab 0 base directory.

File content starts with **-----BEGIN CERTIFICATE-----** and end with **-----END CERTIFICATE-----**



Name	Modified
cert.pem	47s ago
collection_name...	23h ago
function.py	20h ago
ingestion.py	23h ago

3.1.2 Also add **cert.pem** file to lab 04 directory > INGESTION folder before running lab 04

7. Download Environment

7.1 Install podman

Install **podman** by follow instruction below. For further information, please follow this [link](#).

For Mac installation

1. Make sure you have [homebrew](#) installed in your computer, if not please run this command

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

2. After brew is installed, use this command to install podman

```
brew install podman
```

3. Create your first podman machine

```
podman machine init
podman machine start
```

For windows installation

You can install it by searching the **Windows Store** or by running the following winget command

```
winget install Microsoft.WindowsTerminal
```

Checkout this [link](#) for more information

7.2 Running the lab with podman/docker

Execute the following in terminal.

1. Pull images of the environment that will be used in the lab

```
podman pull u1800085/ibm-watsonx-incubation-program-v2:incubation_watsonx_th_latest
```

2. Start the container with the image that was pulled. Please ensure you are executing this outside the `gen_ai_incubation_watsonx_th` directory

```
podman run -p 8888:8888 -p 8501:8501 -v ./:/usr/src/app --name incubation
docker.io/u1800085/ibm-watsonx-incubation-program-v2:incubation_watsonx_th_latest
```

3. Enter the [localhost:8888](#) on your web browser for **LAB 1-2**
4. For **LAB 3-4** please open another terminal and execute the following command

```
podman exec -it incubation /bin/bash
```

then cd to your lab's base directory to run your command.

7.3 ONLY If docker/podman installation is unsuccessful you can try the following method, otherwise **skip this step**. (Create a virtual python environment and install all required libraries.)

Install all the python libraries using this [requirements_venv.txt](#).

You can use your favourite python package manager and create a virtual environment called genai and install all the python using this [requirements_venv.txt](#). For windows users, it is recommended to use conda.

```
conda create --name genai python=3.11 conda activate genai
pip install -r requirements_venv.txt
```

Optionally, if you want to use a virtual environment using `venv`, follow the steps below.

1. Upgrade to Python v3.11 to avoid any conflicts: Minimum python version needed for our workshop is 3.11. Upgrade your python version to Python 3.11
2. Create your Python virtual environment:

- a. Create a folder <my-folder>
- b. Open a terminal/console window and enter the commands below to create a Python environment called `genai`.

```
cd <directory to store your Python environment>
```

```
python -m venv genai
```

- c. Download [requirements_venv.txt](#)
- d. Move the [requirements_venv.txt](#) file to the folder <my-folder>
- e. Activate your Python virtual environment with these commands:

Mac-

```
source genai/bin/activate python -m pip install -r requirements_venv.txt
```

Windows-

```
.\genai\bin\activate python -m pip install -r requirements_venv.txt
```

- f. Validate that the start of the prompt line in your terminal/console window changed to genai.

```
(base) anthonystevens@anthonys-mbp Python.venv % python -m venv genai
(base) anthonystevens@anthonys-mbp Python.venv % source genai/bin/activate
(genai) (base) anthonystevens@anthonys-mbp Python.venv % █
```